

A Combination of Evolutionary Algorithm and Mathematical Programming for the 3D Thermal-Aware Floorplanning Problem

David Cuesta, José L. Risco-Martín, José L. Ayala, J. Ignacio Hidalgo

‡ Dept. of Computer Architecture and Automation, Universidad Complutense de Madrid, 28040 Madrid, Spain
C/ Prof. José García Santesmases, s/n, 28040 Madrid, Spain
dcuestag@pdi.ucm.es, jlrisco@dacya.ucm.es, {jayala, hidalgo}@fdi.ucm.es

ABSTRACT

¹ Heat removal and power density distribution delivery have become two major reliability concerns in 3D stacked technology. Additionally, the placement of *Through-Silicon-Vias* (TSVs) for connecting different layers is one of the key issues in 3D technology. Although a few recent works have considered thermal-aware placement of cores in chip multi-processor architectures, the concepts of 3D and TSVs have not been conveniently incorporated. Therefore, new suitable exploration methods for the 3D thermal-aware floorplanning problem need to be developed. In this paper we analyze the benefits of two different exploration techniques for the floorplanning problem: *Multi-Objective Genetic Algorithm* (MOGA) and a *Mixed Integer Linear Program* (MILP). We present a novel algorithm that uses MILP to minimize average temperature in the 3D chip, whereas uses MOGA to insert TSVs, connecting the layers while the total wire length is minimized. Our experiments with two different 3D chips show that our algorithm achieves 10% reduction in the maximum temperature and thermal gradient.

Categories and Subject Descriptors

C.3 [Special-Purpose and Application-Based Systems]: Real-time and embedded systems
; I.2.8 [Artificial Intelligence]: Problem Solving, Control Methods and Search—*Heuristic methods*

General Terms

Design, Performance

¹This work has been supported by Spanish Government grants TIN2008-00508 and MEC Consolider Ingenio CSD00C-07-20811 of the Spanish Council of Science and Technology, and partially supported by the Swiss National Science Foundation (SNF) grant 200021-127282.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

GECCO'11, July 12–16, 2011, Dublin, Ireland.

Copyright 2011 ACM 978-1-4503-0557-0/11/07 ...\$10.00.

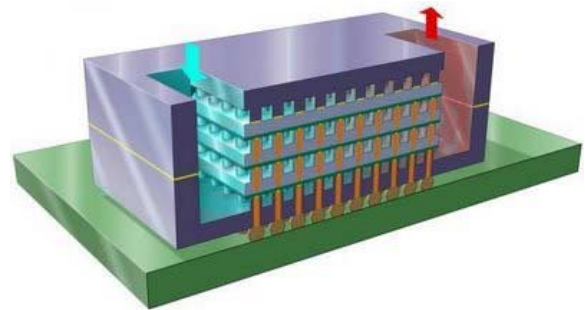


Figure 1: IBM 3D technology

Keywords

Multi-Objective Optimization, Evolutionary Computation, Genetic Algorithms, Mathematical Programming, Floorplanning, Thermal Aware Design

1. INTRODUCTION AND RELATED WORK

Three dimensional (3D) integrated circuits are an emerging technology with great potential to improve performance [17]. In a *3D Integrated Circuit* (3D IC), transistors may be fabricated on top of other transistors, resulting in multiple layers of active components. This can be translated into a reduction of wire length allviating interconexion delay and reducing chip area.

One of the biggest challenges of 3D circuit design is heat dissipation [19]. In 3D circuits, more devices are packed into a smaller area, resulting in higher power densities that are difficult to cool down. These devices may then be wired to other devices in the same layer, to devices in different layers, or both, depending on the process technology. The connection between devices in different layers is addressed via *Through-Silicon-Vias* (TSVs) insertion [13], [2]. Figure 1 shows the schematic view of 3D chip technology proposed by IBM. As can be seen, multiple layers are stacked and communicated through vertical paths (TSVs). Also, IBM proposes complex cooling mechanisms based on circulation of a liquid coolant to reduce the high temperatures in the inner layers.

Many existing works on thermal aware placement and routing for 2D circuits [6], [21], [5], [15] and 3D circuits,

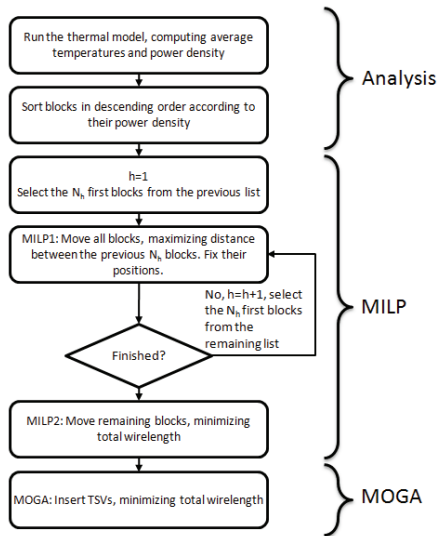


Figure 2: Iterative flow of our approach

[3], [7] do not consider TSVs. A few recent works have considered thermal vias in 3D circuits during routing [18], and placement [22].

