# 3D Floorplanning with Thermal Vias

Eric Wong and Sung Kyu Lim
School of Electrical and Computer Engineering
Georgia Institute of Technology
ewong@ece.gatech.edu, limsk@ece.gatech.edu

***Abstract*— 3D circuits have the potential to improve performance over traditional 2D circuits by reducing wirelength and interconnect delay. One major problem with 3D circuits is that their higher device density due to reduced footprint area leads to greater temperatures. Thermal vias are a potential solution to this problem. This paper presents a thermal via insertion algorithm that can be used to plan thermal via locations during floorplanning. The thermal via insertion algorithm relies on a new thermal analyzer based on random walk techniques. Experimental results show that, in many cases, considering thermal vias during floorplanning stages can significantly reduce the temperature of a 3D circuit.**

## I. Introduction

Three dimensional (3D) integrated circuits are an emerging technology with great potential to improve performance. In a 3D integrated circuit, transistors may be fabricated on top of other transistors, resulting in multiple layers of active components. These transistors may then be wired to other transistors on the same device layer, to transistors on different device layers, or both, depending on the process technology. Several different approaches to fabricating 3D integrated circuits or 3D-compatible transistors have been taken [1] [2] [3] [4]. These techniques vary in terms of the maximum number of device layers and the maximum density of interconnects between these layers. The wafer-bonding approach in [4] joins discrete wafers using a copper interconnect interface, and permits multiple wafers and multiple 3D interconnects. Figure 1 shows the overall structure of a two die stack. The ability to route signals in the vertical dimension enables distant blocks to be placed on top of each other. This results in a decrease in the overall wire length, which translates into less wire delay, less power and greater performance. While the example in Figure 1 only shows two die bonded in a face to face organization, there are several candidate 3D technologies that include face to back bonding and stacking more than two die.

One of the biggest challenges of 3D circuit design is heat dissipation. In 3D circuits, more devices are packed into a smaller area, resulting in higher power densities. In addition, heat from the core of a 3D chip has to travel through layers of low conductivity dielectric, inserted between device layers, to reach a heat sink. One method for mitigating the thermal issue is to insert thermal vias [5], [6] that are used to establish thermal paths from the core of a chip to the outer layers as illustrated in Figure 2. Many existing works on thermal aware placement and routing for 2D circuits [7], [8], [9], [10] and 3D circuits [11], [12], [13], [16] do not consider thermal vias. Note that thermal vias are more useful for 3D circuits due to the multiple device layers. A few recent works have considered thermal vias in 3D circuits during routing [14], placement [15].

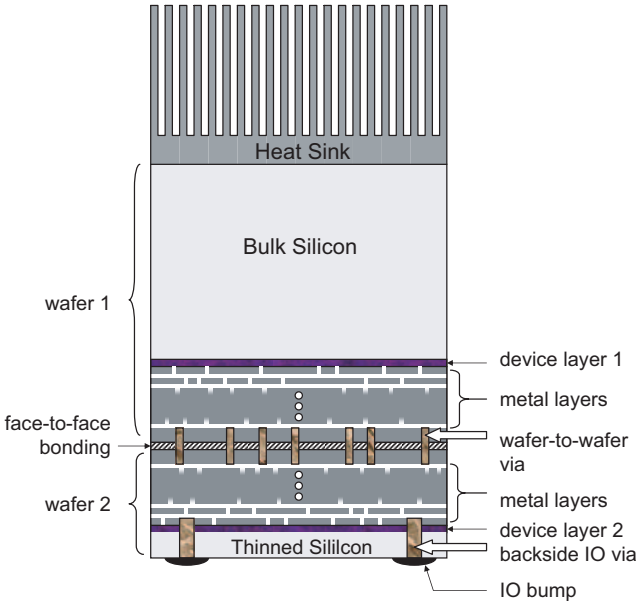


Fig. 1. A 3D chip with two die bonded face to face.

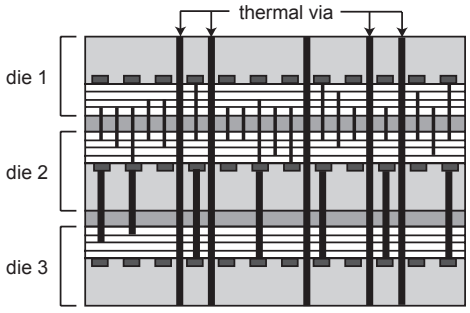


Fig. 2. Thermal vias in a 3D chip.

