Joint Exploration of Architectural and Physical Design Spaces with Thermal Consideration*

Yen-Wei Wu, Chia-Lin Yang, Ping-Hung Yuh
Department of Computer Science and Information Engineering
National Taiwan University
Taipei, Taiwan
{r92089, yangc, r91089}@csie.ntu.edu.tw

Yao-Wen Chang
Department of Electrical Engineering &
Graduate Institute of Electronics Engineering
National Taiwan University
Taipei, Taiwan
ywchang@cc.ee.ntu.edu.tw

ABSTRACT

Heat is a main concern for processors in deep sub-micron technologies. The chip temperature is affected by both the power consumption of processor components and the chip layout. Therefore, for thermal-aware design it is crucial to consider the thermal effects of different floorplans during micro-architectural design space exploration. In this paper, we propose a thermal-aware architectural floorplanning framework. With the aid of this framework, an architect can explore both physical and architectural design spaces simultaneously to find an architecture and the corresponding chip layout that maximizes performance under a thermal limitation.

Categories and Subject Descriptors

B.8.2 [**Hardware**]: PERFORMANCE AND RELIABILITY

General Terms

Algorithm, Design, Performance

Keywords

Thermal, Architectural Floorplanning, Performance

1. INTRODUCTION

As the technology continues to improve, power density in microprocessors increases steadily. Power density is predicted to reach $100W/cm^2$ at technology below 50nm [2]. High die temperatures reduce device reliability and cause timing errors. Moreover, transistor speed is slower at higher temperatures, and leakage power grows exponentially as temperature increases. Therefore, the heat is a critical design consideration for future processor design.

Several studies propose dynamic thermal management schemes to control operating temperatures. When the temperatures exceed a threshold, energy-saving techniques, such as fetch toggling [3] and global clock gating [8], are invoked to cool the chip. However, to tackle the thermal challenge in deep sub-micron technologies, as pointed out in [10], operating temperature should be considered in the entire design flow. Traditionally, a processor architect only considers the performance factor during design space exploration. Since the thermal issue is the first-order concern for future processors, temperature should also be considered early in the design cycle. The chip temperature is affected by two factors, the power consumption of processor components (e.g., caches and ALUs) and the chip layout. Placing colder blocks around a hot block results in a lower temperature than putting hot blocks together. Therefore, a thermal-aware design should explore both architectural and physical design spaces simultaneously.

Several researchers have shown the importance of this joint exploration of architectural and physical design spaces from the performance perspective because wire delay has increasing impact on performance as the feature size continues to shrink. Cong et al.[6] first point out the need to consider both the IPC (instruction per cycle) and cycle time during architectural design exploration. They propose an architectural evaluation methodology to optimize performance in terms of billion instructions per second (BIPS). Ekpanypong et al. [7] propose a profile-guided microarchitectural floorplanner that optimizes IPC for a given clock frequency by inserting flip-flops in interconnections with large delay. This paper proposes a thermal-aware architectural floorplanning framework. To our knowledge, this work is the first to consider thermal effect during architectural floorplanning. The proposed framework allows an architect to maximize performance within the given temperature constraint by conducting efficient design space explorations that consider the interaction between the physical and architectural design.

The rest of the paper is organized as follows. Section 2 presents our thermal-aware micro-architectural floorplanning

*This work is supported in part by research grants from ROC National Science Council (NSC-93-2752-E-002-008-PAE, NSC-93-2220-E-002-001, MOE-93-EC-17-A-01-S1-031).

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.
ISLPED'05, August 8–10, 2005, San Diego, California, USA
Copyright 2005 ACM 1-59593-137-6/05/0008 ...$5.00.

framework. The details of our floorplanning methodology are described in Section 3. The experimental results are shown in Section 4. Section 5 concludes this paper.

