HANDS: Heterogeneous Architectures and Networks-on-Chip Design and Simulation

Davide Zoni, Simone Corbetta and William Fornaciari
Politecnico di Milano – Dipartimento di Elettronica e Informazione
Via Ponzio 34/5, 20133 Milano, Italy
{scorbetta, fornacia}@elet.polimi.it

ABSTRACT

In current multi-core scenario, Networks-on-Chip (NoC) represent a suitable choice to face the increasing communication and performance requirements, however introducing additional design challenges to already complex architectures. In this perspective, there is a need for flexible and configurable virtual platforms for early-stage design exploration. We present the *Heterogeneous Architectures and Networks-on-Chip Design and Simulation* framework for large-scale high-performance computer simulation, integrating performance, power, thermal and reliability metrics under a unique methodology. Moreover, NoC exploration is possible from a reliability/performance and thermal/performance trade-offs.

Categories and Subject Descriptors

B.3.3 [**Perfomance Analysis and Design Aids**]: Simulation

Keywords

Multi-core, Network-on-Chip, Simulation, Reliability

1. INTRODUCTION

Continuous technology scaling of recent decade leads to an exponential increase in processor performance, with power consumption going as faster as clock rate. The transition to multi-core architectures introduced an opportunity for performance to grow faster than power consumption, while the need for even more performance and integration of cores in a single chip leads to the definition of novel architectural solutions to cope with unmanageable communication contention. In this scenario NoC truly became the appropriate design paradigm to manage increasing performance and reliability requirements [2]. However such on-chip networks are expected to consume significant part of the total chip power [4]. In particular a few commercial designs show a NoC power consumption up to 28% of the total chip power [5]. Meanwhile, increasing operating temperature caused by increasing power consumption density is continuously affecting the reliability of VLSI systems: experimental results show that high temperature is responsible for more than 50% of failures in CMOS integrated circuits [12].

The need to consider such a huge amount of architectural design aspects requires appropriate methodologies for accurate analysis. Such methodologies can be focused on simulation and analytical modeling. Simulation represents the most accurate method to extract valuable information on the architecture, while can be very time consuming. On the other side, analytical models reduce evaluation time even if the output data can be affected by greater errors. Moreover, analytical models provide an intrinsic suitable way for further optimization methodology, while simulation data are almost raw. However analytical models are usually employed for the analysis of specific parts of the architecture after a system-wide analysis, since they are difficult to extract and their characterization requires low level information from real or simulated architecture. In this scenario, this work proposes a novel framework for joint thermal, performance and power analysis to be used both at early design stages, while the extracted information can be used for further localized platform optimizations and trade-off exploration.

1.1 Novel contributions

The proposed work focuses on an accurate, flexible and extensible tool that allows to explore different design space dimensions, i.e. performance metrics, thermal and power profiles as well as reliability issues, during early design stages, covering: *(i)* NoC thermal impact and *(ii)* trade-off analysis and optimization. Indeed traditional approaches of solely caring about performance is nowadays superseded by more critical and multi-dimensional constraints. In this perspective the thermal issues are of paramount importance for both performance and architecture lifetime maximization. Moreover, on-chip interconnect contribution to the chip temperature cannot be neglected, since it represents a consistent part of the total heat flow generated. The proposed simulation framework allows for an accurate thermal chip evaluation accounting for both computational and communication blocks, at different levels of detail.

Furthermore, current multi-core architectures exhibit complex architecture design, since multiple design dimensions should be accounted at the same time. Moreover, the analysis and optimization of orthogonal design dimensions must be conducted both at design-time and dynamically. The proposed framework allows to analyze both compile-time and run-time situations. First, our framework can impose

ISLPED'12, July 30–August 1, 2012, Redondo Beach, CA, USA.

a fine control on detail levels and number of optimization dimensions to be extracted from the simulation to manage design-time trade-offs. Moreover, the complexity of both current multi-core architectures as well as parallel applications, negatively impact the possibility for a complete and accurate design-time optimization, since an amount of metrics can be evaluated at run-time only. In this perspective, the proposed framework allows for run-time policy evaluation allowing for further policy optimization, safety analysis and cutting edge stress tests.

