Hindawi Publishing Corporation
Active and Passive Electronic Components
Volume 2012, Article ID 763572, 10 pages
doi:10.1155/2012/763572

Research Article

Thermal-Aware Test Schedule and TAM Co-Optimization for Three-Dimensional IC

Chi-Jih Shih,[1] Chih-Yao Hsu,[1] Chun-Yi Kuo,[1] James Li,[1] Jiann-Chyi Rau,[2] and Krishnendu Chakrabarty[3]

[1] *Department of Electrical Engineering, National Taiwan University, Taipei 10617, Taiwan*
[2] *Department of Electrical Engineering, Tamkang University, New Taipei City 25137, Taiwan*
[3] *Department of Electrical and Computer Engineering, Duke University, Durham, NC 27708, USA*

Correspondence should be addressed to Chi-Jih Shih, terrys47@hotmail.com

Received 27 April 2012; Revised 7 October 2012; Accepted 18 October 2012

Academic Editor: Gerard Ghibaudo

Testing is regarded as one of the most difficult challenges for three-dimensional integrated circuits (3D ICs). In this paper, we want to optimize the cost of TAM (test access mechanism) and the test time for 3D IC. We used both greedy and simulated annealing algorithms to solve this optimization problem. We compare the results of two assumptions: *soft-die mode* and *hard-die mode*. The former assumes that the DfT of dies cannot be changed, while the latter assumes that the DfT of dies can be adjusted. The results show that thermal-aware cooptimization is essential to decide the optimal TAM and test schedule. Blindly adding TAM cannot reduce the total test cost due to temperature constraints. Another conclusion is that soft-die mode is more effective than hard-die mode to reduce the total test cost for 3D IC.

1. Introduction

Three-dimensional integrated circuits (3D ICs) provide a promising solution to process scaling and heterogeneous system integration [1–3]. In spite of many advantages, 3D ICs still have many challenges ahead. Among them, high temperature issue is probably the most critical one, because vertical heat dissipation paths in 3D ICs are longer than those in 2D IC [4–7]. Thus, high temperatures cause serious yield loss problem when testing 3D ICs.

Many papers have proposed algorithm of test schedule optimization for 2D IC [8, 9], including thermal-aware test scheduling [10–13]. In [10], two optimization algorithms are proposed which try to spread heat more evenly in a chip via layout information and a progressive weight function. A rectangular 2D bin packing can solve the test scheduling problem by considering dynamic thermal profiles [11]. A thermal-safe test scheduling method used resource conflict graph for optimization [12]. After a test schedule is obtained, a 2D thermal resistance model is applied to check whether the thermal constraint is met. This technique, however, does not consider the TAM constraint. The thermal-resistance model which used superposition principle has been introduced for 2D IC test scheduling optimization [13]. Many techniques used integer linear programming (ILP) to find an optimal solution. However, when thermal constraints are considered, there could be an exponential growth in the problem size because of the need for evaluating all possible combinations. A die-level test scheduling method for 3D IC was proposed in the previous work [14]. In their work, they addressed the issue of test scheduling to minimize overall test time for stack testing as well as postbond testing without temperature consideration. Previous research in this area lacks the consideration of thermal constraints when dealing with test time and TAM width tradeoff.

The purpose of this paper is to propose a test scheduling method for postbond 3D IC testing to determine the optimal test time and TAM width under the temperature constraint. Two optimization modes can be chosen: *hard-die mode* and *soft-die mode*. The hard-die mode assumes a fixed DfT architecture, where the number of scan chains and TAM assignment cannot be changed. The soft-die mode assumes a configurable DfT architecture, where the number of scan chains and TAM assignment can be changed. In

the optimization process, a simple thermal resistance model is used to quickly estimate the maximum temperature. The temperature of the final test schedule is verified by an academic thermal simulator, *Hotspot* [15]. The contributions of this paper are listed as follows.

(i) A thermal-aware test scheduling and TAM co-optimization method for 3D ICs.

(ii) Two optimization modes are supported for different 3D IC configurations.

(iii) Simplified and accuracy thermal resistance model for temperature estimation to speed up the optimization process.

Thermal-aware co-optimization is essential to decide the optimal TAM assignment and test scheduling. This paper shows the following three important key results.

(i) When the number of TAM is smaller than a threshold, the test time is *TAM limited*. At this stage, adding TAM helps to reduce test time.

(iii) When the number of TAM is larger than the threshold, the test time is *temperature limited*. At this stage, adding TAM is a waste of resource without contribution to test time reduction.

(iii) Compared with the hard-die mode, soft-die mode produces more effective test cost reduction. DfT architecture of each core should be optimized together with the whole 3D IC.

This paper is organized as follows. Section 2 introduces our assumptions and defines the problem. Section 3 describes the details of the proposed test scheduling technique. Section 4 shows our experimental results on three 3D ICs. Finally, Section 5 concludes this paper.

2. Assumptions and Models

2.1. Assumptions. We assume that each core has the same test time, power, and TAM width in different temperatures. The power of the core in the test mode is higher than the function mode [16]. In this paper, we only considered the temperature issue in the test mode. We do not consider the prebond test in this test scheduling. Thermal issue is not serious in prebond test partly because there is no die stacking, and partly because pre-bond tests are usually performed at slow speed.

In our proposed technique, the whole test scheduling is divided into many *slots*, or *test sessions*. Figure 1 shows two examples of test scheduling with and without temperature constraint, where test sessions are separated by vertical bars. Each rectangle corresponds to a *core under test*—the height represents the TAM width while the width represents the test time. The TAM limit (T_{limit}) is indicated as dotted lines in Figure 1. All cores in the same slot are assumed to start at the same time. Cores in the same slot start to be tested concurrently. Because we adopt IEEE 1500 compatible core wrappers [17], the cores under test should be configured via *WIR chains* at the beginning of the test session where the cores are tested. The WIR chain is a scan chain that connects wrapper instruction register (WIR) of different cores together. Figure 1(a) shows an optimized test schedule without temperature constraint. Because many cores are tested in the forth test session, it is overheated (highlighted in red). Figure 1(b) shows the optimized test schedule, which is not overheated, although its test time is slightly longer than the original test schedule. Figure 1 shows that test scheduling with thermal constraints is important for 3D IC.