Most of the algorithms presented for the 3D thermal aware floorplanning problem are based on a Mixed Integer Linear Program (MILP) [14, 16], Simulated Annealing (SA) [7, 14] or Genetic Algorithm (GA) [20]. MILP has proven to be an efficient solution. However, when MILP is used for thermal aware floorplanning, the (linear) thermal model must be added to the topological relations and the resultant algorithm becomes too complex [10]. Regarding SA and GA, both are based on the representation of the solution and only a few of them include TSVs insertion. Some common representations are polish notation [4], combined bucket array [7] and O-tree [20]. Most of these representations do not perform well, because they were initially used to reduce area. In the thermal aware floorplanning problem, hottest elements must be placed as far as possible in the 3D IC.

In this work, we take advantage of MILP and meta-heuristics approaches to design a novel combination of both algorithms that solves the placement and routing problem for the insertion of TSVs in the 3D Thermal-aware floorplanning problem. We formulate a multi-objective optimization since there are two conflicting objectives that will be minimized. These two objectives are the mean temperature and the total wire length. We also take care of the number of TSVs inserted. MILP is used for the placement of functional units, using an incremental floorplanning to avoid hotspots. A *Multi-Objective Genetic Algorithm (MOGA)* is used for the placement of the TSVs.

2. DESIGN FLOW

Our main objective is two fold: (1) to place the functional units of the 3D IC reducing the average on-chip temperature and (2) to insert TSVs connecting different layers satisfying the design constraints. To this end, we propose a novel 3D thermal-aware floorplanner, that includes three different phases, depicted in Figure 2.

As Figure 2 shows, the first phase (*Analysis*) performs an

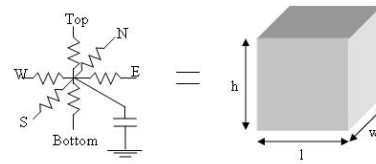


Figure 3: Equivalent RC circuit of a single cell

accurate analysis of the 3D IC thermal behavior, calculating the temperature of each functional unit of the architecture. An initial scenario for our work will be the Niagara architecture [1]. Otherwise, we can randomly create one composed by the selected number of functional units, and then run the thermal simulator.

The second phase (*MILP* in Figure 2) performs two tasks. The first one (MILP1) moves all the blocks until the hottest ones are placed cautiously, trying to set them as far as possible from each other. Then, the second one is in charge of placing the remaining blocks minimizing wire length.

Finally, the third phase (*MOGA* in Figure 2), inserts the TSVs minimizing also wire length in the 3D IC. MOGA gives a Pareto front approximation representing the number of TSVs vs. the total wire length. Since the number of TSVs usually is not large, we decided to provide the whole front, being a designer's responsibility to choose the appropriate range. We describe these three phases in more detail in following paragraphs.

2.1 Thermal analysis

As Figure 2 shows, we first perform a thermal analysis of the chip components. To this end, we have developed an accurate thermal model, which is briefly described in the following paragraphs.

3D integration consists of placing different active layers using silicon dioxide and joining them with a glue material. If inter layer communication is required, Trough Silicon Vias (TSVs) allow this functionality.

Some of the goals of 3D stacks are to achieve a reduction in area and also to decrease the length of the interconnections, that would be translated into a decrease in the data transfer time and the power consumption.

The 3D stack is built over an adiabatic PCB surface and then, traditional technological dies composed of silicon dioxide and silicon, are placed one over the others. The 3D stacks that will be studied in the experimental work are composed of 4 and 5 layers, as will be shown in section 3.

The heat flow through the 3D stack is basically diffusive, hence, it can be characterized with a 3D RC thermal model as the one presented in [2].

The first phase of the model splits the chip into small cubic unitary cells. These cells are modeled with six thermal resistances and one thermal capacitance as can be seen in Figure 3. Four resistances, the ones in the same plane, connect each cell to its lateral neighbors. The other two resistances connect the cell with the upper and bottom cell respectively. The capacitance represents self heat storage.

The model also considers the heat diffusion to the surrounding environment. It is possible to simulate different chip packages by tuning the resistance and conductance parameters. The PCB base is adiabatic, so no heat transfer occurs between the base and the first layer.

Table 1: Thermal properties of materials.

Si linear thermal conductivity	295 W/(mK)
Si quadratic thermal conductivity	-0.491 W/(mK ²)
SiO ₂ thermal conductivity	1.38 W/(μmK)
Si specific heat	1.628 × 10 ⁶ J/m ³ K
SiO ₂ specific heat	4.180 × 10 ⁶ J/m ³ K

The TSVs are considered in the model, also as a resistance element. The most important thermal properties of the material used in the model are listed in Table 1.

Once the values of the resistances and capacitance for each cell are calculated, a set of equations, one for every unitary cell, is created. Then an iterative resolution method (Forward Euler) is used to solve the grid.