We believe that addressing thermal via planning even earlier as an integral part of floorplanning may allow greater flexibility in thermal management. Since the thermal analysis is a major runtime bottleneck during the thermal via aware 3D floorplanning optimization, we propose a fast thermal evaluator based on random walk techniques along with an efficient thermal via insertion algorithm. Our main contribution is the development of the first thermal via aware floorplanner that utilizes random walk based thermal analysis.

3-9810801-0-6/DATE06 © 2006 EDAA

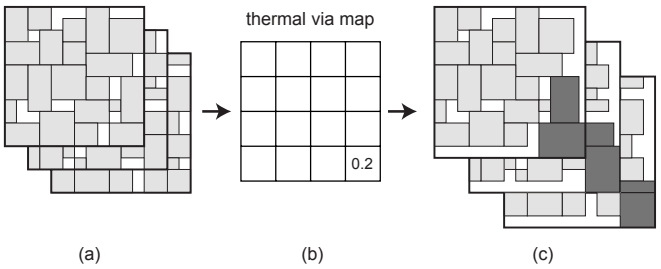


Fig. 3. Thermal via map and 3D area expansion. The area of each block located in the lower right corners of each layer is increased by an amount proportional to the thermal via density value of 0.2.

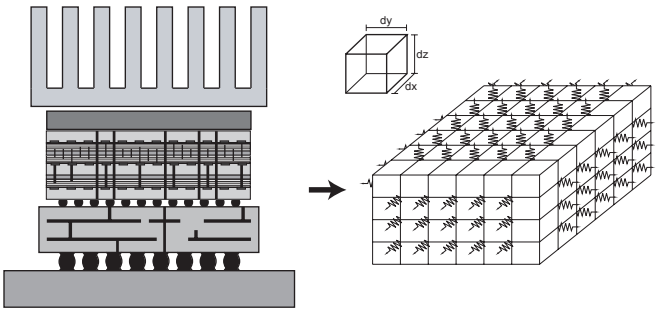


Fig. 4. Thermal grid for thermal modelling.

The organization of the paper is as follows. The problem formulation is presented in Section II. The random walk based thermal analyzer, thermal via insertion algorithm and 3D floorplanning algorithms are presented in Section III. Experimental results are provided in Section IV, and conclusions are in Section V.

## II. Problem Formulation

The following are given as the input to the Thermal Via Aware 3D Floorplanning Problem: (i) a set of blocks, (ii) the width, height, and average power density of each block, (iii) a netlist that specifies how the blocks are connected, and (iv) the number of device layers in the 3D structure. For each net $n$ from a given netlist, let $wl_n$ denote the wirelength of $n$, which is its manhattan distance. Let $A_{tot}$ denote the final footprint area of the 3D floorplan. Let $T_{max}$ denote the maximum temperature of the substrate. The goal of the Thermal Via Aware 3D Floorplanning Problem is to find the location of each block in the floorplan and to plan the locations of thermal vias, such that the following cost function is minimized:

$$w_1 \cdot A_{tot} + w_2 \cdot \sum_{n \in NL} wl_n + w_3 \cdot T_{max} \qquad (1)$$

where $w_1$, $w_2$, and $w_3$ are the weights of the objectives.

A 2D $m \times n$ *thermal via map* is used to represent the thermal via insertion solution. An illustration is shown in Figure 3. Each entry in the map represents the density of thermal vias inserted in each region of the floorplan. Since inter-layer thermal vias are inserted, the area of all vertically overlapping blocks need to be expanded according to the thermal via density to accommodate the thermal vias (see Figure 3). The area objective $A_{tot}$ in Equation (1) includes this extra area from thermal via insertion.

## III. Thermal Via-Aware 3D Floorplanning

### A. Overview of the Algorithm

Simulated annealing is a very popular approach for floorplanning due to its high quality solutions and flexibility in handling non-linear objectives. Sequence pair [17] is used to represent the floorplan. To extend sequence pair to 3D, one sequence pair was used for each layer. Simulated annealing begins with an initial floorplan and its cost in terms of area, wirelength, and maximum chip temperature. Then a random perturbation (move) is made to the initial solution to generate a new floorplan and its new cost is calculated. If the new cost is lower than the old one, then the new floorplan is accepted; otherwise there is a probability of acceptance based on the temperature of the annealing schedule. The higher the annealing temperature, the higher the acceptance probability. At each annealing temperature level a predetermined number of floorplans are examined. The annealing temperature is decreased exponentially, and the annealing process terminates when the freezing temperature is reached.

