Power Profiling-Guided Floorplanner for 3D MPSoCs

Ignacio Arnaldo, José L. Risco-Martín, José L. Ayala, and J. Ignacio Hidalgo

Abstract

3D integration has become one of the most promising techniques for the development of future multi-core processors, since it improves performance and reduces power consumption by decreasing global wire length. However, 3D integration causes serious thermal problems because the closer proximity of heat generating dies makes existing thermal hotspots more severe. Thermal-aware floorplanners can play an important role to improve the thermal profile, but they have failed in considering the dynamic power profiles of the applications. This paper proposes a novel thermal-aware floorplanner guided by the power profiling of a set of benchmarks that are representative of the application scope. The results show how our approach outperforms the thermal metrics as compared with the worst-case scenario usually considered in "traditional" thermal-aware floorplanners.

I. Introduction

In the last years, the semiconductor industry has seen an innumerable number of engineering advances that have permitted a logarithmic growth in the complexity of integrated circuits (ICs). The power density of microprocessors is also increasing with every new process generation since feature size and frequency are scaling faster than the operating voltage [7]. As a result, there has been an increase in maximum chip temperatures because power density directly translates into heat. For example, under high workload conditions, the single layer Intel Core i7 processor needs the fan to be constantly working at the highest possible speed to maintain the temperature below $39.5\,^{\circ}\mathrm{C}$ [23]. In the field of Multi-Processor Systems-on-Chip (MPSoCs), the 3D IC is gaining a lot of interest as a viable solution to help maintain the pace of system demands on scaling, performance, and functionality. The benefits include system-size reduction, performance enhancement due to shorter wire length, power reduction and the potential for hetero-integration. For complex systems with a large number of cores, only 3D technology is able to provide the required integration area while minimizing the inter-core communication delay [14].

Thermal aware floorplanning looks for an optimum floorplan by optimizing the cost function consisting of area, wire length, and temperature. The objective of the problem is to minimize the chip area, minimize the wire length, and minimize the maximum temperature of the chip. Thermal aware floorplanning can be used as one of the methods for decreasing the maximum temperature of the chip. Cooling of the blocks in a floorplan arises due to lateral spreading of heat through silicon blocks [30]. If a hot block is placed besides cooler blocks, lateral spreading of heat takes place. As a result, the temperature of the hot block is reduced.

Ignacio Arnaldo is supported by Spanish Government Avanza Competitividad I+D+I: TSI-020100-2010-962 project. The work has also been supported by Spanish Government grants TIN 2008-00508 and MEC CONSOLIDER CSD00C-07-20811.

José L. Risco-Martín, José L. Ayala, and J. Ignacio Hidalgo are with the Department of Computer Architecture and Automatics (DACYA), Complutense University of Madrid, Spain, (e-mail: jlrisco@dacya.ucm.es; jayala@fdi.ucm.es; hidalgo@dacya.ucm.es)

A common limitation of the previous methods of 3D floorplanning is that they are focused on area and/or wire length minimization with or without thermal considerations. This can be a serious limitation as modern floorplanners often have to work with a fixed die size constraint, or with a fixed outline constraint in low-level design of hierarchical floorplanning flow [1]. However, all recently developed thermal-aware tools deploy temperature estimation techniques only on a single power profile that represents the power response of all inputs and all applications (e.g. using average or peak power values). Different applications lead to different dynamic power profiles of the blocks. Most of the existing work use either average power or peak power per block of the applications for simulating temperature, without analyzing the impact of this assumption.

This work continues the proposal [3] where an efficient thermal-aware 3D floorplanner for heterogeneous architectures of MPSoCs is presented. The algorithm uses as input the power traces obtained during an application power profiling phase. The design of this floorplanner is based on a evolutionary algorithm capable of obtaining optimal solutions, in a short time, for a large number of integrated processors and layers and with minimal overhead.

II. Related Work

Some initial works on thermal aware floorplanning [9] propose a combinatorial optimization to model our problem. However, the simplification of the considered floorplan and the lack of a real experimental framework motivated the further research on the area. Thermal placement for standard cell ASICs is a well researched area in the VLSI CAD community, where we can find works as [8].

In the area of floorplanning for microprocessor-based systems, some authors consider the problem at the microarchitectural level [30], showing that a significant peak temperature reduction can be achieved by managing lateral heat spreading through floorplanning. Other works [20] use genetic algorithms to demonstrate how to decrease the peak temperature while generating floorplans with area comparable to that achieved by traditional techniques. [17] uses a simulated annealing algorithm and an interconnect model to achieve thermal optimization. These works have a major restriction since they do not consider multiple objective factors in the optimization problem, as opposed to our work. In fact, we propose a Multi-Objective Evolutionary Algorithm (MOEA) capable of simultaneously reducing the wire length and several thermal metrics. Other works [26] have tackled the problem of thermal-aware floorplanning with geometric programming but, in this case, the area of the chip is not considered constant.

Thermal-aware floorplanning for 3D stacked systems has also been investigated. Cong [11] proposed a thermal-driven floorplanning algorithm for 3D ICs, which is a natural extension of his previous work in 2D. In [18], Healy et al. implemented a multi-objective floorplanning algorithm for 2D and 3D ICs, combining linear programming and simulated annealing. Recent works as [12] also propose combinatorial techniques to tackle the problem of thermal-aware floorplanning in 3D multi-processor architectures.

In our work, we propose to use application profiling techniques to guide the floorplanner. A work by [27] shows that the power profile does not have major effect on the leakage power as long as the total power remains the same. However, they do not consider the effect of power profile on temperature variation across different applications, especially the peak temperature of the blocks. Only a recent work [31] incorporates multiple power profiles in a

thermal-aware floorplanner. However, this work is not devoted to MPSoC and could not be easily extended to 3D multi-processor stacks, where most traditional thermal-floorplanners fail to find an optimal solution.