The functional units in the 3D stack can be considered as heat sources or sinks. Heat sources are the blocks which dissipate power in the die. This power is translated in heat that is spread throughout the chip. Processors are considered as strong heat sources. On the other hand, memories and communication blocks have a lower power activity and can be considered as heat sinks. Our floorplanner will try to place heat sinks and heat sources as close as possible (provided the routing and performance constraints) in order to balance the thermal profile in the 3D IC.

Once the previous model has been applied to the 3D IC, we obtain the thermal metrics (peak and mean temperature), as well as the thermal gradient and power density, which are used later in MILP and MOGA phases.

2.2 MILP phase

In this optimization phase, we used MILP because of two main reasons: (1) MILP solvers immediately check if a design scenario is feasible or not, and (2) if the problem is correctly formulated, MILP quickly offers feasible solutions.

In order to face the problem with a MILP approach we must perform several linear approximations.

The first one implies the thermal model itself, which includes non-linear and differential equations. The temperature of a unitary cell of the 3D stack, depends not only on the power density dissipated by the cell, but also on the power density of its neighbors. The first factor refers to the increase of the thermal energy due to the activity of the element, while the second one is related to the diffusion process of heat [12]. Taking this into account, we use the power density of each block as an approximation of its temperature in the steady state. This is a valid approximation because the main term of the temperature of a cell is given by the power dissipated in the cell, the contribution of its neighbors does not change significantly the thermal behavior.

The second approximation is the distance between elements, which is approximated as the Manhattan distance.

Continuing with the optimization phase, and after analyzing the temperature contribution from different blocks, we firstly sort the blocks list in descending order, according to their power density. We select the first N_h , $h = 1$ blocks from the previous list, that are supposed to be the hottest ones, according to the assumption explained before. The number of those N_h blocks is chosen by the user, depending on the architecture to optimize. In our work, we included in the list all the cores of the system because heat sources mostly

decide the final temperature behavior of the 3D integrated circuit.

Every block in the model i ($i = 1, 2, \dots, n$) is characterized by a width w_i , a height h_i and a length l_i ($i = 1, 2, \dots, n$) while the design volume has a maximum width W , maximum height H , and maximum length L . In the first MILP search algorithm (MILP1) we aim to find a feasible floorplan by maximizing distance between hottest elements (N_h). We define the vector (x_i, y_i, z_i) as the geometrical location of block i , where $x_i \geq 0$, $y_i \geq 0$, $z_i \geq 0$. We use (x_i, y_i, z_i) to denote the left-bottom-back coordinate of block i while we assume that the coordinate of left-bottom-back corner of the resultant IC is $(0, 0, 0)$. The first proposed MILP for the 3D IC, noted by MILP1, is partially formulated in Figure 4.

In Figure 4, binary variables l_{ij} , r_{ij} , u_{ij} , a_{ij} , b_{ij} and f_{ij} are equal to 1 if block i is in the left of block j , and respectively, right, under, above, behind and in front. Similarly, kx_{ij} is equal to 1 if block i is in the left of block j and in the same layer. The condition of no overlap between two blocks is guaranteed by constraints (4) to (10). The distance between two blocks in the x axis is computed through constraints (11) to (16). The same constraints are contained in the complete MILP1 model for both the y and z axis. Finally, constraint (17) computes the sum of the Manhattan distances among hottest blocks in N_h .

The optimization can be repeated several times allocating the following set of N_h , $h = 2$ blocks of the remaining sorted list (having the previous N_1 blocks fixed in the final design through constraints (1) to (3)). This procedure can be repeated until the sorted list is empty.

Finally, we move the remaining blocks (those blocks that have less power density, and are considered as heat sinks), using a second search algorithm, called MILP2 in Fig. 2. The algorithm works as MILP1 but, in this case, we do not try to maximize the distance among hottest blocks; instead, we minimize total wire length, approximated as the Manhattan distance between connected blocks (C). This way, the thermal profile of the stack will be slightly changed but wire length can be minimized. Note that MILP2 is quite similar to MILP1, with the difference that dx , dy and dz are computed for all the interconnected blocks, and J_1 is replaced by the following objective J_2 .

$$\min J_2 = \sum_{i < j \in C} (dx_{ij} + dy_{ij} + dz_{ij}) \quad (18)$$

2.3 MOGA phase

The last step of our optimization flow is to minimize the wire length in the process of placing TSVs. Technologically, TSVs can only connect two layers. In our work, we have considered connections from the top layer to any other one. The placement of the TSVs is optimized by a multi-objective genetic algorithm (MOGA). The problem of placement of TSVs in the remaining free cells cannot be integrated in the MILP model due to the large resultant number of variables and constraints. In addition, a previous analysis of free available vertical cells is required. Thus, an evolutionary algorithm will exhibit better performance for the convergence.