This paper explores three different ways to use thermal vias during floorplanning. The first approach is to use simulated annealing to optimize for area and wirelength by excluding chip temperature from the cost calculation, followed by inserting thermal vias at hotspots to decrease the maximum temperature. This will be referred to as *area/wirelength driven floorplanning* (AWF). The second approach is to optimize area, wirelength, and chip temperature simultaneously during simulated annealing, and further reduce temperatures by inserting thermal vias at the end. This will be referred to as *thermal driven floorplanning* (TDF). The third approach is to run the thermal via insertion algorithm each time the cost and chip temperature are calculated during simulated annealing so that the simulated annealer can take the effect of thermal vias into consideration. This will be referred to as *integrated thermal via floorplanning* (IVF).

### B. Thermal Model

A 3D thermal resistance mesh is used for thermal analysis. Each node models a small volume of the 3D die stack (substrate, heat sink, dielectric, metal, or transistor), and each edge denotes the thermal conductivity between two adjacent regions as shown in Figure 4. This is equivalent to using a discrete approximation of the steady state thermal equation $-k\nabla^2 T = P$, where $k$ is thermal conductivity, $T$ is temperature, and $P$ is power. This results in the matrix equation $G \cdot t = p$, where $G$ is a thermal conductivity matrix, $p$ is a power vector, and $t$ is a temperature vector. One way to solve this matrix equation would be to invert the matrix $G^{-1} = R$, which takes $O(n^3)$ time. Then $t$ can be calculated through matrix multiplication $t = R \cdot p$, which takes $O(n^2)$ time.

During thermal driven floorplanning, moving blocks around does not significantly change the thermal conductivities. The

power profile changes are mainly responsible for the changes in temperature. This allows the $G$ matrix to be inverted to $R$ once in the beginning and reused for subsequent temperature calculations. Only the power vector needs to be changed, so temperature calculations only require one matrix multiplication. This allows the temperature of each floorplan to be evaluated in $O(n^2)$ time rather than $O(n^3)$ time. This method of reusing $R$ is slightly inaccurate due to the fact that the area of the floorplan will change, which causes slight changes in thermal conductance between thermal grid cells. When inserting thermal vias, however, thermal conductivities change. This means that $R$ cannot be reused, so directly solving the matrix equation would take $O(n^3)$. This is much too slow for use in integrated thermal via floorplanning. To solve this problem we propose another method for calculating temperature.

### C. Random Walk-based Thermal Analysis

Random walks correspond to a classical problem in statistics, and their use in solving linear equations dates back to as early as the 1950s [18] [19] [20]. Recently, Qian et al. [21] [22] applied the random walk concept to power grid analysis. In a random walk game, a walker starts at a node in a graph with a certain amount of money. The walker then randomly visits a neighboring node. The probability of each neighbor being visited is based on the weight of its edge to the current node. At each node, the walker either receives a reward or pays a toll. The walk ends when the walker reaches a home node and the walker will have made or lost some money based on the tolls paid and rewards collected.

The temperature of a thermal grid cell is calculated by placing a walker with no money at the cell. First, the walker will receive a reward of

$$r(i) = \frac{p_i}{\sum_j^{d(i)} g_{ij}} \tag{2}$$

where $p_i$ is the power of the current cell $i$, $d(i)$ is the edge degree of cell $i$, and $g_{ij}$ is the thermal conductance between cell $i$ and its neighbor $j$. The walker will then visit one of its six neighboring cells. The probability of each neighbor $j$ being chosen from cell $i$ is

$$p(i,j) = \frac{g_{ij}}{\sum_j^{d(i)} g_{ij}}. \tag{3}$$

At each step, the walker will receive a reward and visit another neighbor. The walk ends when the walker hits a boundary cell at this point the walker will receive the final reward $r$ = ambient temperature. The total amount of money collected by the walker is an approximation of the temperature of the cell that the walker started from. According to the Central Limit Theorem, if many walks are performed and the results are averaged, then the error is a zero mean Gaussian variable with a variance inversely proportional to the number of walks $k$. This gives a tradeoff between runtime and accuracy. The runtime of the random walk is $O(kmn)$, where $k$ is the number of walks per cell, $m$ is the average length of a walk, and $n$ is the number of cells. Typically, $k$ and $m$ are much smaller than $n$, so a random walk will run much faster than solving the matrix equation $G \cdot t = p$ with a runtime of $O(n^3)$.