In this work, we assume that the heat sink is not used in production test for cost reduction. In production test, heat sinks and heat spreaders are not installed to save test cost. As a result, test scheduling must consider temperature constraint to avoid overheating in test mode. In this paper, we only consider the steady state temperature during optimization. This is because dynamic temperature can settle within milliseconds, which is shorter than a test session. We assume that the heat is only generated by the power consumption of cores under test, ignoring the power of TSV drivers, which are very few in numbers.

In the hard-die optimization mode, we are given a 3D IC which has totally M dies and W_{limit} TAM. Since the DfT architecture is fixed, test time of each core remains unchanged during the optimization. Given the maximum temperature constraint, T_{limit}, the goal of this co-optimization problem is to find the lowest test cost with the consideration of both test time and TAM width.

2.2. Thermal Models for 3D IC. In the optimization process, whenever a new test scheduling is generated, its peak temperature has to be estimated. The peak temperature of a test schedule is the maximum temperature of every test session in the test schedule. Exact thermal simulation is very time consuming so a simple 3D IC temperature estimation is needed. In this work, we adopt the thermal resistance model [15], where the vertical heat flow is modeled as electrical current and the temperature is modeled as electrical voltage. Vertical heat conduction between two adjacent dies is modeled as thermal resistance.

The 3D IC is divided into a two-dimensional array of *tile stacks*. Figure 2(a) shows an example of three-layer 3D IC divided into 3×2 tile stacks. A single tile stack (Figure 2(b)) contains three layers of tiles. Each layer represents a die and each square represents a unit area on the die (*resolution* of thermal estimation). Figure 2(c) shows the corresponding thermal model of a single tile stack. The power dissipation of tile i (P_i) is regarded as a current source while temperature (T_i) is regarded as voltage. The thermal resistance of tile i is modeled as a resistor (R_i). R_b is represented as the ambient resistance. The ambient temperature (specified by the user) is modeled as a voltage source, T_{ambient}.

In our thermal model, the heat flow is assumed unidirectional, from bottom to top. Because I/O pins are accessed at the bottom of our 3D IC model, the heat can only be dissipated from the top of our 3D IC model. We assume the bottom of CUT is connected to a board, which has been heated by previous testing, so heat propagation to the bottom die is ignored. The peak temperature of a single tile stack

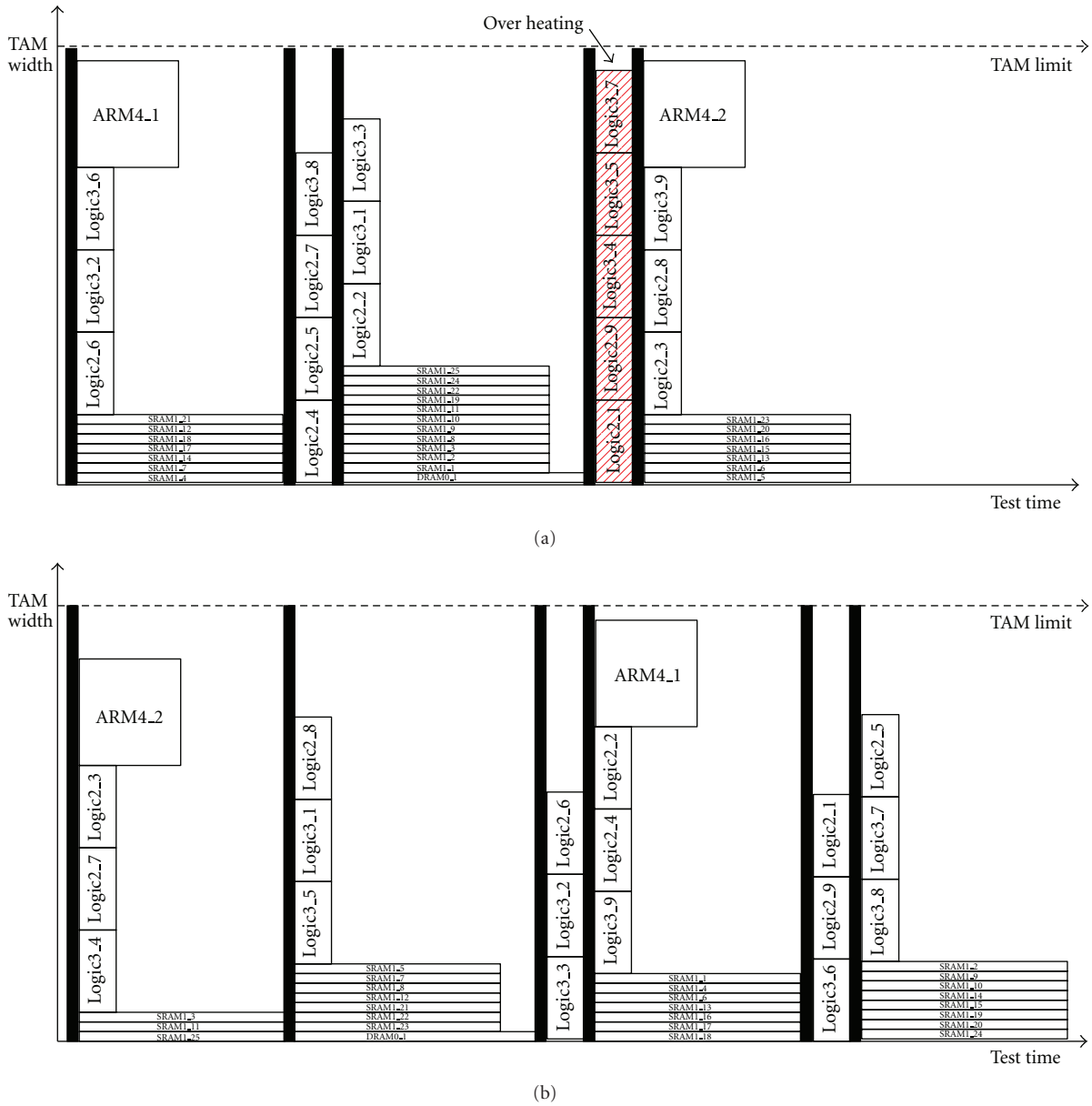

Figure 1: An example test schedule: (a) without temperature constraint and (b) with temperature constraint.

in Figure 2(c) is T_0 at the bottom die, which can be calculated by the following equation:

$$T_0 = P_1 \times (R_1 + R_2 + R_3 + R_b) + P_2 \times (R_2 + R_3 + R_b) + P_3 \times (R_3 + R_b) + T_{\text{ambient}}. \quad (1)$$