III. Methodology

This work approaches the thermal-guided floorplanning problem for manycore heterogeneous architectures. The temperature of a given chip depends on physical factors such as the power dissipation of the processors, the size of the memories etc. but it also depends on the dynamic profile of the applications. One of our contributions is to consider energy profiles based on the simulation of real world applications. In fact, this problem is generally approached considering only the worst case scenario in terms of power dissipation. The power profiles are obtained with OVPsim [22], which is a high level multiprocessor simulator for architectural exploration.

A. Thermal Analysis

Proposed architectures: We study different architectures corresponding to the current and the nearly future state-of-the-art in 3D many-core integration. The main elements composing our architectures are memories and SPARC, ARM CORTEX-A9 and POWERPC 440 9SF processors. We study three scenarios that differ from each other in the number and percentual distribution of cores. In the first architecture there are 30 cores with a small proportion of low-power processors: 20 SPARC, 5 CORTEX-A9 and 5 PPC440. The second platform is composed of a medium number of cores with an homogeneous distribution: 22 SPARC, 22 CORTEX-A9 and 22 PPC440 adding up to a total of 66 cores. Finally, in the last scenario, there are 129 cores: 43 SPARC, 43 ARM and 43 PPC. This case corresponds to a scaled up version of the 66 processors architecture. In all cases, there is a shared memory common to all the processors (used for the inter-processor communication) and a local memory for every core. The size of the local memories must be small, as manycore architectures with big local memories are not easy to fabricate.

Benchmarks: We work with ParMiBench [24] which is composed of parallel versions of typical applications. We select 11 applications grouped into six different categories: Calculus, Network, Security, Office, Multimedia and Mixed. Therefore we have 6 different benchmarks corresponding to several applications that will exhibit very different execution profiles. As a result, the power profiles obtained are different from one benchmark to another. These benchmarks are simulated on bare machines. When an operating system is considered, the power consumption, and hence the temperature tends to be homogeneous due to the execution of the kernel. Therefore, we decide to avoid this overhead in our power profiles. The benchmarks need to be adapted for execution with OVPsim (mapping the compiled code into specific memory regions, replacing the system calls etc.). To obtain the profiling statistics for a given benchmark, we assign a parallel application and a shared memory region to each of the groups of processors working together. Figure 1 shows a task distribution example where 7 groups of six processors compute a Dijkstra shortest path algorithm and two groups of 12 processors compute a Patricia algorithm. The IDs in this table correspond to processor IDs. The processors in the same row form a group that will execute a given parallel application. With these task distributions, we aim to simulate scenarios that could happen on a real platform. Further details of the task distributions used to obtain the profiles of the different applications are given in [4].

To compute the power dissipated by each of the processors and memories of the studied architectures, we perform simulations of the 3 platforms and 6 benchmarks (18 simulations). We split the simulation of a given benchmark

Application	SPARC ID	CORTEX-A9 ID	PPC440 ID
pat1	0,1,2,3	22,23,24,25	44,45,46,47
pat2	4,5,6,7	26,27,28,29	48,49,50,51
dijk1	8,9	30,31	52,53
dijk2	10,11	32,33	54,55
dijk3	12,13	34,35	56,57
dijk4	14,15	36,37	58,59
dijk5	16,17	38,39	60,61
dijk6	18,19	40,41	62,63
dijk7	20,21	42,43	64,65

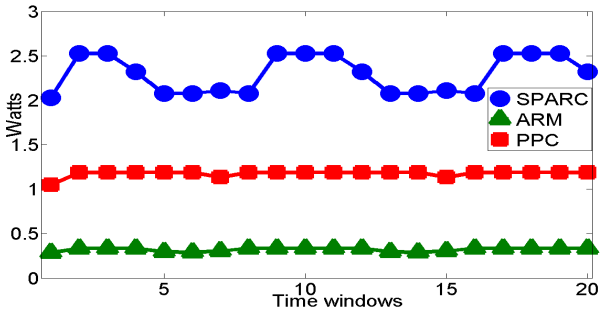


Fig. 1. Task distribution of the Network benchmark for the 66 core architecture and a common power consumption pattern caused by synchronization

in time slices called windows and study the evolution of the dissipated power versus time for every element. For each of these windows we count the executed instructions and the idle cycles per processor, as well as the read and write accesses per memory (local and shared). The simulations have to be long enough to reach a stationary state. Therefore, we obtain at least the statistical data of 100 windows. We fix the window period to 128ms. This value is chosen to be long enough to reduce the impact on performance of the profiling phase, but short enough to capture the dynamic behavior with the required accuracy.

Memories: We fix the size of the local memories to $512KB$ and the size of the shared memory to $4MB$, $8MB$ and $16MB$ for the 30, 66 and 129 cores platforms respectively. We consider both the local and shared memories to be direct mapped SRAM memories, with a block size of 64 bytes and a transistor size of 45 nm. We obtain the energy consumption and area values with the CACTI software [19]. The energy per write access E_w value is approximated by $E_w = E_r * 1.5$, where E_r is the energy per read access (see [19]).

Processors: To obtain realistic power profiles of heterogeneous platforms, we have chosen three processor architectures with a different computing power. In fact, we are specially interested in understanding the effect of synchronization and communication in the temperature of the chip. Figure 1 shows a typical power dissipation pattern of three different processors working together. We can see clearly how the activity of the SPARC core changes periodically over time, waiting for slower processors. We assume that the energy consumption of a given processor depends on its working frequency and its state. We consider two states: active or idle. To compute the power densities of the processors, we need their areas and a power consumption value for both the active and idle states. In [28] we find that the power consumption of the SPARC is 4W at 1.4GHz. In the case of the CORTEX-A9, we find that 0.4W is the estimated power dissipation working at 830 MHz while the PPC440 9SF dissipates 1.1W at 667MHz (see [2] and [21]). As in [5], we approximate the power dissipated in the idle state with $P_{idle} = P_{active}/10$. We consider the following areas: $3.24mm^2$, $1.5mm^2$ and $6.2mm^2$ for the SPARC, CORTEX-A9 and PPC440 respectively (see [28], [2] and [21]).