2. OVERVIEW OF THERMAL-AWARE MICROARCHITECTURAL FLOORPLANNING FRAMEWORK

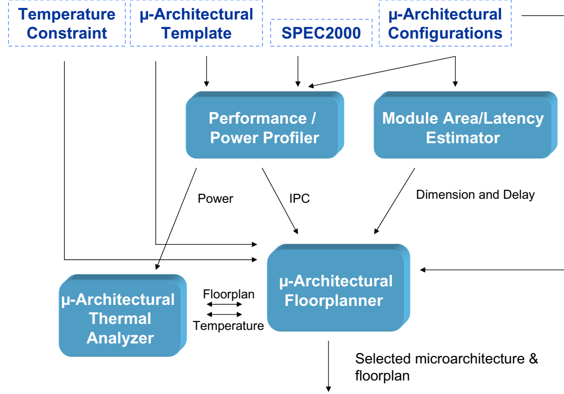

Figure 1: Framework Overview.

Figure 1 shows the flow of the proposed unified exploration framework of physical and architectural design with temperature consideration. There are four inputs to this framework: an micro-architectural template specifying the connectivity among functional blocks and the underlying pipelining architecture, a set of micro-architectural configurations (e.g., different cache sizes) that an architect would like to explore, target applications (SPEC2000) and the temperature constraint. The performance/power profiler generates IPC and power consumption of each micro-architectural configuration for the specified micro-architectural template and target applications. The area and latency of different modules are obtained through the module area/latency estimator. The micro-architectural thermal analyzer is employed to estimate the die temperature for a given floorplan based on the power consumption of the selected micro-architectural configuration. The main component of the proposed framework is the micro-architectural floorplanner that selects an micro-architectural configuration and generates the corresponding chip layout that maximizes performance while meeting the specified temperature constraint. We use existing tools for the performance/power profiler, module area/latency estimator and micro-architectural thermal analyzer. Section 4 lists the set of tools used in this paper. Our micro-architectural floorplanner adopts a novel thermal-aware floorplanning methodology. The design of our micro-architectural floorplanner is detailed below.

3. DESIGN OF THE MICROARCHITECTURAL FLOORPLANNER

The goal of the micro-architectural floorplanner is to select an micro-architectural configuration and produce the corresponding floorplan that satisfies the given temperature constraint while optimizing performance. Performance is measured as IPC/clock cycle time. Our floorplanner is based on simulated annealing (SA) [11] and uses B*-tree [4] as our floorplaning representation.

SA is a wildly-used *non-deterministic* algorithm for solving combinatorial optimization problems. Each iteration of SA is composed of three steps. Perturbation results in a new B*-tree through a set of operations (e.g., swapping two nodes). Therefore, after each perturbation, a packing procedure is invoked to compute the coordinates of modules which generate the corresponding floorplan of this new B*-tree. The quality of this floorplan is then evaluated based on a pre-defined cost function. The whole process is repeated until the SA termination condition is met.

The proposed thermal-aware floorplanner enhances the fundamental SA-based algorithm with three new features. First, we adopt an adaptive cost function; that is, the weight of the cost metric changes during the SA process. Second, instead of randomly choosing modules and operations during perturbation, we use a heuristic-based perturbation. Third, to facilitate micro-architectural configuration searching, we introduce a new type of perturbation - configuration selection. Below we detail each of the three new features.

Adaptive Cost Function

The cost function Φ used in our floorplanner is given by:

$$\Phi = \alpha \frac{CT_P}{IPC(c)} + \beta T, \qquad (1)$$

where CT_P is the estimated cycle time for the floorplan P, $IPC(c)$ is the estimated IPC of the configuration c, and T is the maximum die temperature of P. We estimate the interconnect length according to the half-perimeter measure. Similar to Cong et al. [6], we use IPEM [5] to estimate the interconnect delay, and perform static timing analysis at every iteration of SA to estimate the cycle time. The IPC of the current configuration is obtained through the performance profiler, and the maximum die temperature is generated through the thermal analyzer. We normalize both performance and temperature terms in the cost function. The parameter α is equal to one, while β is changed adaptively as followings:

1. $\beta = 0$, if $T \le T_{max}$.
2. $\beta = T - T_{max} + \epsilon$, if $T > T_{max}$.

, where T_{max} is the temperature constraint, and ϵ is a constant between 0 and 1.