The remainder of the paper is organized as follows. Section 2 provides an overview of the state-of-the-art simulation frameworks, highlighting their major features and limitations, focusing on Network-on-Chip analysis. Section 3 provides an in-depth discussion of the proposed framework, discussing the advantages as compared to state-of-the-art solutions. Experimental results are reported in Section 4, focusing on a detailed analysis of relevant use cases. Conclusions are finally drawn in Section 5.

2. RELATED WORKS

Several proposals can be found in literature for power, performance and thermal estimation of single-core and multi-core processors. Nevertheless, only a few are focused on a comprehensive approach to jointly estimate multiple design dimensions considering Network-on-Chip. This work provides an accurate and flexible design tool that accounts for all the different design aspects for high-performance multi-core architectures, with particular emphasis on NoCs.

Different works that can be found in literature are meant for single-core analysis. Nevertheless, the advent of multi-core architectures is pushing strict requirements on multi-core simulation: power, performance and temperature should be be jointly analyzed at system-level, and considering the on-chip networks due to their increasing relevance in parallel architectures. In this perspective, literature lacks a suitable simulation methodology general enough to deal with all the design aspects described above. The `SESC` simulator [13] provides cycle-accurate simulation of bus-based multi-core processors, based on MIPS architecture. However, it does not support Network-on-Chip architectures. The `Polaris` framework [15] can provide power and area estimates of Network-on-Chip architectures. Since the focus is on the interconnect, it lacks for a detailed power estimation for both processors and memory hierarchy. Although precise, it is not suitable to be employed in system-level computer architecture research. Power, area and thermal modeling are accounted for also in the `SST` framework [6]. The work focuses on large-scale systems, but application traces are emulated, rather than collected from cycle-accurate simulation, with higher simulation rate at the cost of much lower accuracy. Power and thermal models are proposed in [3], based on the `Simics` functional simulator. The interesting achievement in this approach lies on the possibility to develop, analyze and tune different control algorithms for thermal and power management, based on high-level Matlab descriptions. The work is suitable for designing control-theoretic thermal management solutions, although bound to a particular architecture, ISA and floorplan (precisely, the reference architecture is an Intel©Xeon X7350 system). The work presented in [10] is meant to simulate large-scale architectures, and exploits parallel simulation on physical hardware. It can simulate several cores based on the MIPS in-order architecture. However, the output thermal map refers to the only communication infrastructure, without providing a system-wide perspective from a thermal view-point. In addition, each of the above approaches lacks in the reliability aspects, i.e. MTTF projection that is jointly coupled to chip temperature profile. Table 1 finally summarizes the advantages and pitfalls of these works, and reports a comparison with the framework presented in this paper.

3. PROPOSED EVALUATION FLOW

We provided an appropriate set of modifications to third party tools, while a complete set of other tools has been developed from scratch.