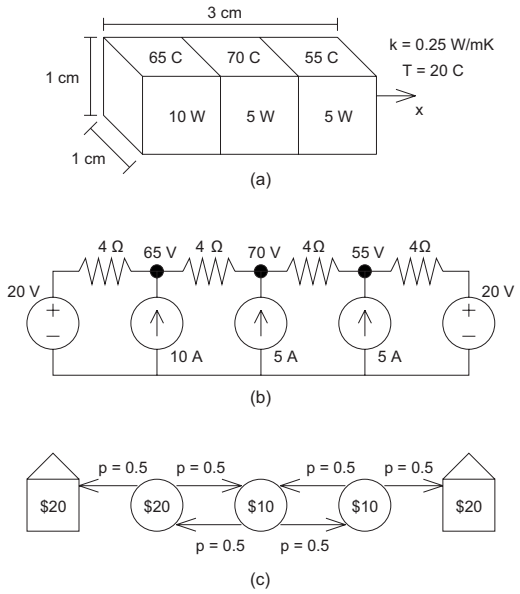


Fig. 5. (a) A simple 1-dimensional thermal problem (heat only flows along the x-axis). The bar is divided into three sections with different power dissipations. (b) The electrical analog of the thermal problem. (c) The random walk game corresponding to the thermal problem.

Several techniques can be used to speed up the random walk-based thermal analysis. It is possible for a random walk to wander around inside the thermal grid and not reach a boundary cell for an extremely long time. To combat this problem, a limit on the path length of a random walk $m_{max}$ is imposed. If $m_{max}$ is set too low, then many random walks will be cut short. Losing too many long walks will tend to cause the calculated temperatures to be low. When $m_{max}$ is set high enough, few random walks will be affected and the underestimation becomes negligible. The next speed up technique is to create new home cells. When the temperature of a cell is calculated, it becomes an additional home cell with a reward equal to its temperature for subsequent random walks starting elsewhere. The new homes cut down on the average length of walks significantly. The temperatures of individual cells can be calculated without having to solve the entire thermal grid, which is done by performing random walks starting from the cells of interest and not performing random walks starting elsewhere. This is especially useful for thermal via insertion since this allows the local impact of thermal vias on a target hot-spot to be calculated without recalculating the entire temperature profile.

Figure 5(a) shows a simple 1-dimensional thermal problem, where heat only flows along the x-axis. The bar has three different regions which have different power dissipations. The ambient temperature around the ends of the bar is $20^\circ$ C. Figure 5(b) is the electrical analog of the thermal problem. Temperatures become voltages and and ambient temperature becomes ground. Power dissipations become electrical current.

TABLE I

THERMAL ANALYZER RESULTS FOR A SAMPLE FLOORPLAN

| | Matrix | Fast Matrix | Random Walk |
|---|---|---|---|
| max temp | 82.6 | 82.4 | 83.2 |
| rms error | - | 0.2 | 3.6 |
| relative runtime | 463 | 1 | 7 |

Thermal conductivities are converted into electrical resistances. Figure 5(c) is the random walk game that corresponds with the thermal problem. For example, to calculate the temperature of the left cell of the bar the walker would start at the left circle in the random walk game. There he would receive a reward of \$20. The walker might move left to the home and receive another \$20 for a total of \$40. That walk would estimate the temperature to be $40^\circ$ C. Another possible walk would also receive \$20 from the starting point. The first move might be to the right, receiving \$10. The next move might be back to the left, receiving \$20. The final move might be left to the home for another \$20. This would give an estimated temperature of $70^\circ$ C. Averaging the results from the two random walks gives a temperature of $55^\circ$ C. Performing additional walks and averaging the results would bring the estimated temperature closer to the true temperature of $65^\circ$ C.

Table I compares the three methods of calculating the temperature of a sample floorplan. The matrix thermal analyzer solves $G \cdot t = p$ and is used as the baseline. The fast matrix thermal analyzer reuses $R = G^{-1}$ calculated previously. The random walk thermal analyzer shown did 100 random walks per cell. The fast matrix thermal analyzer is both faster and more accurate than the random walk thermal analyzer. This makes it a better choice for thermal-driven floorplanning. The fast matrix thermal analyzer cannot be used when thermal conductivities change, so the random walk thermal analyzer is used for via insertion.