B. Multi-Objective Evolutionary Algorithm

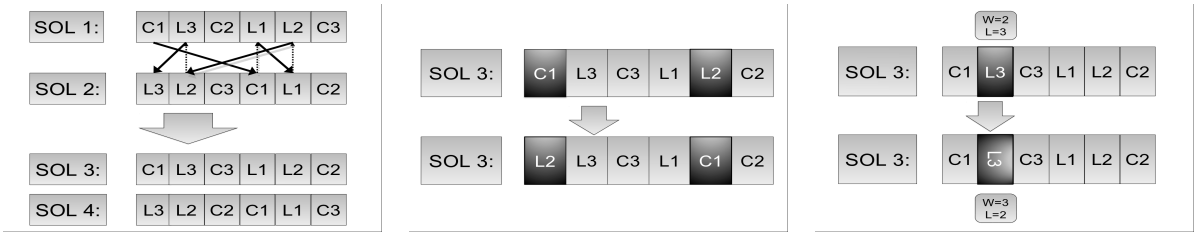
Introduction to MOEA: Most of the algorithms presented for the 3D thermal aware floorplanning problem are based on a Mixed Integer Linear Program (MILP) [18], [25], Simulated Annealing (SA) [11], [18] or Genetic Algorithms (GA) [33]. MILP has proven to be an efficient solution. However, when MILP is used for thermal

aware floorplanning, the linearized thermal model must be added to the topological relations and the resultant algorithm becomes too complex [16], specially as the problem size (number of cores and memories, in our case) increases. Regarding SA and GA, the main issue is based on the representation of the solution. Some common representations are polish notation [6], combined bucket array [11] and O-tree [33]. Most of these representations do not perform well, because they were initially developed to reduce area. In the thermal aware floorplanning problem, the hottest elements must be placed as far as possible in the 3D IC. In this work, the optimization process is achieved with the Multi-Objective Evolutionary Algorithm presented in [3]. The algorithm is based on Non-Dominated Sorting Genetic Algorithm II (NSGA-II) [15] and tries to minimize the maximum temperature and the total wire length while fulfilling all the topological constraints. In this work, as opposed to traditional floorplanning problems in 2D, the area of the chip is not initially targeted as we consider a fixed die size.

Block placement problem: All the blocks that model the different components of the many-core system must be placed in the 3D stack, which imposes the physical boundaries of maximum length L, width W, and height H. Every block i in the model $B_i(i = 1, 2, \ldots, n)$ is characterized by a width w_i, a height h_i and a length l_i. We define the vector (x_i, y_i, z_i) as the geometrical location of block B_i, where $0 \leq x_i \leq L - l_i$, $0 \leq y_i \leq W - w_i$, $0 \leq z_i \leq H - h_i$. We use (x_i, y_i, z_i) to denote the left-bottom-back coordinate of block B_i while we assume that the coordinate of left-bottom-back corner of the resultant IC is $(0, 0, 0)$.

In order to apply MOEAs to our problem, a suitable representation of each individual has first to be found. Furthermore, an initial population has to be created, as well as defining a cost function to measure the fitness of each solution. For an overview of MOEAs the reader is referred to [10].

Representation and operators: We must guarantee that all the chromosomes represent real and feasible solutions to the problem and ensure that the search space is covered in a continuous and optimal way. To this end, we use a permutation encoding [10], where every chromosome is a string of labels, that represents the block placement sequence. Figure 2 depicts the different operators used in our MOEA, the example shows a platform composed of 6 blocks: 3 processors $C_i(i = 1, 2, 3)$ and 3 memories $L_i(i = 1, 2, 3)$.



(a) Cycle crossover (b) Swap mutation (c) Rotation

Fig. 2. MOEA operators: Cycle crossover and two mutation operators (swap or rotate).

The selection operator implements a binary tournament strategy: random couples of individuals are formed and the best solution of each pair is selected. As Figure 2(a) depicts, we apply the cycle crossover: starting with the first allele of chromosome 1 (C_1), we look at the allele at the same position in chromosome 2 (L_3). Next, we move

to the position containing that same allele in the first chromosome, and add this allele to the cycle. The previous step is repeated until the first allele of the first individual (C_1) is selected for a second time. Finally, the alleles of the cycle are placed in the first child respecting the positions occupied in the first parent, and the next cycle is obtained from the second parent. Mutation can be executed in two different ways, both with the same probability (see Figure 2(b) and 2(c)). As a result, some blocks are chosen and swapped, and others are rotated 90 degrees.

Fitness function: Each chromosome represents the order in which blocks are being placed in the design area. Every block B_i is placed taking into account all the topological constraints, the total wire length, and the maximum temperature in the chip with respect to all the previously placed blocks $B_j : j < i$. In order to place a block i, we take the best point (x_i, y_i, z_i) in the remaining free positions. To select the best point we stablish a dominance relation that takes into account the m following objectives in our multi-objective evaluation.

The first objective is determined by the topological relations among placed blocks. It represents the number of topological constraints violated (no overlapping between placed blocks and current volume less or equal than maximum volume). The second objective is the wire length. The wire length is approximated as the Manhattan distance between interconnected blocks. The following objectives $(3, 4, \ldots, m)$ are a measure of the thermal impact, each one based on a profile of power consumption. To compute the thermal impact of every profile of power consumption we cannot use an accurate thermal model, which includes non-linear and differential equations. In a classical thermal model, the temperature of a unitary cell of the chip, 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 [29]. We use the simplified thermal model presented by Cuesta *et. al.* [13] in which both the power densities of the different blocks and the contribution of their neighbors are considered. Thus, our remaining objectives can be formulated as:

$$J_{k\in 3..m} = \sum_{i<j\in 1..n} (dp_i^{k-2} * dp_j^{k-2})/(d_{ij}) \tag{1}$$

where dp_i^p is the power density of block i for power consumption p, and d_{ij} is the Euclidean distance between blocks i and j.