The logical snapshot of the proposed virtual platform is given in Figure 1. Steps are executed in pipeline fashion, i.e. each step provides input to the subsequent one, and requires output from the previous, ensuring a flexible and extensible tool. The four steps involved are: cycle-accurate simulation, power consumption estimation, floorplan generation and thermal/reliability models. Cycle-accurate simulation is required to provide relevant access and usage statistics at architecture and microarchitecture level. During simulation, the most relevant information from the modeled architecture are acquired, e.g. accesses to instruction fetch unit, number of committed integer instructions, number of stall cycles in the pipeline and the like. For processors, this means that we can collect sensible metrics about the status of the hardware pipeline while executing the software application; for NoC routers, on the other hand, we collect raw metrics about network interface accesses and traffic patterns. Memory-related statistics are also required, providing a system-wide set of performance and access statistics. Accurate power consumption estimates are required for both processing cores and interconnect primitives, as well as for storage blocks. Last, temperature profile is required to evaluate the impact of hardware or software design choices on the reliability and power consumption of the entire architecture. There are also three different feedback paths: temperature feedback exists from the thermal model to the power model for leakage power analysis; an additional temperature feedback is used to back-annotate the simulator, useful for temperature-aware scheduling policies evaluation; last, power back-annotation is used to allow for power management policies to be developed and estimated along the flow. These feedback paths can be employed in a more general fashion to provide an analysis of power-related and temperature/reliability-related design choices, either at design-time or run-time. In Figure 1, white boxes are those related to third-party software, gray boxes represent in-house developed tools, while striped boxes are those tools for which relevant modifications have been performed. The rest of this section gives an insight of each step, the tools involved in each stage and the modifications that have been applied to third-party tools.

3.1 Cycle-accurate simulation

We employ `GEM5` cycle-accurate performance simulator (`http://gem5.org/`), using the *syscall emulation* approach that models bare-metal execution and can simulate the underlying hardware with precise processor models and collecting statistics at microarchitecture level. `GEM5` can also supports *Full-system* simulation mode, that requires and simulates an OS to support application execution, but introduces a

Table 1: Features of state-of-the-art multi-core simulation frameworks compared to the proposed flow.

Framework	Cycle-accurate simulation	NoC support	Power support	Thermal support	Reliability projection	Floorplan exploration	Objectives
Renau et al. (SESC) [13]	✓	✗	✗	✗	✗	✗	multi-core simulation, parallel applications
Soteriou et al. (Polaris) [15]	✗	✓	✓	✗	✗	✗	Network-on-Chip design-space exploration
Hsieh et al. (SST) [6]	✗	✓	✓	✓	✗	✗	microarchitecture, power and thermal
Lis et al. (HORNET) [10]	✗	✓	✓	✓	✗	✗	many-core processors, mainly NoC interconnect
Bartolini et al. [3]	✗	✓	✓	✓	✓	✗	run-time control policies evaluation
Our flow	✓	✓	✓	✓	✓	✓	microarchitecture, NoC, reliability, design-space exploration

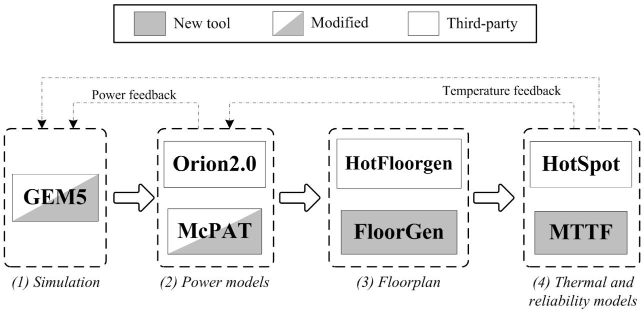


Figure 1: Proposed estimation flow.

great computational overhead. We are mainly interested in bare-metal execution, providing HW-based simulation of synchronization primitives where required; we also introduce the support for clock-toggling processors. Indeed, current and future multi-core architectures have to face with thermal and reliability issues, as well as performance and power ones. Most of the run-time thermal management and power management techniques rely on dynamic voltage and frequency scaling (DVFS) to control both chip temperature and power consumption, or on clock-gating hardware support to cool down processors. We provide a per-CPU clock-toggling configurable implementation, in which the user can synthesize the desired duty-cycle for thermal/performance tradeoff exploration, either at design-time or run-time;