### D. Thermal Via Insertion Algorithm

An iterative method is used for thermal via insertion. First the thermal grid cell with the highest temperature is found. Then the target thermal conductivity of the cell is calculated according to the formula

$$k_{new} = k_{old} \cdot \frac{t_{curr}}{t_{target}} \tag{4}$$

where $k_{old}$ is the current thermal conductivity of the cell, $t_{curr}$ is the current temperature of the cell, and $t_{target}$ is the target temperature. The via density of the x-y location of the cell is calculated with the formula

$$v = \min\left(v_{max},\ c \cdot \frac{k_{new} - k_{old}}{k_{via} - k_{old}}\right) \tag{5}$$

where $v_{max}$ is the maximum thermal via density, $c$ is a user defined constant, and $k_{via}$ is the thermal conductivity of a thermal via. Next, the thermal conductivities are updated according to the thermal via density. Random walk is used to calculate the temperature of the cells that the new vias pass through as well as the temperature of adjacent cells. Then another grid cell with the highest temperature is found and the process repeats. This process is iterated until the maximum temperature is less than the target temperature or when the maximum number of iterations has been reached.

After thermal via insertion, blocks that occupy areas with thermal vias need to be expanded to make room for the vias. The average via density of a block is the amount that it will expand by. Next, a sequence pair floorplan compaction calculation is used to update the location of the expanded blocks. With updated block sizes and locations, a final temperature calculation can be performed. If the via insertion is integrated into the floorplanning, then the random walk thermal analyzer is used for the temperature calculation. If the via insertion is done as a postprocess, then the temperature is calculated with the matrix thermal analyzer. The result of the thermal via inserter is a thermal via density map introduced in Section II. Thermal vias can then be placed according to the thermal via density map, where they will be fixed obstacles during the placement phase of physical design.

### E. Integrated Floorplanning with Thermal Vias

The floorplanner is based on simulated annealing. An array of sequence pairs was used to represent to solution space, with one sequence pair per layer. Each move is made by modifying the sequence pair. Then, the area of the floorplan and the location of the blocks is calculated from the sequence pair using an algorithm based on longest common subsequence [23]. The wirelength of a net is estimated by drawing a bounding box around the blocks connected by the net and taking the half perimeter of the bounding box. The temperature before thermal via insertion is calculated using the fast matrix thermal analyzer. Then thermal vias are inserted. The random walk based thermal analyzer is used to calculate the temperature after thermal via insertion. Then a weighted average of the area, wirelength and temperature *after* thermal via insertion is used as the cost function for the simulated annealer for integrated thermal via floorplanning.

In area and wirelength driven floorplanning, the cost function is a weighted average of the area and wirelength. In thermal driven floorplanning the cost function is a weighted average of the area, wirelength, and temperature *without* vias. When thermal vias are inserted for the final floorplan, the matrix thermal analyzer is used to calculate the temperature before and after thermal via insertion to ensure accurate final results.

## IV. EXPERIMENTAL RESULTS

The algorithms in the paper were implemented in C++. The experiments were run on Pentium IV 2.4 Ghz dual processor systems running linux. Ten GSRC benchmarks were used. The blocks were randomly assigned power densities between $10^6\ W/m^2$ and $5 \cdot 10^6\ W/m^2$. All floorplans have four placement layers.

Table II shows the results of the AWF algorithm (area and wirelength driven floorplanning) with thermal via insertion as a postprocess. The eighth column shows what the temperature

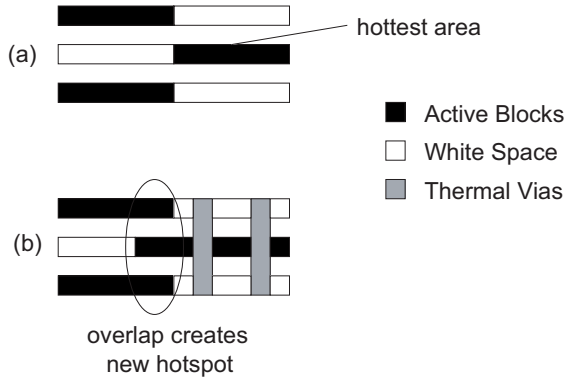


Fig. 6. A three layer floorplan before thermal via insertion (a) and after thermal via insertion (b). In this case, adding thermal vias expanded the block in the middle layer and created a new hotspot.

of the floorplan would be if the floorplan were expanded but thermal vias were not added. Floorplan expansion was responsible for a temperature drop of approximately 4%, while the increase in thermal conductivity due to thermal vias was responsible for an additional 13% temperature drop. Average thermal via density is the proportion of the area reserved for thermal vias. Note that this is not necessarily equal to the area expansion because the expansion of individual blocks is not uniform. An average thermal via density of under 3% was able to decrease temperatures by almost 17% while expanding the area by less than 4% and increasing wirelength by only 1%.