In our case, we obtain up to 600 different power consumptions (100 time windows $\times$ 6 applications). Obviously, 600 objectives is too high for a MOEA, since it will converge too slow. Therefore, we discuss how to reduce this number of objectives in the next section.

IV. Experimental setup

The experimental work will analyze the thermal optimization achieved by the floorplanner in the three different scenarios presented in section 3.1 (30, 66 and 129 cores architectures). The floorplanner will place the processors, the local and shared memories of the 3D manycore platforms in 3, 4 and 5 layers respectively. For each of the three scenarios, we obtain four different floorplans:

1) As we do not have any original configuration to compare with, we propose as baseline a performance optimized floorplan targeting only the feasibility and the wire length. From now on, this configuration

will be called BAS.

In order to obtain the thermally optimized floorplans, it is not possible to take directly into account all the data retrieved from our simulations. In fact, if the power dissipation of every element for every benchmark and time window was considered, the floorplanner would target 600 objectives and would hardly converge. Therefore, to obtain the other three floorplans, we consider different power metrics computed with the data retrieved from the simulation of 100 time windows for the 6 different execution profiles.

2) The first one of the remaining configurations is obtained considering the mean power dissipation for each element and profile (AVG). Then, the floorplanner looks for feasible solutions that minimize six thermal objectives (one per profile) and the wire length.
3) Another configuration is obtained considering only the highest power consumption per element (WOR). Hence, three objectives are targeted: feasibility, a thermal objective and the wire length. This case corresponds to the strategy followed by other thermal-aware floorplanners.
4) Finally, a weighted sum of the power consumptions of the different profiles (all weights are equal) is considered for each element (WSM). In this case, the algorithm targets feasibility, a thermal objective and the wire length.

We run the evolutionary algorithm with a population of 100 individuals and 500 generations. These parameters are fixed according to previous research. The crossover probability p_c is fixed to 0.90 and the mutation probability p_m to $1/\#blocks$ (see [15]). We consider a fixed area equal to the total sum of the areas of the different elements. This value is increased in a 15% corresponding to the minimal area overhead necessary for wire routing. The configurations obtained are chosen among a front of non-dominated solutions returned by the floorplanner. In all the studied cases, we select the configuration that minimizes the wire length. In fact, finding the best solution of the non-dominated front is out of the scope of this paper and is postponed to a future work.

V. Results

As explained in the previous section, we obtain four different floorplans (BAS, AVG, WOR and WSM) for each of the three considered scenarios (30, 66 and 129 cores platforms). In this section, we compare the thermal profiles of these different configurations. To evaluate them, we propose two experiments. The temperatures presented in this work are given in degrees Kelvin and obtained with the thermal simulator 3D-ICE presented in [32].

A. Thermal response in the worst case

In the first experiment, the obtained floorplans are evaluated with the highest power dissipation per element. This case corresponds to the worst scenario. The metrics considered for the analysis of the experimental results are the mean and maximum temperature of the chip and the maximum thermal gradient. These metrics are usually found in thermal-related analysis. In Table I, we present the thermal profiles of the four different configurations.

The results show that our power profiling-guided floorplanner produces thermally optimized configurations. The hotspots found in the performance optimized floorplans (reaching 443.15K) justify the thermal optimization presented in this work. Compared to the baseline, we can see that in all the cases our floorplanner reduces the peak

$\#cores$	30			66			129		
	T_{max}	T_{mean}	Gr_{Max}	T_{max}	T_{mean}	Gr_{Max}	T_{max}	T_{mean}	Gr_{Max}
BAS	416.30	355.22	108.44	440.24	357.35	136.10	443.15	361.56	137.57
AVG	394.80	350.10	84.93	410.187	352.41	101.92	396.33	355.32	87.06
WOR	388.42	349.03	73.46	401.27	349.55	91.22	414.45	355.98	104.59
WSM	387.68	349.73	74.12	403.34	349.40	91.742	400.44	354.63	91.75

TABLE I

WORST CASE

and mean temperatures and the thermal gradient. Therefore, not only the temperature of the chip is reduced but it is also more evenly distributed. For example, we can appreciate that the dramatic peak temperature of 416.30K found in the baseline is reduced to 387.68K in the WSM configuration of the 30 core platform. We illustrate this example in Figure 3 where we show the thermal maps of the first layer of the BAS and WSM configurations. In the baseline configuration, the floorplanner tends to place the processors near their local memories ignoring the presence of hotspots. In the other, the hottest elements (the SPARC cores) are separated as much as possible, generally placed in the borders of the chip. As a consequence, we observe a much better thermal response of the WSM configuration. Vertical heat spread is also taken into account. Hence, the floorplanner avoids placing cores above the others. In both cases the shared memory is placed in the second layer to minimize the wire length.