In our thermal model, we assume the thickness of each die is 50 μm. The thermal resistivity of each die is directly derived from the thermal resistivity of silicon. But the thermal resistivity of bonding interface is calculated. We use *benzocyclobutene* (BCB) as our bonding interface material, whose thermal resistivity is 3.45. We make an assumption that TSV is 1% of total die area. As a result, the average thermal conductivity of bonding interface can be calculated by the following equation. The thermal resistivity is the reciprocal of thermal conductivity:

$$\kappa_{\text{avg}} = \kappa_{\text{TSV}} \times \frac{\text{Area}_{\text{TSV}}}{\text{Area}_{\text{total}}} + \kappa_{\text{BCB}} \times \left(1 - \frac{\text{Area}_{\text{TSV}}}{\text{Area}_{\text{total}}}\right), \quad (2)$$

In this equation, κ_{TSV} is the thermal conductivity of copper, which is 400, and κ_{BCB} is the thermal conductivity of BCB. Area_{TSV} is the area of TSV, and $\text{Area}_{\text{total}}$ is the total area of die. κ_{avg} is the average thermal conductivity of bonding interface.

Please note that our test scheduling technique is independent of the thermal model. We could include downward heat propagation to the board or also include the lateral heat propagation by adding more thermal resistances into our thermal model [15]. In this way, the accuracy of our thermal model will be improved.

```
begin
Get an initialize schedule[0];
Get an initialize temperature T > 0;
Set the temperature threshold T′;
Set the decay rate r < 1;
while T < T′ do
   for 1 ≦ i ≦ P do
      next ← a random selected perturbation of current
      ΔE ← value[next] − value[current]
      if ΔE < 0 then current ←next
      else current ← next only with probability e^(ΔE/−T)
   T ← rT;
   end for
end while
```

Algorithm 1

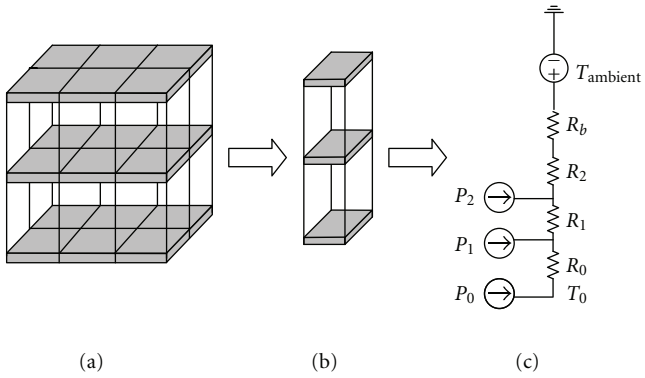

Figure 2: (a) A 3×2 array of tile stacks. (b) A tile stack. (c) Thermal resistance model of a tile stack.

3. Proposal Test Scheduling Technique

3.1. Overall Flow. Our co-optimization tool supports two modes: hard-die mode and soft-die mode. The hard-die mode uses the greedy algorithm to minimize both the total TAM width and test time in the hard-die mode flow of Figure 3. Users can specify a TAM limit (W_{limit}) and a temperature limit (T_{limit}). We initialize a constraint ($W_{\max} \leq W_{\text{limit}}$) and we schedule one core at a time using the greedy algorithm. We sort cores in descending order according to the test time of them. We choose the first core that has the longest test time and put it in the first slot. After a slot is finished, we estimate the maximum temperature of all dies ($T_{\max}$) using the thermal model. We also need to calculate the TAM width (W) used by all dies. We need to verify whether the temperature and TAM constraints are met: $T_{\max} \leq T_{\text{limit}}$ and $W \leq W_{\max}$. If any core violates the above two conditions, we reschedule the core to another slot. After all cores are finished, we estimate the total test cost. Then, we increase $W_{\max}$ by one unit and redo the whole process until all $W_{\max}$ have been tried. During the whole optimization, T_{limit} is always fixed, but $W_{\max}$ can be adjusted in each iteration.

The soft-die mode is slightly different from the hard-die mode in the soft-die mode flow of Figure 3. Given a hard-die schedule, we perform simulated annealing to improve the test time under the temperature and TAM constraints. Unlike hard-die mode where core width and length are fixed, in soft-die mode we can adjust the width and length of cores, as long as the test data volume (width times length) remains unchanged.

There are four input files to our optimization tool. The first file provides the power information of each core in the design. The second file describes the floorplanning information such as the location of each core. The third file offers the test information, such as the number of scan chains, the number of test pins, and the number of test cycles. The last file provides the 3D IC thermal model, such as thermal resistances and environmental temperatures.

3.2. Greedy Algorithm. This is a simple *first-fit packing* algorithm.

(1) First, we sort the cores in decreasing order according to its test time.

(2) The first core with the largest number of test time is scheduled into the first slot.

(3) Pick the next core and schedule it into the existing slots if it "fits"; otherwise, the core is scheduled to a new empty slot. In this algorithm, a core that fits a slot means both temperature and TAM constraints are met.

(4) Repeat step 3 until all cores are scheduled.

3.3. Simulated Annealing Algorithm. In soft-die mode, after the greedy algorithm, we use simulated annealing to refine the solution. The simulated annealing algorithm is described in Algorithm 1.

The value[*next*] is the cost of the test schedule after perturbation, and the value[*current*] is the cost of the test scheduling before perturbation. The value is calculated by a cost function, which will be shown in (3) later in this section. E is the difference of the current value and the next value.