3.2 Power estimation

Power estimation is accounted for using different tools for processor and Network-on-Chip router, as a function of the access statistics provided by the cycle-accurate simulation phase. McPAT [9] is used to generate power estimates of the core and memory architectures, as a function of the access statistics provided by the cycle-accurate simulation phase presented in Section 3.1. Orion2.0 [8], on the other hand, is employed for computing power contribution of NoC routers and links, as a function of traffic and packets traversing the network. Two significant improvements have been made to these tools. The original version of McPAT takes as input a single temperature value to compute leakage contribution: the entire chip region is assumed to work at the same operating temperature. This assumption does not fit well with architecture modeling, for two reasons. At first, it is impractical that different regions of the chip experience the same amount of temperature [14], due to the asymmetric load assignment, especially in multi-core architectures. In addition, the chip temperature profile is an aspect of paramount importance for reliable design, while the assumption provides an overlay simplistic scenario. The proposed flow, on the other hand, is able to annotate the correct temperature to each microarchitecture block in the processor, thus providing a way to better estimate leakage power contribution. In addition, McPAT provides a discretized amount of leakage levels, ranging from 300K to 400K temperatures at steps of 10K. The available temperature range is thus reduced to 11 values, providing an impractical scenario for aggressive thermal simulations. In our flow, on the other hand, the temperature range is fully covered, and leakage curve within temperature steps at distance 10K are approximated linearly. This aspect, along with the feedback from thermal model (refer to Section 3.3) provides a comprehensive and more accurate estimation.

3.3 Temperature estimation

Temperature estimation is an essential phase in the computer architecture research, due to the increasing relevance of temperature-aware designs. We use HotSpot [14] thermal model for multi-core thermal map evaluation. This model requires a chip floorplan, and a set of power measurements to compute steady-state temperature. The main improvement we propose in this work is related to a flexible and customizable floorplan generation tool coupled to HotFloorplan, part of the HotSpot model release. Since HotFloorplan does provide single-core floorplan only, we developed FloorGen to generate the floorplan for the desired multi-core architecture. We focused on two main aspects: to provide flexibility to generate any desired floorplan, and to provide the flexibility to generate the floorplan at any desired level of details. Up to now, we target only 2D mesh topologies, based on the Alpha-21364 network architecture [11], but we are able to generate the floorplan of each core according to user-defined requirements: the user can thus specify core floorplan, and let the tool generate a multi-core architecture with core replication. An interesting support in this direction has been made to integrate the output from HotFloorplan with FloorGen. Notice that this step is entirely decoupled from the cycle-accurate simulation, since core access statistics are not affected by the floorplan since the wires are assumed to be dense in their respective microarchitecture block. The floorplan impact on the power consumed by router links is disscussed in Section 3.2.

Table 2: Processor and router setup.

Processor core	4GHz, in-order Alpha21264 core
Int-ALU	4 integer ALU functional units
Int-Mult/Div	4 integer multiply/divide functional units
FP-Mult/Div	4 floating-point multiply/divide functional units
L1 cache	64kB 2-way set assoc. split I/D, 2 cycles latency
L2 cache	1.75MB per bank, 8-way associative (shared)
Router	3-stage wormhole switched (Garnet network [1])
Topology	2D-mesh, based on Alpha21364 network processor
Technology	65nm at 1.2V, and 45nm at 1.1V

3.4 Reliability analysis

The last logical block from Figure 1 is used to compute reliability projection. Temperature-dependent reliability estimate is done through Mean Time To Failure (MTTF) analysis of different mechanisms: electromigration, stress-migration and thermal cycling. MTTF for these processes is known to be exponentially dependent on temperature, and this library provides an easy way to perform reliability-directed design optimizations with direct input from the simulated architecture. This contribution makes the proposed framework suitable for aggressive reliability projections and hardware/software estimation.