Next, we describe the chromosome encoding depicted in Figure 5. Since we have already placed the functional units in the MILP phase, we examine the remaining free cells in the resultant stack and build an array of x-y coordinates of allowed TSVs. Given a 3D IC with N layers, a first region

$$\begin{aligned}
& \text{(MILP1) } \max J_1 \\
& \text{s.t.} \\
& \quad \text{xmin}_i \leq x_i \leq \text{xmax}_i & i = 1, \dots, n, & (1) \\
& \quad \text{ymin}_i \leq y_i \leq \text{ymax}_i & i = 1, \dots, n, & (2) \\
& \quad \text{zmin}_i \leq z_i \leq \text{zmax}_i & i = 1, \dots, n, & (3) \\
& \quad l_{ij} + r_{ij} + u_{ij} + a_{ij} + b_{ij} + f_{ij} \geq 1 & i < j = 1, \dots, n, & (4) \\
& \quad x_i - x_j + L * l_{ij} \leq L - l_i & i < j = 1, \dots, n, & (5) \\
& \quad x_j - x_i + L * r_{ij} \leq L - l_j & i < j = 1, \dots, n, & (6) \\
& \quad y_i - y_j + W * b_{ij} \leq W - w_i & i < j = 1, \dots, n, & (7) \\
& \quad y_j - y_i + W * f_{ij} \leq W - w_j & i < j = 1, \dots, n, & (8) \\
& \quad z_i - z_j + H * u_{ij} \leq H - h_i & i < j = 1, \dots, n, & (9) \\
& \quad z_j - z_i + H * a_{ij} \leq H - h_j & i < j = 1, \dots, n, & (10) \\
& \quad dx_{ij} \geq x_i + l_i/2 - x_j - l_j/2 & i < j \in N_h, & (11) \\
& \quad dx_{ij} \geq x_j + l_j/2 - x_i - l_i/2 & i < j \in N_h, & (12) \\
& \quad x_i + l_i/2 + L * kx_{ij} \geq x_j + l_j/2 & i < j \in N_h, & (13) \\
& \quad x_j + l_j/2 + L * (1 - kx_{ij}) \geq x_i + l_i/2 & i < j \in N_h, & (14) \\
& \quad dx_{ij} \leq 2 * L * kx_{ij} + x_i + l_i/2 - x_j - l_j/2 & i < j \in N_h, & (15) \\
& \quad dx_{ij} \leq 2 * L * (1 - kx_{ij}) + x_j + l_j/2 - x_i - l_i/2 & i < j \in N_h, & (16) \\
& \quad \dots \\
& \quad J_1 = \sum_{i < j \in N_h} (dx_{ij} + dy_{ij} + dz_{ij}) & & (17)
\end{aligned}$$

Figure 4: MILP1 3D placement

of this array contains the coordinates of TSVs connecting layers N and 1, a second region contains the coordinates of TSVs connecting layers N and 2, and so forth. If the total number of allowed TSVs is M , we next build a chromosome of M 0-1 variables. If 1, a TSV is inserted in the corresponding (x,y) position (and it connects the number of layers defined in the corresponding region). In this way, Figure 5 encodes 7 TSVs in four layers ($N = 4$): 1 TSV connecting layers 4 and 1, 2 TSVs connecting layers 4 and 2, and 4 TSVs connecting layers 4 and 3. The corresponding (x,y) coordinates are stored in the array of coordinates. The larger the number of TSVs, the shorter the total wire length.

Using this representation, we run the Non-dominated Sorting Genetic Algorithm II (NSGA-II) [9] with a maximum population of one hundred, and a maximum number of 250 generations. The probability of mutation is set depending on the number of variables; in this particular case, it is the inverse of the number of available points in the plane. Then, we set a single point crossover with a probability of 0.9 and the tournament selection method.

The algorithm returns a set of solutions, considering the number of TSVs and the total wire length. This makes a Pareto front approximation, and it will be the designer who has to select the optimal solution in terms of economic cost and wire length reduction, considering that a minimum number of TSVs must be included in the design in order to fulfill communication constraints. The minimum number of TSVs is calculated considering the communication bandwidth among cores. We have calculated the data that is transferred considering an FM modulation/demodulation

application as the one explained in [8]. The maximum number of TSVs is given by the technological parameters of the TSVs and the amount of data that is transferred [11].

Now, we will define the experimental set-up, showing the floorplans that will be thermally analyzed and compared with the results obtained by our floorplanner.

3. EXPERIMENTAL SET-UP

The 3D IC stack studied in our experimental work is based on the Niagara architecture, with SPARC cores fabricated in 90nm technology. The original Niagara architecture has been modified in order to include an increased number of cores, that will be placed in several layers of our 3D stack. For this purpose, the original architecture has been replicated several times and stacked vertically to build the 3D system. Intra-layer communication is provided by a crossbar, that is scaled according to the number of cores in a layer and their required bandwidth. Inter-layer communication is resolved with a set of TSVs that route the communication signals.

The floorplanner will place the functional units that compose the 3D multi-processor architecture targeting both temperature and wire length optimization.

The thermally-optimized floorplans proposed by the floorplanner will be compared with the original configuration presented in Figure 6 and 7. In these stacks, the cores (C) are disposed in 4 and 5 layers respectively, where also the level 2 cache memories (L2), the shared memories (L2B) and the crossbar (Cross) can be seen.

Our experimental work will analyze the thermal optimization achieved by our floorplanner in two different scenarios.