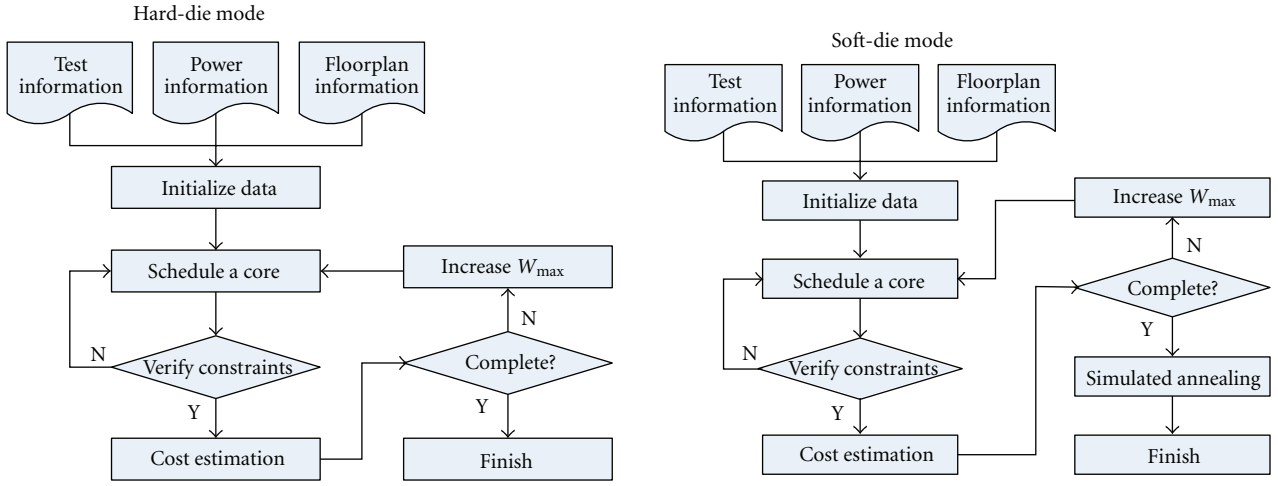

Figure 3: The overall flow.

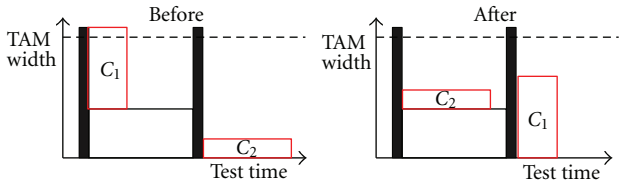

Figure 4: Swap perturbation.

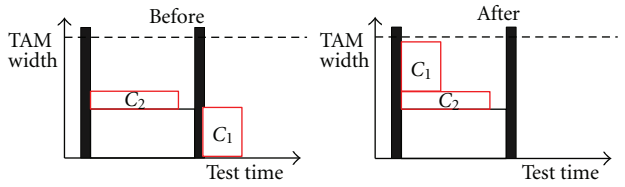

Figure 5: Move-to-existing-slot perturbation.

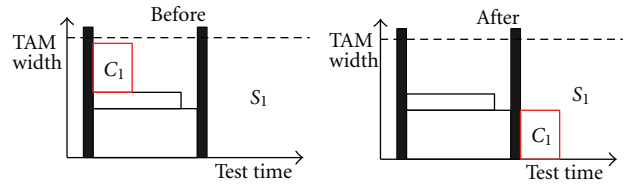

Figure 6: Move-to-empty-slot perturbation.

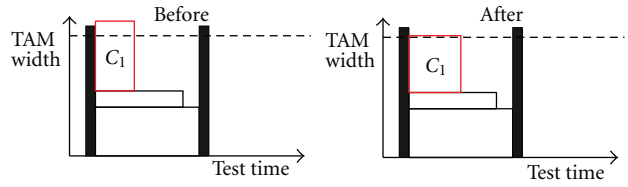

Figure 7: Resize perturbation.

In soft-die mode, four types of perturbation are used in simulated annealing: *swap*, *move-to-existing-slot*, *move-to-empty-slot*, and *resize*. Swap perturbation exchanges two cores in two different slots, and we depict such swap perturbation in Figure 4. Two cores involved in the perturbation are denoted as C_1 and C_2. Before swap, C_1 was over the TAM width constraint. After swap, C_1 and C_2 are both under the TAM width constraint.

The move-to-existing-slot perturbation moves one core to an existing slot that contains at least one test. Figure 5 shows an example of this perturbation. The two cores involved in this perturbation are denoted as C_1 and C_2. C_1 is moved to a slot which is occupied by C_2. The original slot occupied by C_1 is now empty so the test time can be reduced by this perturbation.

Although the above two perturbations can potentially reduce the total test time, the simulated annealing might be stuck in a local optimum. Therefore, the move-core-to-empty-slot perturbation is used to escape the local optimum. Figure 6 shows the move-to-empty-slot perturbation. One core C_1 is selected and moves to an empty slot S_1. Although this increases the total test time, this perturbation may help to escape the local optimum and find a better solution later.

For the fourth perturbation, the selected core C_1 must be a soft core. A soft core has a constraint of its aspect ratio (TAM width : test time). After we parse the test information file of each core, we can get the total test data of each core. The total test data is equal to the TAM width multiplied by test time of each core. Resizing our perturbation, maximum TAM width change allowed is five. The TAM width change is generated randomly in this perturbation. This restriction avoids too much change on a core at a time. Because the generated solution is not too far from the current solution, simulated annealing algorithms are likely to reach a local optimum solution. Figure 7 shows the resize perturbation.

In order to estimate the cost, we define a cost function

$$\text{cost function} = \alpha \frac{\text{test cycle}}{\text{AVG test cycle}} + \beta \frac{\text{TAM width}}{\text{AVG TAM width}}. \quad (3)$$