Table III shows the results of the TDF algorithm (thermal driven floorplanning) followed by thermal via insertion. In half the cases, adding thermal vias actually increased the temperature. The temperature without thermal via column suggests the reason for this. The TDF tends to separate high power blocks. The area expansion of blocks due to thermal vias can cause the blocks to shift enough to bring some high power density blocks closer together, which can increase temperature. Figure 6 shows an example of this effect. The increased thermal conductivity from the thermal vias can sometimes make up for this effect, but often it cannot. TDF without thermal vias is more effective at reducing temperatures than AWF followed by thermal via insertion. However, TDF has higher area due to looser module packing.

Table IV shows the results of IVF algorithm (integrated thermal via floorplanning). IVF solved the problem that TDF had with thermal vias by being aware of thermal vias throughout floorplanning. This allowed it to produce the lowest temperatures out of the three methods. Table V displays the tradeoff between area, wirelength and temperature in the three approaches to floorplanning. Adding thermal vias to AWF reduced the temperature by 17% at a cost of 4% area expansion and 1% wirelength increase. TDF without thermal vias reduced temperature by 32% at a cost of 20% area increase and 5% wirelength increase. Finally, IVF reduced temperature by 38% at a cost of 47% area increase and 22% wirelength increase. The thermal via density of IVF averages 2.5%, so most of the area increase came from loose module packing.

## V. Conclusions

A fast approximation algorithm for thermal analysis was presented in this paper. This thermal analyzer was incorporated into an efficient thermal via insertion algorithm. The thermal via inserter successfully lowered temperatures with minimal thermal via densities. Integrating thermal via insertion into the floorplanner resulted in lower temperatures than inserting vias as a postprocess.

## References

[1] C. W. Eichelberger, "Three-dimensional multichip module system," 1992, u.S. Patent 5 111 278.

[2] G. Roos, B. Hoefflinger, M. Schubert, and R. Zingg, "Manufacturability of 3D-epitaxial-lateral-overgrowth CMOS circuits with three stacked channels," *Microelectron*, 1991.

[3] V. Subramanian, P. Dankoski, L. Degertekin, B. Khuri-Yakub, and K. Saraswat, "Controlled two-step solid-phase crystallization for high-performance polysilicon TFTs," *IEEE Electron Device Letter*, 1997.

[4] A. Fan, A. Rahman, and R. Reif, "Copper wafer bonding," *Electrochemical Solid-State Letter*, 1999.

[5] T. Chiang, K. Banerjee, and K. Saraswat, "Effect of Via Separation and Low-k Dielectric Materials on the Thermal Characteristics of Cu Interconnects," in *IEEE International Electron Devices Meeting*, 2000.

[6] D. Pinjala, M. Iyer, C. Guan, and I. Rasiah, "Thermal characterization of vias using compact models," in *Proc. of the Electronics Packaging Technology Conf.*, 2000.

[7] C. N. Chu and D. F. Wong, "A matrix synthesis approach to thermal placement," *IEEE Trans. on Computer-Aided Design of Integrated Circuits and Systems*, 1998.

[8] C. Tsai and S. Kang, "Cell-level placement for improving substrate thermal distribution," *IEEE Trans. on Computer-Aided Design of Integrated Circuits and Systems*, 2000.

[9] G. Chen and S. Sapatnekar, "Partition-driven standard cell thermal placement," in *Proc. Int. Symp. on Physical Design*, 2003.

[10] W. Hung, Y. Xie, N. Vijaykrishnan, C. Addo-Quaye, T. Theocharides, and M. Irwin, "Thermal-aware floorplanning using genetic algorithms," in *Proc. Int. Symp. on Quality Electronic Design*, 2005.

[11] S. Das, A. Chandrakasan, and R. Reif, "Timing, energy, and thermal performance of three-dimensional integrated circuits," in *Proc. Great Lakes Symposum on VLSI*, 2004.

[12] K. Balakrishnan, V. Nanda, S. Easwar, and S. K. Lim, "Wire congestion and thermal aware 3d global placement," in *Proc. Asia and South Pacific Design Automation Conf.*, 2005.