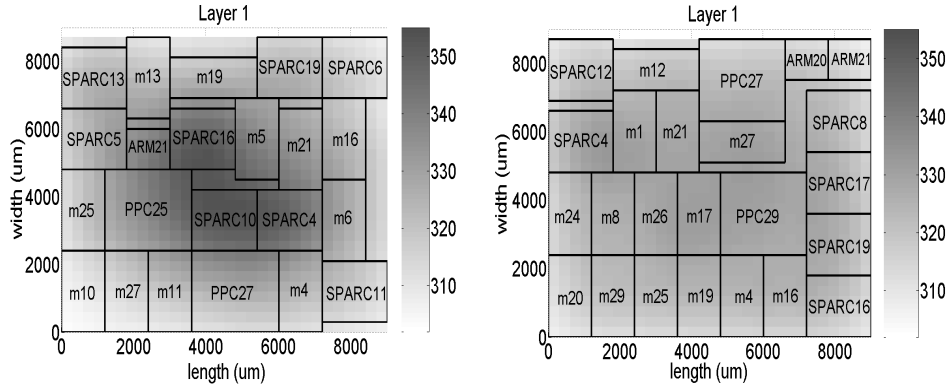


Fig. 3. Thermal map of the first layer of the 30 cores BAS(left) and WSM(right) configurations

The baseline configuration (BAS) is not an acceptable solution, in fact it reaches peaks of 443.15K in the 129 cores configuration. Moreover, the peak temperature increases with the number of cores in all of the cases. Therefore, from now on we will use the baseline to compare the wire length of the thermally optimized floorplans, but we will not study its thermal response. Choosing a configuration among the thermally optimized floorplans is not immediate. It is due to the fact that some of these configurations present a better performance (wire length) while others have a better thermal behavior. We propose a selection criterion in section V-C.

B. Thermal response for the different benchmarks

In the second experiment, we evaluate the same floorplans in a more realistic way. The thermal behavior of the different configurations is simulated for 100 time windows with the power dissipation values obtained from our execution profiles. Three metrics are considered in this case to compare the different configurations. The mean of

the maximum temperatures of the different time windows is computed as well as its standard deviation. This metric is a good indicator of the existence of hotspots in the studied chip. We also compute the overall mean temperature of the chip and the mean of the maximum thermal gradients as well as their respective standard deviations. Table II shows the metrics retrieved for the six studied benchmarks in the 30, 66 and 129 cores scenarios. We now comment the results obtained in the three different scenarios.

1) 30 cores scenario: The results obtained for the peak temperatures are not conclusive as none of the configurations clearly outperforms the others. The highest difference is obtained in the case of the first Benchmark where the temperature reached by the WSM is 2.8K and 5.02K lower than the one obtained with the WOR and AVG configurations respectively. The mean temperatures are very similar for all the configurations. In fact there is a maximum difference of 1.12K between the different configurations (Benchmark 5). As for the peak temperatures, a first analysis of the maximum thermal gradients does not lead to an immediate selection of the best configuration.

2) 66 cores scenario: The peak temperatures obtained in the WSM are lower than those obtained in the AVG configuration saving up to 7.22K. Only in the case of the Benchmark 1 the peak temperature of the WOR configuration is the lowest one, with a difference of 1.63K over the WSM. On the other hand, selecting WSM leads to a maximum reduction of 2.63K in the peak temperature for the Benchmark 3. Once again, there are no significant differences between the mean temperatures of the studied configurations. In the case of the maximum thermal gradients, the results are similar to those obtained for the peak temperature: the WSM outperforms the other configurations in five out of six cases, reducing up to 10.05K the gradient of the AVG (Benchmark 5) and 3.09K the gradient of the WOR (Benchmark 3). In the case of the Benchmark 1, the best thermal gradient is obtained for the WOR configuration, with a difference of 1.27K and 4.75K with the WSM and AVG configurations respectively.

3) 129 cores scenario: The best peak temperature is obtained for the AVG configuration in four out of six benchmarks saving up to 17.12K and 6.71K comparing with the WOR (Bench 5) and WSM (Bench 1) configurations respectively. The WSM outperforms the others in the remaining two cases, reducing up to 10.80K the maximum temperature of the WOR configuration and 2.98K the one of the AVG. The mean temperatures obtained are again homogeneous. The results for the thermal gradients are similar to the ones obtained for the peak temperature. Globally, the WOR configuration shows a worst thermal behavior than the WSM or AVG. Nevertheless, it is the configuration that presents the best performance (wire overhead), therefore it can not be discarded.

Further analysis is required to obtain the best overall solutions as the achieved temperatures seem to show a random behaviour. In the next section, we perform a deeper analysis of the quality of the solutions in terms of wire length and thermal behaviour in each of the three scenarios. The proposed method shows that there is an optimal metric to guide the floorplanning of the studied architectures.

C. Performance/temperature tradeoff

As showed in the previous section, there is a tradeoff between performance (translated into the extra wiring) and temperature. We propose a method to evaluate the different solutions and select the one with the best overall behavior. To this end, we consider 21 different experiments: 3 architectures $\times$ (worst case $+$ 6 benchmarks). For