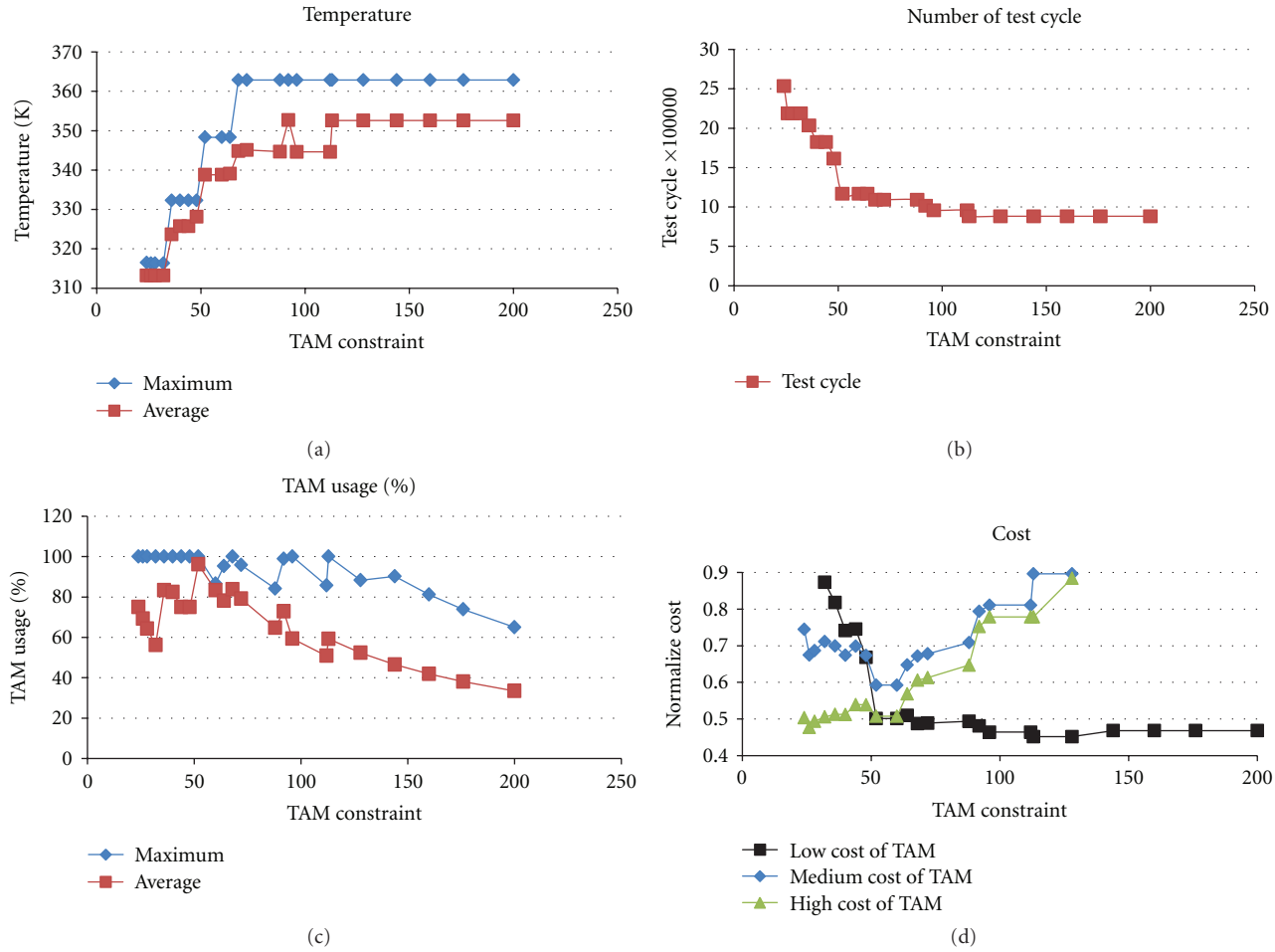

Figure 8: Results of case 1.

To normalize the test time and TAM width, we first perform a hundred random test schedulings. The denominators are the average number of test cycles and the average TAM of a hundred random test schedulings. The α and the β are the weighting coefficients of the test time and TAM width, respectively. If α is larger than β, the program will try to reduce the test time before the reduction of TAM.

To decide the $\alpha:\beta$ ratio, in this paper, we quote the numbers in [18] of a typical 3D IC. The test time for a single die is 6 seconds, and the test cost is \$0.23. Assuming that there are 5 dies stacked in a 3D IC, the test cost for a 3D IC is \$1.15. One wafer has 1,278 dies, so the total manufacturing cost is \$2,779. The total manufacturing cost of all TSVs in one wafer is \$190. Total manufacturing for a 5-layer 3D IC is therefore (\$2,279 + \$190) × 5/1,278 = \$11.6. If we assume that a single TAM requires 10%, 1%, and 0.1% of area overhead, then the ratio of $\alpha:\beta$ can be set to 1 : 10, 1 : 1, and 10 : 1, respectively. The ratio $\alpha:\beta = 1:10$ means high TAM cost with respect to test time. $\alpha:\beta = 10:1$ means low TAM cost with respect to test time. $\alpha:\beta = 1:1$ means test time and TAM are approximately the same. The actual ratio of $\alpha:\beta$ can be adjusted by the users based on real data.

Please note that in this paper, we only consider the postbond test so prebond test and die probing costs are not included. TSV interconnect test time is very short so it is ignored in our test cost.

4. Experiment Results

4.1. 3D IC Test Cases. In this paper, we show results of three 3D IC test cases, each of which consists of five dies, indexed from zero to four. Die number zero (#0) is placed at the bottom of the 3D IC and die number four (#4) is placed on the top. Although there is no heat sink in production test, the heat sink is supposed to be installed on the top die number four (#4) in the system. In 3D IC, *bonding interfaces* are gluing materials between upper and lower dies. The thickness of each die is 50 μm, and the thickness of bonding interface is 2 μm. The thermal resistance of each die is 0.01 (k × m/W) and that of bonding interface is 0.25 (k × m/W) [19].

The first test case is a *heterogeneous* 3D IC, which contains logic dies and memory dies of different technologies. The information of each die is listed in Table 1. The area of each die is the same 5 mm × 5 mm. The second column shows the circuit in each die, and the third column lists the technology process used for each die. The forth column shows the total power consumption of each die. The fifth column is the number of cores in each die. The sixth column shows

Table 1: The first test case.

Die	Circuit	Technology (nm)	Die power (W)	No. of cores	No. of scan chain	No. of test pattern	TAM width (hard-die mode)	Test time (No. of test cycles)
Die 4	Logic	180	36.0	9	15	130	17	76,440
Die 3	Logic	180	36.0	9	15	130	17	76,440
Die 2	ARM9	180	6.0	2	20	300	22	210,000
Die 1	SRAM	90	0.65	25	N/A	N/A	2	425,984
Die 0	DRAM	32	0.3	1	N/A	N/A	2	500,000