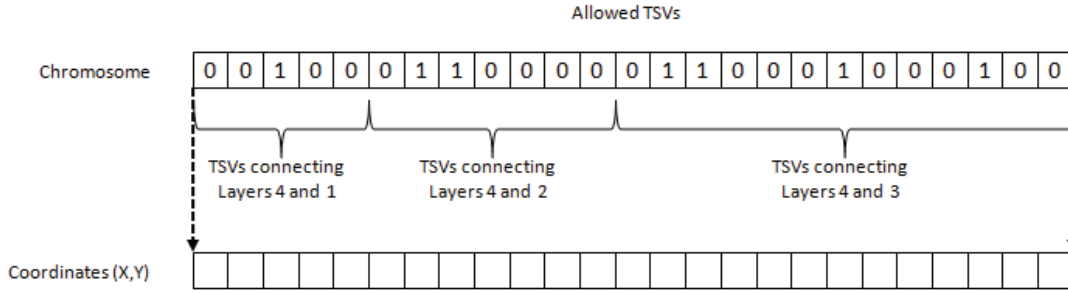


Figure 5: Chromosome description.

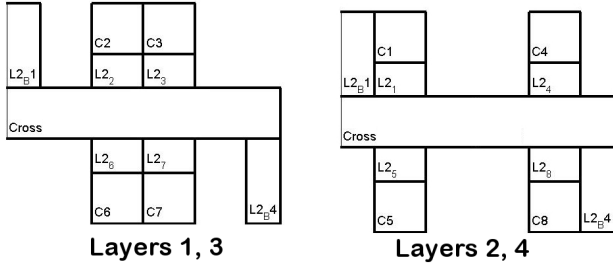


Figure 6: 16-core stack.

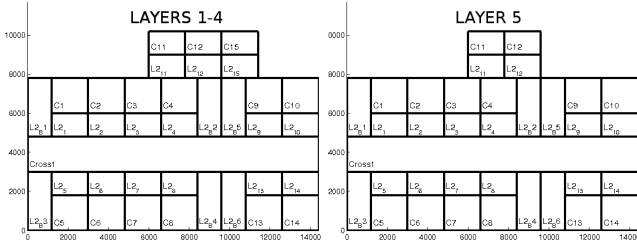


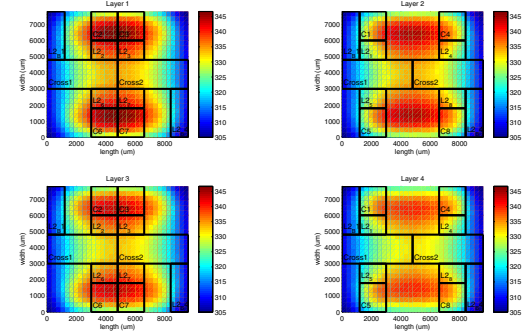
Figure 7: 64-core stack.

The first one compares the stacked Niagara architecture for 16 cores in 4 layers, with the floorplan obtained by our optimizer. The second one, compares a modified 3D Niagara architecture where 64 SPARC cores are integrated in 5 layers, with the solution of our floorplanner.

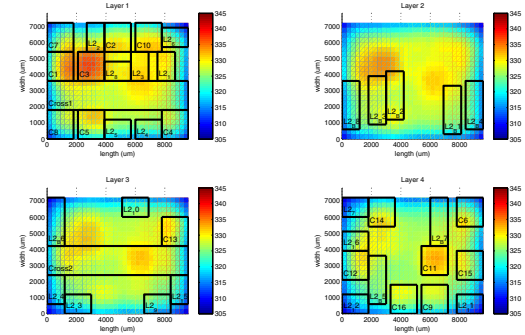
4. RESULTS

We will firstly present the thermal profile of the two original scenarios described in Section 3, that can be seen in Figures 8(a) and 9(a). In these figures, the hottest region of the 3D IC is found just in the area where the cores are. This impact is even higher when the cores are placed in the middle of the layer, as can be seen in the hot spot in Fig. 9(a).

In this paper we have considered the metrics that are usually found in all the thermal-related works for the analysis of the experimental results. These metrics are the maximum temperature and the thermal gradient, both directly related to the reliability of the integrated circuit, and the mean temperature related to the cooling process. In order to evaluate the overhead in the performance introduced by our floorplanner we evaluate also the wire length.



(a) 16-core original system



(b) 16-core optimized system

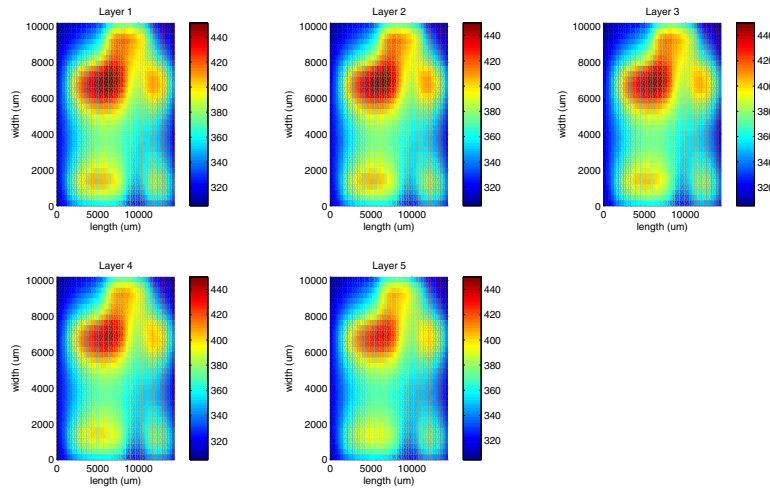
Figure 8: Comparison of the original and optimized 16-core system

Then, these results will be compared with the thermal profiles exhibited by the outputs of the floorplanner.