30						
	Tmax					
	BENCH1	BENCH2	BENCH3	BENCH4	BENCH5	BENCH6
AVG	$340.95_{6.58}$	$342.55_{4.56}$	$340.22_{7.80}$	$310.35_{3.15}$	$348.33_{2.68}$	$343.75_{3.38}$
WOR	$338.73_{7.44}$	$345.51_{2.66}$	$341.12_{11.09}$	$311.43_{2.82}$	$347.48_{2.60}$	$340.29_{3.22}$
WSM	$335.93_{6.30}$	$345.56_{3.79}$	$340.80_{10.02}$	$309.37_{2.89}$	$346.65_{2.56}$	$341.38_{2.63}$
	Tmean					
	BENCH1	BENCH2	BENCH3	BENCH4	BENCH5	BENCH6
AVG	$322.43_{6.14}$	$320.11_{2.26}$	$324.70_{5.59}$	$306.28_{1.95}$	$331.41_{1.65}$	$325.56_{2.86}$
WOR	$322.19_{5.79}$	$319.82_{2.19}$	$324.33_{5.49}$	$306.24_{1.91}$	$330.29_{1.58}$	$324.80_{2.70}$
WSM	$322.50_{6.07}$	$319.96_{2.36}$	$324.64_{5.65}$	$306.21_{1.98}$	$331.37_{1.64}$	$325.49_{2.95}$
	Tgrad					
	BENCH1	BENCH2	BENCH3	BENCH4	BENCH5	BENCH6
AVG	$35.88_{5.83}$	$39.45_{4.58}$	$34.40_{6.32}$	$8.47_{2.63}$	$42.70_{2.43}$	$38.73_{3.60}$
WOR	$31.85_{6.25}$	$40.66_{2.65}$	$32.25_{7.97}$	$9.54_{1.95}$	$36.52_{2.08}$	$31.57_{4.45}$
WSM	$29.36_{4.44}$	$41.99_{3.99}$	$34.05_{8.39}$	$7.03_{2.24}$	$36.59_{2.10}$	$34.18_{2.83}$
66						
	Tmax					
	BENCH1	BENCH2	BENCH3	BENCH4	BENCH5	BENCH6
AVG	$334.17_{9.35}$	$365.43_{7.36}$	$386.49_{3.14}$	$369.64_{1.64}$	$349.52_{3.12}$	$369.16_{4.64}$
WOR	$329.77_{5.60}$	$366.91_{6.16}$	$385.90_{5.75}$	$369.60_{1.94}$	$342.91_{2.63}$	$368.72_{5.68}$
WSM	$331.40_{6.28}$	$364.66_{7.52}$	$383.27_{1.96}$	$369.33_{1.62}$	$342.30_{2.62}$	$367.96_{5.62}$
	Tmean					
	BENCH1	BENCH2	BENCH3	BENCH4	BENCH5	BENCH6
AVG	$319.61_{5.22}$	$325.63_{1.06}$	$338.79_{3.43}$	$324.90_{0.60}$	$328.27_{1.51}$	$324.63_{0.62}$
WOR	$318.36_{4.16}$	$325.91_{0.92}$	$337.96_{3.81}$	$324.77_{0.54}$	$325.23_{1.36}$	$324.51_{0.71}$
WSM	$318.57_{4.40}$	$325.27_{0.96}$	$337.29_{3.18}$	$324.33_{0.55}$	$325.40_{1.36}$	$323.83_{0.65}$
	Tgrad					
	BENCH1	BENCH2	BENCH3	BENCH4	BENCH5	BENCH6
AVG	$29.50_{8.42}$	$62.35_{7.65}$	$80.75_{2.22}$	$66.99_{1.87}$	$44.48_{3.04}$	$66.25_{4.98}$
WOR	$24.75_{3.95}$	$64.35_{6.60}$	$79.99_{5.51}$	$67.35_{2.24}$	$35.35_{2.56}$	$66.42_{6.43}$
WSM	$26.02_{4.93}$	$61.41_{7.94}$	$76.90_{1.23}$	$66.91_{2.02}$	$34.43_{2.38}$	$65.45_{6.59}$
129						
	Tmax					
	BENCH1	BENCH2	BENCH3	BENCH4	BENCH5	BENCH6
AVG	$342.31_{7.94}$	$357.95_{0.78}$	$373.18_{4.97}$	$352.95_{1.50}$	$370.42_{4.13}$	$331.63_{2.46}$
WOR	$354.08_{8.90}$	$363.76_{5.07}$	$382.00_{9.27}$	$352.40_{2.51}$	$387.54_{5.19}$	$338.28_{2.93}$
WSM	$349.02_{4.73}$	$363.99_{0.87}$	$371.20_{4.10}$	$351.14_{1.54}$	$373.87_{4.34}$	$336.10_{2.68}$
	Tmean					
	BENCH1	BENCH2	BENCH3	BENCH4	BENCH5	BENCH6
AVG	$327.32_{5.80}$	$328.68_{1.34}$	$338.78_{4.53}$	$321.70_{1.86}$	$344.24_{2.36}$	$322.20_{1.57}$
WOR	$328.25_{5.95}$	$329.19_{1.49}$	$339.01_{4.77}$	$321.57_{1.89}$	$344.87_{2.40}$	$322.57_{1.61}$
WSM	$327.99_{5.52}$	$328.52_{1.43}$	$338.20_{4.99}$	$320.56_{1.98}$	$344.12_{2.35}$	$322.55_{1.59}$
	Tgrad					
	BENCH1	BENCH2	BENCH3	BENCH4	BENCH5	BENCH6
AVG	$36.08_{6.31}$	$54.77_{1.00}$	$66.81_{3.52}$	$51.02_{1.07}$	$62.04_{3.77}$	$25.86_{2.14}$
WOR	$47.75_{7.79}$	$60.06_{4.79}$	$75.95_{8.28}$	$50.70_{1.97}$	$78.82_{4.80}$	$32.12_{2.56}$
WSM	$43.71_{3.80}$	$61.04_{0.91}$	$65.84_{3.22}$	$49.45_{1.14}$	$66.28_{3.98}$	$31.90_{2.49}$

TABLE II

THERMAL METRICS RETRIEVED IN THE 30, 66, AND 129 CORES SCENARIOS

each of the scenarios and thermal metrics studied (T_{Max}, T_{mean} and $Grad_{Max}$) we establish a confidence interval. In a second step we see whether or not the retrieved metrics fall into these ranges of acceptable values.