[13] J. Minz, E. Wong, and S. K. Lim, "Thermal and Power Integrity-aware Floorplanning for 3D Circuits," in *Proc. IEEE Int. SOC Conf.*, 2005.

[14] J. Cong and Y. Zhang, "Thermal-driven multilevel routing for 3-D ICs," in *Proc. Asia and South Pacific Design Automation Conf.*, 2005.

[15] B. Goplen and S. Sapatnekar, "Thermal Via Placement in 3-D ICs," in *Proc. Int. Symp. on Physical Design*, 2005.

[16] J. Cong, J. Wei, and Y. Zhang, "A thermal-driven floorplanning algorithm for 3D ICs," in *Proc. IEEE Int. Conf. on Computer-Aided Design*, 2004.

[17] H. Murata, K. Fujiyoshi, S. Nakatake, and Y. Kajitani, "Rectangle packing based module placement," in *Proc. IEEE Int. Conf. on Computer-Aided Design*, 1995, pp. 472–479.

[18] G. Forsythe and R. Leibler, "Matrix inversion by a monte carlo method," *Mathematical Tables and Other Aids to Computation*, 1950.

[19] W. Wasow, "A note on the inversion of matrices by random walks," *Mathematical Tables and Other Aids to Computation*, 1952.

[20] P. Doyle and J. Snell, "Random walks and electric networks," *Mathematical Association of America*, 1984.

[21] Q. H. Qian, S. Nassif, and S. Sapatnekar, "Random walks in a supply network," in *Proc. ACM Design Automation Conf.*, 2003.

[22] Q. H. Qian and S. Sapatnekar, "Hierarchical random-walk algorithms for power grid analysis," in *Proc. Asia and South Pacific Design Automation Conf.*, 2004.

[23] X. Tang, R. Tian, and D. F. Wong, "Fast evaluation of sequence pair in block placement by longest common subsequence computation," *IEEE Trans. on Computer-Aided Design of Integrated Circuits and Systems*, 2001.

TABLE II

AREA AND WIRELENGTH DRIVEN FLOORPLANNING WITH THERMAL VIA INSERTION AS A POSTPROCESS.

| | before via insertion | | | after area expansion | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| benchmarks | area | wirelength | temp | area | wirelength | temp w/ vias | temp w/o vias | average via density | time |
| n50 | 58491 | 91521 | 136.6 | 59309 | 91986 | 126.5 | 132.4 | 0.011 | 121 |
| n50b | 66490 | 87838 | 145.1 | 72564 | 90886 | 115.7 | 145.0 | 0.065 | 110 |
| n50c | 63666 | 92418 | 129.2 | 64251 | 92900 | 122.1 | 125.6 | 0.008 | 111 |
| n100 | 57664 | 135970 | 123.6 | 61431 | 138729 | 92.3 | 113.6 | 0.046 | 193 |
| n100b | 49950 | 120431 | 112.9 | 51095 | 121297 | 98.0 | 112.2 | 0.013 | 408 |
| n100c | 53040 | 132142 | 128.8 | 54135 | 133800 | 95.4 | 117.6 | 0.027 | 251 |
| n200 | 50190 | 215549 | 135.6 | 52472 | 218601 | 105.2 | 131.9 | 0.036 | 1407 |
| n200b | 55385 | 226447 | 125.9 | 57579 | 228792 | 103.7 | 123.6 | 0.031 | 887 |
| n200c | 52877 | 250970 | 123.6 | 53601 | 251855 | 110.8 | 120.8 | 0.011 | 643 |
| n300 | 81340 | 313680 | 186.9 | 83801 | 316041 | 146.9 | 174.1 | 0.026 | 4395 |
| RATIO | 1.000 | 1.000 | 1.000 | 1.035 | 1.012 | 0.831 | 0.963 | - | - |

TABLE III

THERMAL DRIVEN FLOORPLANNING WITH THERMAL VIA INSERTION AS A POSTPROCESS.