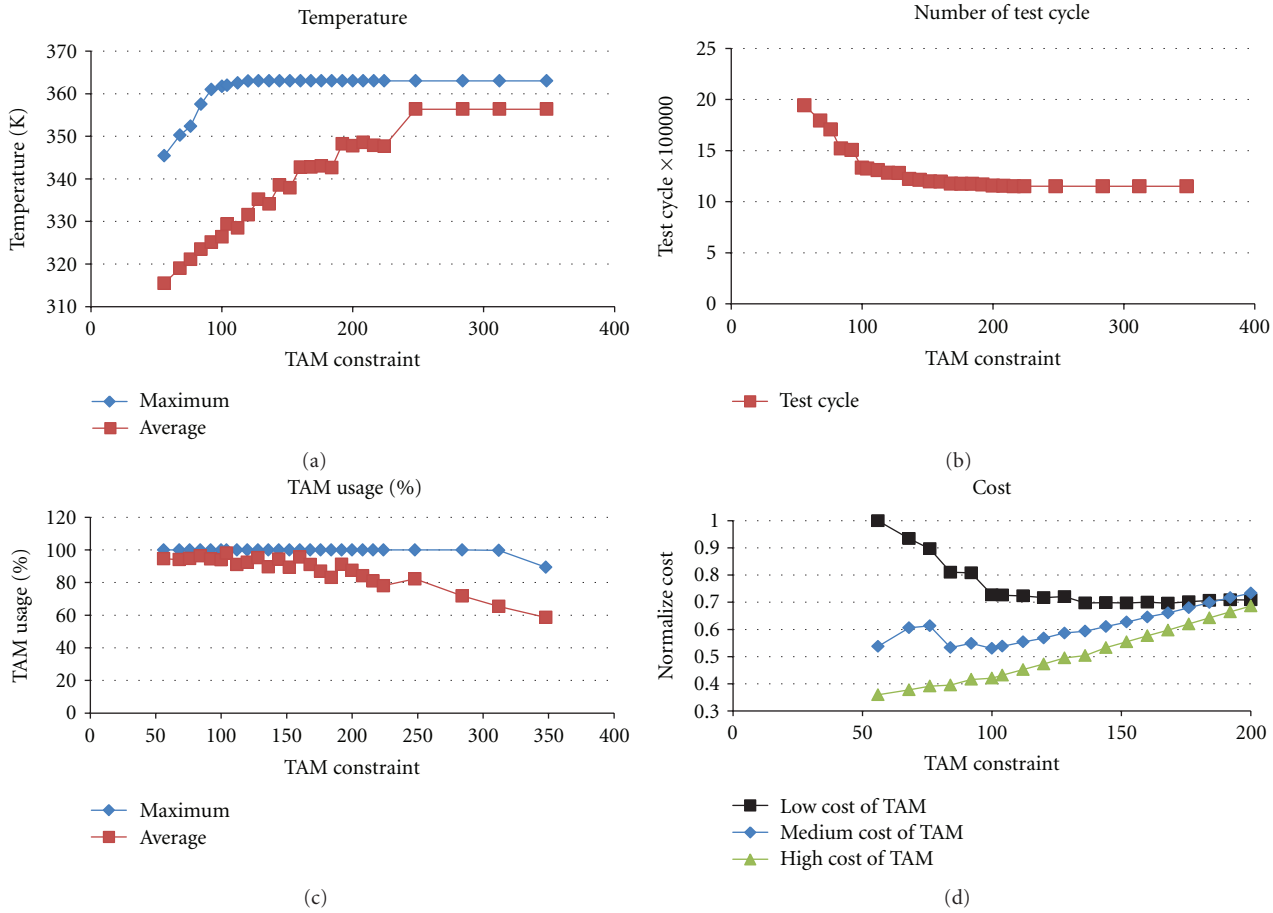

Figure 9: Results of case 2.

the number of scan chains, and the seventh column is the number of test patterns. The eighth column is the TAM width for each circuit, and the last column is test time (represented by the number of test cycles) for a single core. The ARM die is a real design and its test data is from the commercial ATPG tool. The logic cores are chosen from IWLS'05 benchmark circuits, whose test data is obtained by commercial tools. We assume memory BIST for memory cores. The test time for memories is calculated by the following equation, given by a TurboBIST Memory: $Test_{Time} = 13 \times Address \times pattern_{num}$, where *Address* is the address size of a single core, and *pattern_num* is the pattern number. Each core is identical in this test case.

Besides the first heterogeneous test case, we also hand-crafted two *homogeneous* test cases, which consist of pure logic circuits. We choose ten ITC'02 SOC benchmark [20] circuits in these two test cases. As most of the benchmark circuits have only test length information that can be found in the ITC benchmark website, we have to assume the other information: area, power, and floorplan. The area of each core is computed by the summation of input pins, output pins, and scan cells, multiplied by an area density, 3.18×10^{-4} (mm^2/number), which is obtained by average synthesis results of TSMC 180 nm technology. The test power is computed by the power density, 1.4 (W/mm^2), multiplied by the core area. The floorplan of each die is generated using the tool *HotFloorplan* [15].

Tables 2 and 3 show information of the second and third 3D IC test cases. The third column is the total area of each die. The last column shows the total test power consumption