We observe that the thermal metrics retrieved fit to a normal distribution. The intervals are obtained by adding and subtracting the standard deviation to the mean of the considered metric, resulting in 68% confidence intervals. We apply this procedure in all the studied scenarios to obtain the confidence intervals. However, we only show the intervals obtained in the case of Benchmark 4 in the 30 cores platform as a motivational example:

- T_{Max}: The confidence interval is obtained by adding and subtracting the standard deviation σ_{Tmax} to the mean of the peak temperature of the different configurations called μ_{Tmax}. The resultant interval is: $[\mu_{Tmax} - \sigma_{Tmax}; \mu_{Tmax} + \sigma_{Tmax}] \simeq [310.38 - 1.03; 310.38 + 1.03] \simeq [309.35; 311.41]$
- T_{mean}: The same method is used to obtain the confidence interval for the mean temperatures: $[\mu_{Tmean} - \sigma_{Tmean}; \mu_{Tmean} + \sigma_{Tmean}] \simeq [306.24 - 0.035; 306.24 + 0.035] \simeq [306.205; 306.275]$
- $Grad_{Max}$: In a similar way, we obtain the range of acceptable values for the maximum thermal gradient: $[\mu_{MaxGr} - \sigma_{MaxGr}; \mu_{MaxGr} + \sigma_{MaxGr}] \simeq [8.35 - 1.26; 8.35 + 1.26] \simeq [7.09; 9.61]$

These intervals allow to group similar values into the same thermal level. In fact a difference of 1K might not be relevant to decide which configuration is better. On the other hand, stepping from a 15% of wire overhead to 25% would result in a dramatic decrease of the chip performance. Once we have these intervals, we see which metrics fall into these intervals (marked as $\surd$ in the following example) and which ones do not (X). There is a third possibility where the metric considered is even below the confidence interval (marked as $\surd\surd$), which is the most desirable situation as the goal is to minimize the wire length and the different thermal metrics. We perform the analysis of the worst case scenario and the 6 benchmarks for the three considered architectures (30, 66 and 129 cores). Based on the obtained confidence intervals, we decide which is the best configuration for each of the 21 considered cases. Table III shows the result of this analysis for the Benchmark 4 in the 30 cores scenario. In this example, only the WSM configuration satisfies all the constraints.

30	T_{max}	T_{mean}	$Grad_{Max}$
AVG	$\surd$	X	$\surd$
WOR	X	$\surd$	$\surd$
WSM	$\surd$	$\surd$	$\surd\surd$

TABLE III
BENCHMARK 4

$\#cores$	30	66	129
AVG	36.88%	16.19%	24.81%
WOR	23.24%	26.68%	17.69%
WSM	13.15%	17.25%	22.31%

TABLE IV
WIRE OVERHEAD OF THE DIFFERENT CONFIGURATIONS

In order to select the configuration with the best overall behavior, we take into account the thermal response and the wire length of the different configurations (see Table IV). We now summarize the results of this analysis.

1) 30 cores scenario: In this scenario, the AVG configuration only presents acceptable values in one of the seven scenarios. The WSM and WOR configurations satisfy all the constraints in six out of seven benchmarks. They offer a satisfactory performance while respecting the different thermal metrics studied in this work. However the WSM offers a better performance as it presents 10.09% less of wire overhead. Therefore the WSM is the best configuration for the 30 cores scenario.

2) 66 cores scenario: The WOR configuration offers a poor performance. The AVG presents the best performance but it does not satisfy the thermal constraints in four out of seven benchmarks, including the worst case scenario. Therefore it offers a poor response in extreme situations, leading to hotspots. Only the WSM configuration satisfies all the constraints in all the cases. It presents the best overall thermal behavior while only the AVG presents a slightly better performance (1.06%). Hence, the WSM is also chosen in this scenario as the best configuration.

3) 129 cores: The selection of the best configuration is not as immediate as in the previous scenarios. Even though the WOR configuration presents the best performance, it is discarded as the metrics retrieved do not fall in the acceptance intervals for four out of seven benchmarks, reaching 414.45K in the worst case scenario. Both AVG and WSM satisfy all the constraints in all the cases. While the AVG offers a better overall thermal response, the WSM configuration presents a shorter wire length increasing the performance in 2.5%. Therefore, in this scenario, there is a tie between these two configurations.

Globally, the WSM configuration always presents an acceptable performance. In addition, the thermal constraints are satisfied in 20 out of 21 cases. The AVG and WOR configurations only present an acceptable thermal response in 10 and 14 cases respectively. Moreover, these two configurations do not offer acceptable performances: the wire overhead of the AVG reaches 36.88% in the 30 cores scenario while in the 66 cores scenario the WOR configuration presents a 26.68% of extra wire. Furthermore, the convergence analysis shows that using the WSM metric leads to an average convergence time reduction of 43.1% and 36.2% as compared to the WOR and AVG metrics respectively. Therefore, the WSM presents the best overall behavior, outperforming AVG and the traditionally used WOR metric.

VI. Conclusion

This work has proposed an efficient approach that incorporates power-profiling information to guide a thermal-aware floorplanner for 3D multi-processor architectures. The implementation of the tool with evolutionary algorithms has provided thermal-optimized floorplans as compared with a baseline heterogenous system. The proposed analysis of the tradeoff between thermal behavior and performance shows that considering the worst power consumption does not lead to optimal floorplans. In fact, the configurations targeting a weighted sum of the power consumption of the different benchmarks present a better overall behavior. Such configuration offers a better thermal response in extreme conditions reducing in 14.01K and 12.84K the peak temperature and the thermal gradient of the chip respectively.