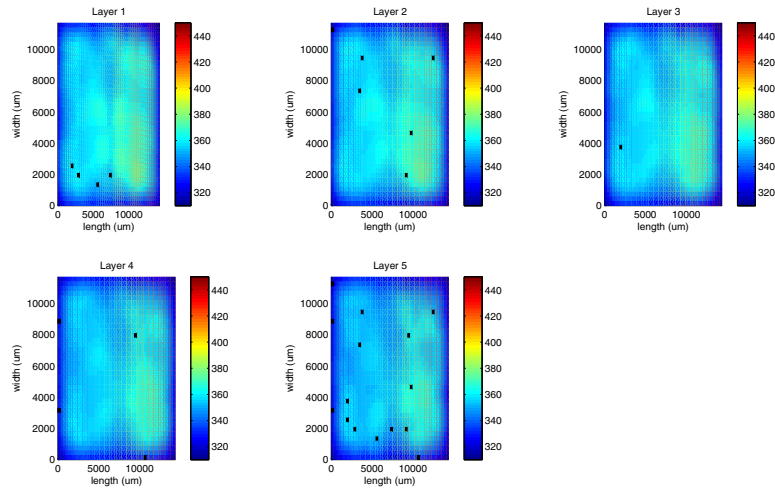
The worst case of power consumption in the Niagara2 (84W at 1.1V and 1.4GHz [1]) is considered to extract the power densities of every SPARC unit. Also, the area of the layers has been scaled accordingly to the number of cores, and the number of layers is also increased.

4.1 MILP 1 and 2 results

In the following, thermal maps obtained by MILP1 and MILP2 algorithms, explained in section 3, will be shown. MILP2, does not have a big impact in terms of thermal behavior. This is because it places heat sinks, that have a



(a) 64-core original system



(b) 64-core optimized system

Figure 9: Comparison of the original and optimized 64-core system

negligible power consumption compared to our heat sources. We will see this effect with the 16-core scenario.

In figure 10, only the cores are placed. The algorithm has tried to place the cores maximizing the distance among them, and as close as possible to the edge of the chip. Cooling down the system when the cores are in the border is much easier because their heat is directly transferred to the ambient. As can be seen in the figure, all the cores except core 13 (C13) are actually placed in the corners or in the limits of the layer, avoiding, when possible, inner layers. This can be explained because these layers can only diffuse heat to upper or lower layers, which are already hot, making the cooling much more difficult.

Figure 8(b) shows the thermal map for the whole optimized 16-core system. As was said before, no important changes in the thermal behavior can be detected. Now, our algorithm will try to minimize the performance related metric, wire length. In this algorithm, no TSVs are placed.

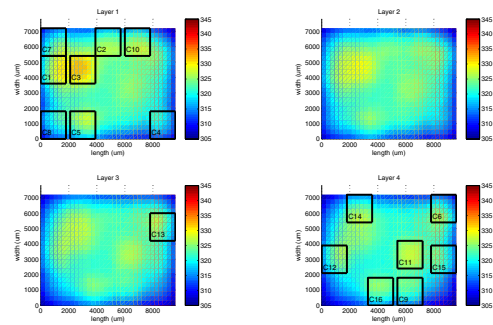


Figure 10: Thermal maps of MILP 1 output.

Inter-layer wiring is supposed to be like a single TSV that connects every layer in the stack.

The previously defined thermal metrics, mean temperature, thermal gradient and maximum temperature, have been calculated for every layer of the configuration, and are shown in table 2.

In scenario 1, the comparison with the baseline system shows that our floorplanner is capable of optimizing 36 degrees in maximum temperature, 4 degrees in mean temperature and 40 degrees in the gradient. This reduction in the temperature will optimize both the mean time to failure, reducing maximum temperature and gradient, and the lifetime of the IC by the reduction in the mean temperature.

For the sake of clarity, we will just show MILP 2 output thermal map for scenario 2. In this case the algorithm, due to the high number of functional units, cannot place cores as far as desired. On the contrary, it can locate these cores homogeneously, always maximizing the distance among them, and minimizing the distance to other functional units connected to the cores.

Similarly to the previous setup, the mean temperature, thermal gradient and maximum temperature for every layer of the configuration have been calculated. The results comparing the baseline case and the optimized 3D stack are shown in table 3.