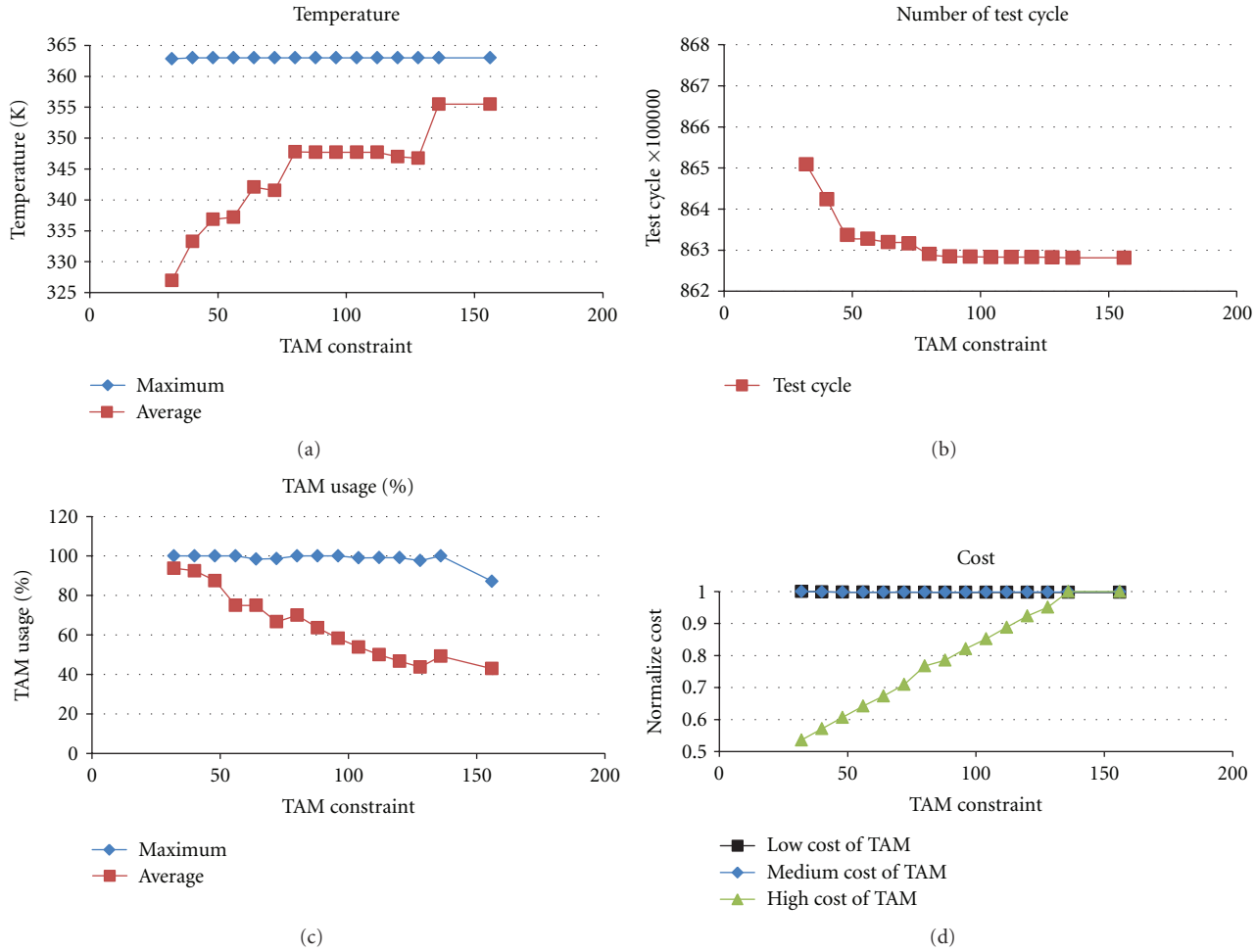

Figure 10: Results of case 3.

Table 2: The second test case.

Die	Circuit	Die area (mm^2)	Die power (W)
Die 4	p93791	7.30 × 4.21	42.97
Die 3	p22810	3.37 × 2.92	13.79
Die 2	p34392	2.54 × 2.86	10.16
Die 1	f2126	3.29 × 1.32	6.09
Die 0	d695	1.33 × 1.97	3.63

Table 3: The third test case.

Die	Circuit	Die area (mm^2)	Die power (W)
Die 4	t512505	4.92 × 4.95	34.08
Die 3	a586710	3.72 × 3.69	17.22
Die 2	q12710	2.99 × 2.79	11.65
Die 1	h953	1.09 × 1.61	2.46
Die 0	g1023	1.23 × 1.32	2.28

of each die. We stacked dies in increasing order of their die power, from the bottom up.

The ambient temperature is set to 25°C. The ambient resistance models the interface of 3D IC and the ambient environment. This parameter is decided by the package. In our experiment, we set it to 4°C/W, assuming a medium priced package. Users can change these parameters according to different situations. In addition, we have to add the temperature constraint to our optimization process. For our experiment, we set it to 90°C, which is the same as other papers in 2D IC test scheduling.

4.2. Results of Hard-Die Mode Optimization. Figure 8 shows the optimization results of case 1. In this case, when the coefficient ratio between α and β is medium that we choose 10 : 1, the optimal total TAM width is 52.

Figure 8(a) shows the maximum and average temperatures of the optimization test schedule. The temperature constraint T_{limit} is set to 90°C (363°K) in this experiment. We can see that temperature reaches its peak after $W_{\text{limit}} = 75$. The maximum number of total TAM allowed is 75.

Figure 8(b) shows the number of test cycles (test time) of our co-optimization results. When total TAM width is less than 75, this 3D IC test time is *TAM limited*. When total TAM width is larger than 75, this 3D IC test time is *temperature limited*. Adding more than 75 TAM widths does not reduce

Table 4: Results of soft-die mode.

Type		Case 1		Case 2		Case 3	
$\alpha:\beta$		1 : 1	10 : 1	1 : 1	10 : 1	1 : 1	10 : 1
Hard-die mode	No. of test cycle	1,823,944	882,200	1,216,486	1,181,825	86,509,578	86,509,576
	Opt. TAM	38	113	138	162	31	31
	Normalized test cost	1.00	1.00	1.00	1.00	1.00	1.00
Soft-die mode	No. of test cycle	1,823,944	882,200	766,191	611,629	23,142,087	41,852,141
	Opt. TAM	38	113	138	162	30	27
	Normalized test cost	1.00	1.00	0.69 (−31%)	0.59 (−41%)	0.54 (−46%)	0.80 (−20%)

test time. When TAM width is equal to 75, it becomes the boundary of these two stages.

Figure 8(c) shows the *TAM usage* versus the TAM constraint ($W_{\max}$). TAM usage is defined as the number of TAM used divided by $W_{\max}$. The maximum TAM curve (in diamond shape) shows the maximum TAM usage among all slots, while the average TAM curve (in square shape) shows the average TAM usage among all slots.

Figure 8(d) shows the optimal total cost of our co-optimization results (normalized to the maximum cost). Three different $\alpha:\beta$ ratios—10 : 1 (square curve), 1 : 1 (diamond curve), and 1 : 2 (triangle curve)—are shown for low, medium, and high TAM hardware cost, respectively. We see that the optimal test cost occurs at different TAM width for different $\alpha:\beta$ ratios. For $\alpha:\beta = 1:1$ (the diamond curve), the test cost is a convex curve and the optimal cost occurs when total TAM width is equal to 50. Adding more TAM width after 50 is a waste of hardware resources. This case shows that optimal TAM width is dependent on the relative cost of testing and silicon area.

Figure 9 shows the results for case 2. In this case, the temperature-limit and TAM-limit boundaries are around when TAM is equal to 100. The $\alpha:\beta$ ratios are the same as the experiment of case 1. Because this case has many small cores, TAM usage is quite high compared with case 1, which contains only few large cores. This case shows that, due to temperature limit, test time does not improve much by adding more TAM.

Figure 10 shows the results for case 3. In this case, the test power is very high, so it is always temperature-limited no matter how many TAM widths are added. The total test cost is dominated by the test time, so adding more TAM in this case just increases the total test cost with little improvement in test time. This case shows that heat dissipation is a key factor for testing some 3D ICs.

4.3. Result of Soft-Die Mode Optimization. In Table 4, it compares the results of soft-die and hard-die optimizations. There are two ratios ($\alpha:\beta = 1:1$ and 10 : 1) for three cases. In hard-die mode, we show the results of the optimal size of TAM. In soft-die mode, we further optimize the results of the hard-die mode. For case 1, there is no significant reduction in test cost after soft-die optimization. It is not easy to optimize the test cost by adjusting the total TAM width of any single core because the sizes of every core are approximately the same. For cases 2 and 3, we see significant improvements (20%∼46%) by reduction in test cost. The diversified cores are easier to adjust the test cost by simulated annealing.