References

[1] Adya, S., et al.: Fixed-outline floorplanning: enabling hierarchical design. VLSI Systems, IEEE Transactions on 11(6), 1120 – 1135 (dec 2003)

[2] ARM: www.arm.com/products/processors/cortex-a/cortex-a9.php

[3] Arnaldo, I., Risco-Martin, J.L., Ayala, J.L., Hidalgo, J.I.: Power Profiling-Guided Floorplanner for Thermal Optimization in 3D Multiprocessor Architectures. In: Integrated Circuit and System Design. Power and Timing Modeling, Optimization, and Simulation, Lecture Notes in Computer Science, vol. 6951, pp. 11–21. Springer Berlin / Heidelberg (2011)

[4] Arnaldo, I.: Evolutionary approaches to solve the 3D thermal-aware floorplanning problem using heterogeneous processors. Master's thesis, Complutense University, Madrid (2011)

[5] Atienza, D., Del Valle, P.G., Paci, G., Poletti, F., Benini, L., Micheli, G.D., Mendias, J.M., Hermida, R.: HW-SW emulation framework for temperature-aware design in MPSoCs. ACM Trans. Des. Autom. Electron. Syst. 12(3), 26:1–26:26 (May 2008)

[6] Berntsson, J., Tang, M.: A slicing structure representation for the multi-layer floorplan layout problem. In: EvoWorkshops. pp. 188–197 (2004)

[7] Borkar, S.: Design challenges of technology scaling. Micro, IEEE 19(4), 23 –29 (jul-aug 1999)

[8] Chen, G., Sapatnekar, S., Inc, S.: Partition-driven standard cell thermal placement (2003)

[9] Chu, C., et al.: A matrix synthesis approach to thermal placement. CADICS, IEEE Transactions on 17(11), 1166 –1174 (nov 1998)

[10] Coello, C., et al.: Evolutionary Algorithms for Solving Multi-Objective Problems. Kluwer (2002)

[11] Cong, J., Wei, J., Zhang, Y.: A thermal-driven floorplanning algorithm for 3D ICs. In: Proceedings of the 2004 IEEE/ACM International conference on Computer-aided design. pp. 306–313. ICCAD '04, IEEE Computer Society, Washington, DC, USA (2004)

[12] Cuesta, D., Risco, J.L., Ayala, J.L., Atienza, D.: 3D Thermal-Aware Floorplanner for Many-Core Single-Chip Systems. IEEE Latin-American Test Workshop (2011)

[13] Cuesta, D., Risco-Martin, J.L., Ayala, J.L.: 3D thermal-aware floorplanner using a MILP approximation. Microprocessors and Microsystems (2012), http://www.sciencedirect.com/science/article/pii/S0141933112000294

[14] Davis, W., Wilson, J., Mick, S., Xu, J., Hua, H., Mineo, C., Sule, A., Steer, M., Franzon, P.: Demystifying 3D ICs: the pros and cons of going vertical. Design Test of Computers, IEEE 22(6), 498 – 510 (nov-dec 2005)

[15] Deb, K., Pratap, A., Agarwal, S., Meyarivan, T.: A fast and elitist multiobjective genetic algorithm: NSGA-II. Evolutionary Computation, IEEE Transactions on 6(2), 182 –197 (apr 2002)

[16] Ekpanyapong, M., Healy, M.B., Ballapuram, C.S., Lim, S.K., hsin S. Lee, H.: Thermal-aware 3D Microarchitectural Floorplanning. Tech. rep., Georgia Institute of Technology Center for (2004)

[17] Han, Yongkuiand Koren, I.: Simulated annealing based temperature aware floorplanning. J. Low Power Electronics 3(2), 141–155 (2007)

[18] Healy, M., et al.: Multiobjective microarchitectural floorplanning for 2D and 3D ICs. CADICS, IEEE Transactions on 26(1), 38–52 (jan 2007)

[19] HPlabs: www.hpl.hp.com/research/cacti/

[20] Hung, W.L., Xie, Y., Vijaykrishnan, N., Addo-Quaye, C., Theocharides, T., Irwin, M.: Thermal-aware floorplanning using genetic algorithms. In: Quality of Electronic Design, 2005. ISQED 2005. Sixth International Symposium on. pp. 634 – 639 (march 2005)

[21] IBM, C.D.C.: Complex SoC design (2009)

[22] Imperas: www.ovpworld.org

[23] Intel: http://www.intel.com/cd/channel/reseller/apac/tha/products/desktop/processor/processors/corei7-900/feature/431132.htm

[24] Iqbal, S., et al.: ParMiBench - an open-source benchmark for embedded multiprocessor systems. CAL 9(2), 45 –48 (feb 2010)

[25] Li, X., et al.: A novel thermal optimization flow using incremental floorplanning for 3D ICs. In: ASPDAC. pp. 347–352. IEEE Press (2009)

[26] Li, Y., aothers: Temperature aware floorplanning via geometry programming. In: CSE WORKSHOPS IEEE International Conference on. pp. 295 –298 (july 2008)

[27] Liu, Y., et al.: Accurate temperature-dependent integrated circuit leakage power estimation is easy. In: DATE. pp. 1526–1531 (2007)

[28] OpenSPARC: http://www.opensparc.net/pubs/preszo/07/n2isscc.pdf (2007)

[29] Paci, G., Marchal, P., Poletti, F., Benini, L.: Exploring "temperature-aware" design in low-power MPSoCs. Design, Automation and Test in Europe Conference and Exhibition 1, 180 (2006)

[30] Sankaranarayanan, K., Velusamy, S., andCharles L, M.S., Skadron, K.: A case for thermal-aware floorplanning at the microarchitectural level. JILP 7(1), 8–16 (2005)

[31] Singhal, L., Oh, S., Bozorgzadeh, E.: Statistical power profile correlation for realistic thermal estimation. In: Proceedings of the 2008 Asia and South Pacific Design Automation Conference. pp. 67–70. IEEE Computer Society Press, Los Alamitos, CA, USA (2008)

[32] Sridhar, A., Vincenzi, A., Ruggiero, M., Brunschwiler, T., Atienza, D.: 3D-ICE: Fast compact transient thermal modeling for 3D ICs with inter-tier liquid cooling. In: Computer-Aided Design (ICCAD), 2010 IEEE/ACM International Conference on. pp. 463 –470 (nov 2010)

[33] Tang, M., Yao, X.: A Memetic Algorithm for VLSI Floorplanning. Systems, Man, and Cybernetics, Part B: Cybernetics, IEEE Transactions on 37(1), 62 –69 (feb 2007)