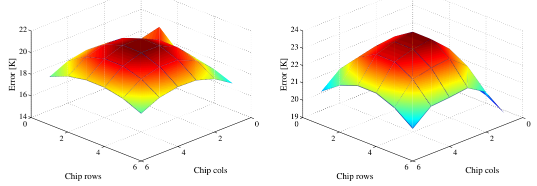
4. EXPERIMENTAL RESULTS

This section highlights the flexibility of the proposed framework when different design dimensions are considered. Three main results are discussed: Section 4.1 shows how the proposed flow can be used to easily analyze the impact of the interconnect on the chip thermal profile. Section 4.2 presents a simple exploration and evaluation of the thermal coupling coefficients accounting for both communication and computational logic. Last, Section 4.3 presents a reliability/performance trade-off analysis exploiting the provided clock toggling implementation.

In the experiments, we consider an Alpha-21364 network processor as reference architecture [11]. This is composed of tiles, organized along a 2D-mesh topology: each tile is composed of an in-order version of the Alpha-21264 processor core, with private L1 cache and shared distributed L2 cache; the router is used to interface the processing core to the local and shared memory.The router is a 3-cycles delay architecture, and local L2 cache bank surrounds both processing core and router as in [11]. The main features and technology parameters used throughout the experiments are summarized in Table 2.

4.1 The role of Network-on-Chip

The NoC has great impact on the temperature distribution within the SoC. This section details how such thermal contribution increases with integration density, and technology node scaling down. Figure 2 shows the thermal impact of the NoC on the architecture. The results are shown for a 36-cores processor, considering 45nm and 65nm technology nodes. The surface reports the absolute estimated temperature mismatch while neglecting the NoC contribution, considering integer-intensive workload from SPEC-CPU 2006 benchmark suite. Each point approximating the surface represents the absolute temperature error in a specific core in the floorplan; thus, points are ordered by means of rows and columns. At first, it is clear how the peak contribution of the NoC routers increases while technology node scales down. As a matter of fact, the maximum temperature error increases from 20K to 24K while going through 65nm and 45nm nodes, respectively. Moreover, the spread is increased, and a higher variation is observed in lower technology nodes. Secondly, the higher error is experienced a the central region of the chip, where coupling is maximum (refer to Section 4.2 for more details on this).



(a) 65nm (b) 45nm

Figure 2: The estimation error obtained without considering the impact of the Network-on-Chip temperature is maximum at the center of the chip, and diminishes towards the perimeter.

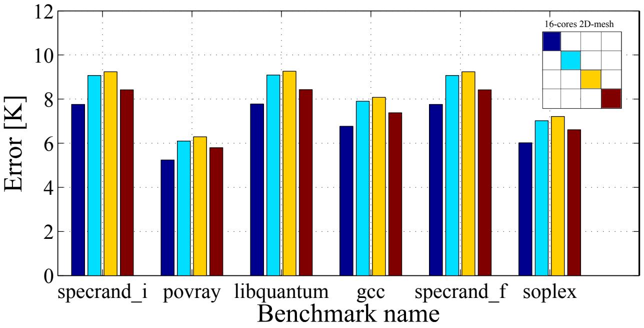


Figure 3: Maximum observed temperature mismatch while considering the NoC impact.

The impact that the NoC has on the processing cores, from a temperature view-point, is also summarized in Figure 3 considering a 45nm 16-cores processor running different applications from the SPEC-CPU 2006 suite. Six different applications are proposed, each core running the same application. The error is reported in temperature degrees, and corresponds to the absolute difference between steady-state temperatures when cores only and cores plus NoC routers are considered. Results are given for the four cores along the diagonal of the architecture, and have been collected after executing 30×10^6 instructions[1]. As it can be seen, the NoC router has a great impact on the temperature of adjacent cores: on average, 6K up to 9K degrees are experienced. This demonstrates the importance of a system-level temperature estimation methodology and thermal design, to ensure proper design choices and increase reliability of the final system. In addition, notice that different applications lead to different impact: in general, the higher the communication required by the applications running in the processor, the higher will be the impact of the NoC.

[1] Notice that for steady-state analysis and single-phase applications, this is almost enough to get realistic temperature values.