| | before via insertion | | | after area expansion | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| benchmarks | area | wirelength | temp | area | wirelength | temp w/ vias | temp w/o vias | average via density | time |
| n50 | 62517 | 91363 | 120.7 | 66393 | 93772 | 98.5 | 118.3 | 0.057 | 852 |
| n50b | 68694 | 85173 | 118.1 | 71571 | 86781 | 130.8 | 144.2 | 0.042 | 1307 |
| n50c | 64532 | 91808 | 110.6 | 66912 | 93797 | 105.3 | 113.8 | 0.051 | 1106 |
| n100 | 68480 | 142521 | 82.0 | 69541 | 143942 | 90.2 | 93.4 | 0.028 | 1708 |
| n100b | 61490 | 127801 | 84.8 | 63066 | 129821 | 81.8 | 96.2 | 0.052 | 2175 |
| n100c | 63745 | 138324 | 85.7 | 65655 | 139790 | 91.3 | 105.6 | 0.032 | 1425 |
| n200 | 62220 | 270123 | 97.9 | 63406 | 271742 | 91.0 | 97.9 | 0.019 | 3389 |
| n200b | 70596 | 250672 | 69.1 | 72039 | 253473 | 105.1 | 109.7 | 0.051 | 5509 |
| n200c | 66150 | 250582 | 74.4 | 67874 | 252339 | 69.7 | 78.2 | 0.032 | 6202 |
| n300 | 117600 | 334304 | 51.4 | 118397 | 335528 | 67.5 | 67.6 | 0.046 | 17659 |
| RATIO | 1.000 | 1.000 | 1.000 | 1.028 | 1.013 | 1.071 | 1.169 | - | - |

TABLE IV

INTEGRATED THERMAL VIA FLOORPLANNING

| benchmarks | area | wirelength | temp | average via density | time |
|---|---|---|---|---|---|
| n50 | 86093 | 102425 | 94.1 | 0.035 | 9175 |
| n50b | 82925 | 94088 | 108.6 | 0.017 | 13107 |
| n50c | 80303 | 100013 | 86.8 | 0.050 | 8979 |
| n100 | 83311 | 155972 | 87.7 | 0.007 | 16110 |
| n100b | 81893 | 148806 | 71.7 | 0.022 | 23514 |
| n100c | 81596 | 152045 | 76.4 | 0.020 | 24524 |
| n200 | 74414 | 310017 | 75.8 | 0.029 | 29417 |
| n200b | 82599 | 284590 | 98.7 | 0.022 | 25140 |
| n200c | 77465 | 304035 | 68.7 | 0.022 | 32053 |
| n300 | 136907 | 468086 | 56.4 | 0.027 | 47337 |

TABLE V

COMPARISON OF THERMAL-DRIVEN AND INTEGRATED THERMAL VIA FLOORPLANNING

| | area/wirelength driven | | | area/wirelength w/vias | | | thermal driven w/o vias | | | integrated thermal via | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| benchmarks | area | wirelength | temp | area | wirelength | temp | area | wirelength | temp | area | wirelength | temp |
| n50 | 58491 | 91521 | 136.6 | 59309 | 91986 | 126.5 | 62517 | 91363 | 120.7 | 86093 | 102425 | 94.1 |
| n50b | 66490 | 87838 | 145.1 | 72564 | 90886 | 115.7 | 68694 | 85173 | 118.1 | 82925 | 94088 | 108.6 |
| n50c | 63666 | 92418 | 129.2 | 64251 | 92900 | 122.1 | 64532 | 91808 | 110.6 | 80303 | 100013 | 86.8 |
| n100 | 57664 | 135970 | 123.6 | 61431 | 138729 | 92.3 | 68480 | 142521 | 82.0 | 83311 | 155972 | 87.7 |
| n100b | 49950 | 120431 | 112.9 | 51095 | 121297 | 98.0 | 61490 | 127801 | 84.8 | 81893 | 148806 | 71.7 |
| n100c | 53040 | 132142 | 128.8 | 54135 | 133800 | 95.4 | 63745 | 138324 | 85.7 | 81596 | 152045 | 76.4 |
| n200 | 50190 | 215549 | 135.6 | 52472 | 218601 | 105.2 | 62220 | 270123 | 97.9 | 74414 | 310017 | 75.8 |
| n200b | 55385 | 226447 | 125.9 | 57579 | 228792 | 103.7 | 70596 | 250672 | 69.1 | 82599 | 284590 | 98.7 |
| n200c | 52877 | 250970 | 123.6 | 53601 | 251855 | 110.8 | 66150 | 250582 | 74.4 | 77465 | 304035 | 68.7 |
| n300 | 81340 | 313680 | 186.9 | 83801 | 316041 | 146.9 | 117600 | 334304 | 51.4 | 136907 | 468086 | 56.4 |
| RATIO | 1.000 | 1.000 | 1.000 | 1.035 | 1.012 | 0.831 | 1.195 | 1.054 | 0.678 | 1.473 | 1.220 | 0.625 |