In scenario 2, the achieved reduction is even bigger than in the previous case. We can see how our floorplanner has been able to decrease in 50 degrees the maximum temperature and the gradient in 60 degrees, keeping invariable the mean temperature, because of the spreading of the cores throughout the chip. This fact is easily noticed in Figure 9(b)

In this case, as shown in scenario 1, the reduction in the maximum temperature, and the thermal gradient across the layers, determines a more homogeneous thermal distribution, which is translated into a reduced reliability risk and diminished leakage currents.

4.2 TSVs optimization

The optimization of the placement of the TSVs is carried out using a multi-objective genetic algorithm as explained in section 2.

The algorithm gives the designer a Pareto front approximation with the number of TSVs and chip wire length. The designer will choose which solution is more convenient in every case, depending on the economic cost and technological issues. Every thermal improvement entails an overhead in the performance of the IC because of communication delays caused by the increase in wire length.

Figure 11 shows the Pareto front approximation for the 16-core system. As can be seen in the figure, the tendency is a negative exponential. This front provides to the designer the option to choose which scenario is the optimum, considering that a minimum number of TSVs must be placed. Taking into account fabrication costs 4 TSVs would be the minimum number of TSVs that still meet the computer bandwidth requirements.

Figure 12 shows the Pareto front approximation for the 64-core system. Again the exponential tendency can be seen.

In the baseline case, wire length was 969 μm and 5044 μm for the 16 and 64 core systems respectively. If the minimum, but sufficient number of TSVs is chosen, we are incurring in a 36% overhead when compared to the original distribution

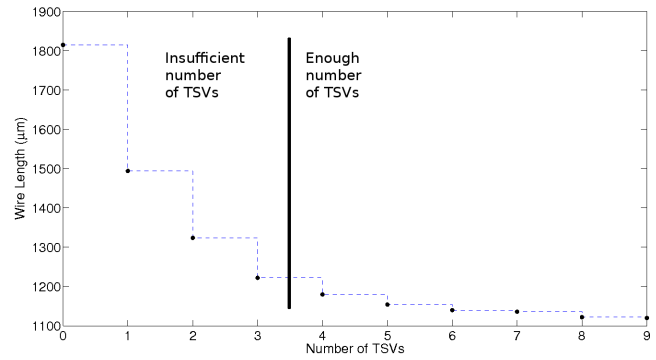


Figure 11: Pareto front approximation for 16-core system.

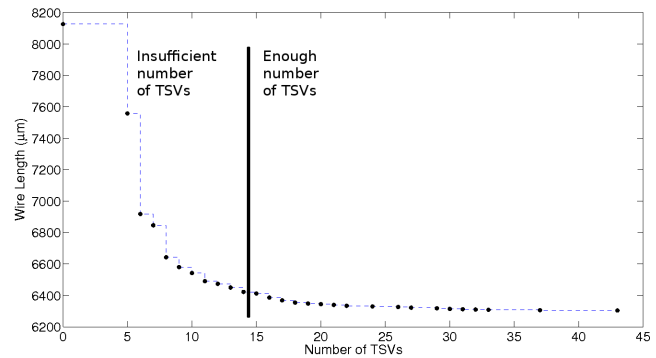


Figure 12: Pareto front approximation for 64-core system.

for 16 cores, and 32% overhead for the 64-core stack. The placement of these TSVs can be seen in Figure 9(b) where the black spots are the TSVs.

This overhead in the wiring is not directly translated into an increase of the communication delay because core-to-core communication is regulated by the crossbar. As the crossbar is the module that limits the bandwidth and speed of the link, this overhead is seen minimized. On the other hand, the big savings reached in temperature justify the overhead in wiring.

5. CONCLUSIONS

This paper has proposed a novel algorithm that uses MILP to minimize average temperature in a 3D chip, whereas uses MOGA to insert through-silicon-vias (TSVs), connecting the layers while the total wire length is minimized. Our experimental set-up analyzes two different 3D multiprocessor scenarios where the thermal- and reliability-related metrics are optimized, while still meeting the inter-core communication constraints and wiring length. These results show clear benefits in these metrics and explain the necessity of optimized placement tools for modern 3D chips.

6. REFERENCES

- [1] <http://www.opensparc.net/pubs/preszo/07/n2isscc.pdf>.
- [2] J. L. Ayala et al. Through Silicon Via-Based Grid for Thermal Control in 3D Chips. In *Nano-Net*, number 1

Table 2: 16-core Thermal Results

Scenario	Metric	Layer 1	Layer 2	Layer 3	Layer4
Original Floorplan	Max.	371.6	370.7	368.9	366.1
	Mean	329.2	328.9	328.1	327.0
	Grad	68.7	67.9	66.1	63.4
Optimized Floorplan	Max.	337.1	334.3	332.1	333.1
	Mean	325.8	324.7	324.1	323.7
	Grad	27.9	24.9	22.9	24.2