4.2 Thermal coupling in multi-core architectures

The increasing availability of multi-core architectures is driven by continuous shrinking technology and achievements in architecture design, e.g. Network-on-Chip supporting hard power/performance constraints. With increased power density, operating temperature increases, and with shrinking technologies the role of mutual influence increases as well. Core-to-core thermal coupling has been shown to get worse with technology scaling [7], however the impact of Network-on-Chip has not been quantified, yet. The proposed estimation flow allows to easy quantify the relative impact of computational cores and communication blocks. To this extent, the multi-core processor is modeled as a set $A = \{a_1, a_2, ...\}$ of $|A|$ blocks being either processor cores, interconnect routers or memory banks. Architecture blocks are connected each other and their physical position is determined by the floorplan. Notice that there might be cases in which the floorplan is either unknown, or its complexity is just unmanageable. For these reasons, we capture the floorplan information indirectly, through the use of observed temperature measurements, and adopt an appropriate thermal coupling model to quantify the mutual influence. The effects of heat transfer between adjacent blocks are seen as a net increase in temperature, induced by proximity to active blocks. The magnitude of this influence is modeled through the thermal coupling coefficient, denoted with $\psi_{ij}(p) = \frac{T^p_{ji} - T^p_i}{T^p_{ji} - T_{amb}}$: a non-linear function of the blocks relative position, the relative temperatures and power levels, and system configuration (package solution and thermal design solution. We denote with T^p_j the *steady-state* temperature of block a_j while consuming (on average) a power of p Watts. This is the temperature value due to self-heating, i.e. while consuming power. We then introduce T^p_{ji} denoting the steady-state temperature of block a_j when block a_i is loaded at power level p, and a_j is assumed to be idle. At quiet-state, each block a_i is assumed to be at T_{amb}.

For each block a_i, we compute the aforementioned coefficient; an extensive procedure is employed for this purpose. The procedure takes as input the chip floorplan and the power traces generated by the cycle-accurate simulation and power/area models, and it is repeated for each tile j in the architecture; notice that we hereby consider the core, the router and the L2 cache as a whole, as done in [7]. We then selectively switching off a portion of the available tiles, and then compute the self-heating contribution T_j. Coupling contribution, on the other hand, is computed by switching off alternatively the remaining tiles k different from j, to get an estimate of T^k_j. Last, coupling coefficient is computed using the aforementioned $\psi_{ij}(p)$ metric.

Figure 4 shows the thermal coupling index considering both cores and routers as the "destination" block of induced temperature. The results are taken considering a single tile for temperature coupling as a whole (Figure 4(c)), focusing on the direct impact that neighbor tiles have on cores (path (a) in Figure 4(c)) and routers (path (b)). The temperature values have been simulated using applications from the SPLASH-2 benchmark suite. The proposed metric is able to capture the spatial correlation of temperature, since coefficients have decreasing value moving far from the destination (tile at position $(0, 0)$ in this case). Also, there is a slightly different absolute value in thermal coupling for the core and router case due to chip geometry, although their magnitude is comparable.

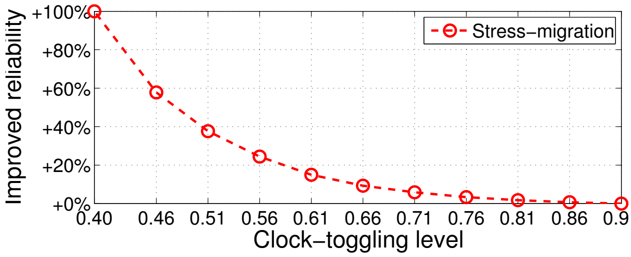
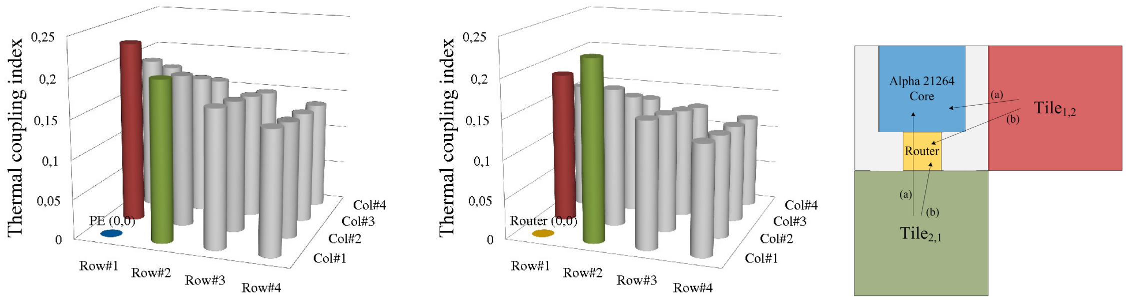