Table 5: Temperature comparison for hard-die mode.

Test case	HotSpot (°K)	Proposed (°K)	Error
Case 1	362.10	362.68	0.16%
Case 2	354.98	356.86	0.52%
Case 3	361.47	362.89	0.39%

4.4. Accuracy Validation. To verify the accuracy of our thermal model, we use HotSpot to simulate our 3D IC and test schedules. In HotSpot simulation, we use exactly the same setup, such as core power, core location, thermal resistance of each die, and thermal resistance of ambient. Table 5 compares the maximum temperature of our thermal-resistance model and Hotspot simulation results. The second column is the maximum temperature simulated by HotSpot and the third column is the maximum temperature obtained by our proposed model. The last column shows the error between these two temperatures. The difference of temperature between our proposed thermal model and HotSpot is very small. The maximum error is just below 3%.

5. Conclusions

A thermal-aware test schedule and TAM co-optimization technique for 3D IC are proposed in this paper. Two optimization modes are supported: hard-die mode and soft-die mode. We use a simplified thermal-resistance model to quickly estimate the temperature of a test schedule without simulation.

The results show that thermal-aware co-optimization is important to decide the optimal TAM width and scheduling. The optimal TAM width and test scheduling are very design dependent. Blindly adding TAM width does not necessarily reduce test time due to the temperature constraint. Another important conclusion is that soft-die optimization greatly reduces the test time so DfT architecture of each core should be optimized together with the whole 3D IC.

Possible future work includes the consideration of pre-bond test, more sophisticated thermal models, and more realistic cost model.

References

[1] W. R. Davis, J. Wilson, S. Mick et al., "Demystifying 3D ICs: the pros and cons of going vertical," *IEEE Design and Test of Computers*, vol. 22, no. 6, pp. 498–510, 2005.

[2] E. J. Marinissen and Y. Zorian, "Testing 3D chips containing through-silicon vias," in *Proceedings of the International Test Conference (ITC '09)*, paper ET1.1, November 2009.

[3] R. S. Patti, "Three-dimensional integrated circuits and the future of system-on-chip designs," *Proceedings of the IEEE*, vol. 94, no. 6, pp. 1214–1224, 2006.

[4] S. Das, A. Chandrakasan, and R. Reif, "Timing, energy, and thermal performance of three-dimensional integrated circuits," in *Proceedings of the ACM Great lakes Symposium on VLSI (GLSVLSI '04)*, pp. 338–343, April 2004.

[5] S. Im and K. Banerjee, "Full chip thermal analysis of planar (2-D) and vertically integrated (3-D) high performance ICs," in *Proceedings of IEEE International Electron Devices Meeting (IEDM '00)*, pp. 727–730, December 2000.

[6] K. Puttaswamy and G. H. Loh, "Thermal analysis of a 3D die-stacked high-performance microprocessor," in *Proceedings of the ACM Great Lakes Symposium on VLSI (GLSVLSI '06)*, pp. 19–24, May 2006.

[7] C. Sun, L. Shang, and R. P. Dick, "Three-dimensional multiprocessor system-on-chip thermal optimization," in *Proceedings of the 5th International Conference on Hardware/Software Codesign and System Synthesis*, pp. 117–122, October 2007.

[8] K. Chakrabarty, "Test scheduling for core-based systems using mixed-integer linear programming," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 19, no. 10, pp. 1163–1174, 2000.

[9] V. Iyengar and K. Chakrabarty, "Precedence-based, preemptive, and power-constrained test scheduling for system-on-a-chip," in *Proceedings of the 19th IEEE VLSI Test Symposium (VTS' 01)*, pp. 368–374, May 2001.

[10] C. Liu, K. Veeraraghavan, and V. Iyengar, "Thermal-aware test scheduling and hot spot temperature minimization for core-based systems," in *Proceedings of the 20th IEEE International Symposium on Defect and Fault Tolerance in VLSI Systems (DFT '05)*, pp. 552–560, October 2005.

[11] T. E. Yu, T. Yoneda, K. Chakrabarty, and H. Fujiwara, "Thermal-safe test access mechanism and wrapper co-optimization for system-on-chip," in *Proceedings of the 16th Asian Test Symposium (ATS '07)*, pp. 187–192, October 2007.

[12] P. Rosinger, B. M. Al-Hashimi, and K. Chakrabarty, "Thermal-safe test scheduling for core-based system-on-chip integrated circuits," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 25, no. 11, pp. 2502–2511, 2006.

[13] C. Yao, K. K. Saluja, and P. Ramanathan, "Power and thermal constrained test scheduling under deep submicron technologies," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 30, no. 2, pp. 317–322, 2011.

[14] B. Noia, K. Chakrabarty, and E. J. Marinissen, "Optimization methods for post-bond die-internal/external testing in 3D stacked ICs," in *Proceedings of the 41st International Test Conference (ITC '10)*, pp. 1–10, November 2010.

[15] K. Skadron, M. R. Stan, K. Sankaranarayanan, W. Huang, S. Velusamy, and D. Tarjan, "Temperature-aware microarchitecture: modeling and implementation," *ACM Transaction on Architecture and Code Optimization*, vol. 1, no. 1, pp. 94–125, 2004.

[16] P. Girard, "Survey of low-power testing of VLSI circuits," *IEEE Design and Test of Computers*, vol. 19, no. 3, pp. 82–92, 2002.

[17] E. J. Marinissen, J. Verbree, and M. Konijnenburg, "A structured and scalable test access architecture for TSV-based 3D stacked ICs," in *Proceedings of the 28th IEEE VLSI Test Symposium (VTS '10)*, pp. 269–274, April 2010.

[18] M. Taouli, S. Hamdioui, K. Beenakker, and E. J. Marinissen, "Test impact on the overall die-to-wafer 3D stacked IC cost," *Journal of Electronic Testing*, vol. 28, pp. 15–25, 2011.

[19] Y. R. Huang, J. H. Pan, and Y. C. Lu, "Thermal-aware router-sharing architecture for 3D network-on-chip designs," in *Proceedings of the Asia Pacific Conference on Circuit and System (APCCAS '10)*, pp. 1087–1090, December 2010.

[20] E. J. Marinissen, V. Iyengar, and K. Chakrabarty, "ITC'02 SoC Test Benchmarks," http://www.extra.research.philips.com/itc02socbenchm.