Table 3: 64-core Thermal Results

Scenario	Metric	Layer 1	Layer 2	Layer 3	Layer4	Layer 5
Original Floorplan	Max.	422.6	422.8	421.2	419.5	399.0
	Mean	350.2	341.4	349.9	348.2	340.6
	Grad	121.6	111.8	118.9	118.6	98.0
Optimized Floorplan	Max.	374.2	372.2	369.0	367.6	366.6
	Mean	350.3	341.6	350.6	349.0	347.3
	Grad	63.1	61.0	57.4	55.1	53.4

- in Lecture Notes in Computer Science, pages 1–7. Springer, 2009.
- [3] K. Balakrishnan, V. Nanda, S. Easwar, and S. K. Lim. Wire congestion and thermal aware 3d global placement. In *Proceedings of the 2005 Asia and South Pacific Design Automation Conference, ASP-DAC '05*, pages 1131–1134, New York, NY, USA, 2005. ACM.
- [4] J. Berntsson and M. Tang. A slicing structure representation for the multi-layer floorplan layout problem. In *EvoWorkshops*, pages 188–197, 2004.
- [5] G. Chen and S. Sapatnekar. Partition-driven standard cell thermal placement. In *Proceedings of the 2003 international symposium on Physical design, ISPD '03*, pages 75–80, New York, NY, USA, 2003. ACM.
- [6] C. C. N. Chu and M. D. F. Wong. A matrix synthesis approach to thermal placement. *IEEE Trans. on CAD of Integrated Circuits and Systems*, 17(11):1166–1174, 1998.
- [7] J. Cong, J. Wei, and Y. Zhang. A thermal-driven floorplanning algorithm for 3d ics. In *Proceedings of the 2004 IEEE/ACM International conference on Computer-aided design, ICCAD '04*, pages 306–313, Washington, DC, USA, 2004. IEEE Computer Society.
- [8] D. Cuesta, J. L. Ayala, J. Hidalgo, D. Atienza, A. Acquaviva, and E. Macii. Adaptive task migration policies for thermal control in mpsoes. In *ISVLSI*, pages 110–115, 2010.
- [9] K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan. A fast and elitist multiobjective genetic algorithm: NSGA-II. *IEEE Transactions on Evolutionary Computation*, 6(2):182–197, 2002.
- [10] M. Ekpanyapong, M. B. Healy, C. S. Ballapuram, S. K. Lim, and H. hsin S. Lee. Thermal-aware 3d microarchitectural floorplanning. Technical report, Georgia Institute of Technology Center for, 2004.
- [11] A. W. et Al. Bandwidth optimization for through silicon via (tsv) bundles in 3d integrated circuits. In *Design, Automation and Test in Europe*, 2009.
- [12] L. B. G. Paci, F. Poletti and P. Marchal. Exploring temperature-aware design in low-power mpsoes. *International journal of embedded systems*, 3(1):43–51, 2007.
- [13] S. Govind and C. S. Tan. Impact of thermal through silicon via (ttsv) on the temperature profile of multi-layer 3d device stack. *IEEE, 3D System Integration*, pages 1–4, 2009.
- [14] M. Healy, M. Vittes, M. Ekpanyapong, C. S. Ballapuram, S. K. Lim, H.-H. S. Lee, and G. H. Loh. Multiobjective microarchitectural floorplanning for 2-d and 3-d ics. *Computer-Aided Design of Integrated Circuits and Systems, IEEE Transactions on*, 26(1):38–52, 2007.
- [15] W.-L. Hung, Y. Xie, N. Vijaykrishnan, C. Addo-Quaye, T. Theocharides, and M. J. Irwin. Thermal-aware floorplanning using genetic algorithms. In *Proceedings of the 6th International Symposium on Quality of Electronic Design, ISQED '05*, pages 634–639, Washington, DC, USA, 2005. IEEE Computer Society.
- [16] X. Li, Y. Ma, and X. Hong. A novel thermal optimization flow using incremental floorplanning for 3d ics. In *Proceedings of the 2009 Asia and South Pacific Design Automation Conference*, pages 347–352, Piscataway, NJ, USA, 2009. IEEE Press.
- [17] G. Loh and Y. Xie. 3d stacked microprocessor: Are we there yet? pages 289–292, 2010.
- [18] M. Pathak and S. K. Lim. Performance and thermal-aware steiner routing for 3-d stacked ics. *Trans. Comp.-Aided Des. Integ. Cir. Sys.*, 28:1373–1386, September 2009.
- [19] J. Srinivasan et al. The impact of technology scaling on lifetime reliability. In *DSN*, page 177, 2004.
- [20] M. Tang and X. Yao. A memetic algorithm for vlsi floorplanning. *IEEE Transactions on Systems, Man, and Cybernetics, Part B*, 37(1):62–69, 2007.
- [21] C.-H. Tsai and S.-M. Kang. Cell-level placement for improving substrate thermal distribution. *IEEE Trans. on CAD of Integrated Circuits and Systems*, 19(2):253–266, 2000.
- [22] E. Wong and S. K. Lim. 3d floorplanning with thermal vias. In *Proceedings of the conference on Design, automation and test in Europe: Proceedings, DATE '06*, pages 878–883, 3001 Leuven, Belgium, Belgium, 2006. European Design and Automation Association.