Figure 5: Performance/reliability improvement against base-case scenario (MTTF = 1).

4.3 Reliability/Performance Exploration

Thermal analysis represents a requirement of paramount importance in current multi-core designs. For instance, reliability optimizations focus on minimizing operating temperature, to increase the MTTF and reduce the probability of faults. However, a constrained chip temperature can degrade performance. In this section we discuss the performance/reliability trade-off addressing hard-faults only, while considering MTTF degradation caused by stress-migration. We consider a 45nm 36-cores 2D-mesh architecture and we apply different clock toggling levels to diminish temperature, thus improving MTTF. Even if the current clock toggling implementation allows for per-core clock manipulation, we apply the same clock toggling level to each core to provide the same performance degradation on each core. Clock-toggling in this context is defined as the amount of idleness that is required to synthesize the desired duty-cycle level of the processor: thus, a clock-toggling level of 0.9 allows to synthesize a 10% duty-cycle processor usage. More sophisticated clock toggling designs should consider different clock toggling levels on different cores, based on the running applications, multi-core floorplan and desired temperature profile, however such methodology is out the scope of this work. Starting from the described architecture, we compute, for different clock toggling levels, the MTTF expression $MTTF_{SM} \propto |T_0 - T|^{-n} \cdot exp\left\{\frac{E_{SM}}{k\ T}\right\}$ taken from [16], where E_{SM} is the energy activation for stress-migration, k is the Boltzmann's constant, T the operating temperature, T_0 the reference temperature for (melting temperature), and n is a technology-dependent parameter. We use the values for these parameters as given in [16].

Figure 5 shows the reliability projection of the 36-cores processor, while constraining the clock toggling level. Notice that the hottest core only is reported, having all the other cores at lower temperature, thus with higher MTTF. The dotted line shows the reliability trend (MTTF values) when applying the desired clock-toggling. The horizontal axis reports the clock toggling level required to accommodate the MTTF improvement. The vertical axis reports the target reliability improvement relative to base case when reliability equals 1 and no clock modifications have been applied. For instance to increase by 40% MTTF caused by stress-migration in 36-cores processor, performance level should be diminished to 51%.



(a) PE coupling (b) Router coupling (c) Configuration

Figure 4: Thermal coupling coefficients trend, considering tiles as source and PEs/routers as destination.

5. CONCLUSIONS

This work presented a cycle-accurate simulation framework for accurate analysis of thermal issues in modern multicore architectures, focusing on a joint analysis of computational blocks and Network-on-Chip communication fabric. Moreover, the proposed tool allows for exploring different design-time metrics of interest, such that thermal/performance and reliability/performance trade-off optimization. The flexibility of the proposed work allows to perform accurate and detailed joint analysis on power, performance, thermal and reliability metrics. The simulation flow is composed of state-of-the-art tools, as well as tools developed from scratch. We then described and discussed a set of experiments, using SPEC CPU 2006 and SPLASH-2 benchmark suites, showing the flexibility of the framework to support several kind of estimation objectives and methodologies.

6. ACKNOWLEDGMENTS

This research work is supported by European Community Seventh Framework Programme (FP7/2007-2013), under agreements no. 248716 (2PARMA project www.2parma.eu).

7. REFERENCES

[1] N. Agarwal, T. Krishna, L.-S. Peh, and N. Jha. Garnet: A detailed on-chip network model inside a full-system simulator. In *Performance Analysis of Systems and Software IEEE International Symposium on*, 33–42, 2009.

[2] A. Banerjee, R. Mullins, and S. Moore. A Power and Energy Exploration of Network-on-Chip Architectures. In *NOCS '07*, 163–172. IEEE Computer Society, 2007.

[3] A. Bartolini, M. Cacciari, A. Tilli, L. Benini, and M. Gries. A virtual platform environment for exploring power, thermal and reliability management control strategies in high-performance multicores. In *GLSVLSI'10*, 311–316, New York, NY, USA.

[4] S. Borkar. Networks for multi-core chips: a contrarian view. In *Special Session at ISLPED*, 2007.

[5] Y. Hoskote, S. Vangal, A. Singh, N. Borkar, and S. Borkar. A 5-ghz mesh interconnect for a teraflops processor. *Micro, IEEE*, 27(5):51 –61, 2007.

[6] M.-y. Hsieh, A. Rodrigues, R. Riesen, K. Thompson, and W. Song. A framework for architecture-level power, area, and thermal simulation and its application to network-on-chip design exploration. *SIGMETRICS Perform. Eval. Rev.*, 38:63–68, 2011.

[7] M. Janicki, J. H. Collet, A. Louri, and A. Napieralski. Hot spots and core-to-core thermal coupling in future multi-core architectures. In *26th Annual IEEE SEMI-THERM Symposium*, 205–210. 2010.

[8] A. Kahng, B. Li, L.-S. Peh, and K. Samadi. Orion 2.0: A fast and accurate noc power and area model for early-stage design space exploration. In *DATE '09.*, 423 –428, april 2009.

[9] S. Li, J. H. Ahn, R. Strong, J. Brockman, D. Tullsen, and N. Jouppi. Mcpat: An integrated power, area, and timing modeling framework for multicore and manycore architectures. In *Microarchitecture, 42nd Annual IEEE/ACM International Symposium on*, 469 –480, 2009.

[10] M. Lis, P. Ren, M. H. Cho, K. S. Shim, C. Fletcher, O. Khan, and S. Devadas. Scalable, accurate multicore simulation in the 1000-core era. In *Performance Analysis of Systems and Software (ISPASS), IEEE International Symposium on*, 175 –185, 2011.

[11] S. S. Mukherjee, P. Bannon, S. Lang, A. Spink, and D. Webb. The alpha 21364 network architecture. In *Proceedings of the The 9th Symposium on High Performance Interconnects*, Washington, DC, USA, 2001.

[12] M. Pedram and S. Nazarian. Thermal Modeling, Analysis, and Management in VLSI Circuits: Principles and Methods. *Proceedings of the IEEE*, 94(8):1487–1501, August 2006.

[13] J. Renau, B. Fraguela, J. Tuck, W. Liu, M. Prvulovic, L. Ceze, S. Sarangi, P. Sack, K. Strauss, and P. Montesinos. SESC simulator, January 2005.

[14] K. Skadron, M. R. Stan, K. Sankaranarayanan, W. Huang, S. Velusamy, and D. Tarjan. Temperature-aware microarchitecture: Modeling and implementation. *ACM Transactions on Architecture and Code Optimization (TACO)*, 1(1), 2004.

[15] V. Soteriou, N. Eisley, H. Wang, B. Li, and L.-S. Peh. Polaris: A system-level roadmap for on-chip interconnection networks. In *ICCD* , 134 -141, 2006.

[16] J. Srinivasan, S. Adve, P. Bose, and J. Rivers. The case for lifetime reliability-aware microprocessors. In *Computer Architecture, Proceedings. 31st Annual International Symposium on*, 276 - 287, 2004.