Recall that our optimization goal is to maximize performance while satisfying the temperature constraint. Therefore, in the first case when the temperature of the current solution is below the temperature constraint, we focus on performance optimization by setting β to 0. In the second case, the temperature is higher than T_{max}, therefore, both performance and temperature factors should be considered in searching for the solutions. The temperature factor is given more weight as the difference between the temperature constraint and the current die temperature gets larger. Note that we include ϵ in the cost function to ensure that a solution that violates the constraint obtain much higher cost than a feasible one. The effectiveness of the proposed adaptive cost function requires careful tuning of the ϵ parameter. In this paper, we set the ϵ parameter to 0.25.

Heuristic-based Perturbation

Instead of randomly choosing modules for perturbation, we propose a heuristic-based perturbation approach. We introduce two new types of operations: Critical-Module-Swap and Hot-Cold-Mix. The Critical-Module-Swap operation swaps Module-X with a neighbor of Module-Y, where Module-X and Module-Y are both in the critical path. The Hot-Cold-Mix operation places the hottest module around the coldest one. When the temperature constraint is not violated, we focus on performance optimization. Therefore, the Critical-Module-Swap operation is considered during each perturbation to shorten the cycle time by placing modules in the critical path close to one another. When the temperature constraint is violated, the Hot-Cold-Mix operation is considered during each perturbation to achieve even thermal distribution thereby lowering the die temperature. Note that to support these two types of operations, we need to know the neighboring modules. We obtain this information during the packing process. The details of the neighboring module identification can be found in the technical version of this paper [14].

Configuration Selection

One way to find the optimal solution among all possible configurations is to individually obtain the best floorplan for each configuration. However, this process is very time consuming. In our work, we treat configuration selection as one of the perturbation operations. Cong et al.[6] also use a similar approach. They randomly choose a configuration, and then perform a small number of additional low-temperature moves on this configuration to decide to accept or reject it. To further improve the efficiency of configuration selection, instead of randomly selecting a configuration, we evaluate the cost of all configurations under the current layout, and choose the first three configurations in the increasing order of their costs as our configuration alternatives. The idea is that configurations with lower costs are more likely to lead to a better solution. Therefore, trying these configurations first could let the SA engine converge faster. However, with this configuration selection policy, those configurations with higher power density may not get a chance to be explored because it is harder to obtain a floorplan satisfying the temperature constraint. Therefore, in addition to low cost configurations, we also choose configurations that violate the temperature constraint as our configuration alternatives. Consequently, each configuration evaluation invokes a packing process thereby requiring more computation time per perturbation than the random approach. However, as shown later in Section 4, our approach still generates solutions more efficiently than the random approach since our SA engine converges faster.

4. EXPERIMENTAL RESULTS

The micro-architectural template used in our experiments is illustrated in Figure 2, and all 32 possible micro-architectural configurations are listed in Table 1. We vary the size of branch predictor (Bpred), load/store queue(LSQ), level one I-cache (I1 cache), level one D-cache (D1cache), and level two union cache (U2cache). We obtain the module area and delay data based on the information provided in [13] and [12]. The power/performance profiler is based on the Wattch simulator [1]. Wattch is an architecture-level simulator that

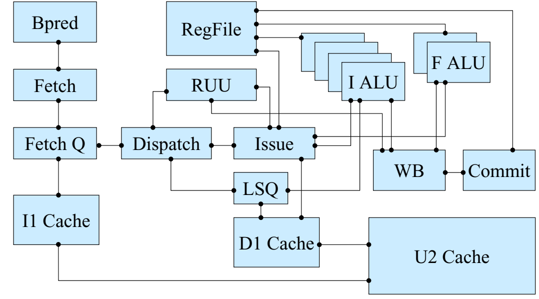

Figure 2: The Microarchitectural Template.

	Bpred	LSQ	I1cache	D1cache	U2cache		Bpred	LSQ	I1cache	D1cache	U2cache
1	256	16	32K	32K	128K	17	256	16	32K	32K	512K
2	2048	16	32K	32K	128K	18	2048	16	32K	32K	512K
3	256	128	32K	32K	128K	19	256	128	32K	32K	512K
4	2048	128	32K	32K	128K	20	2048	128	32K	32K	512K
5	256	16	64K	32K	128K	21	256	16	64K	32K	512K
6	2048	16	64K	32K	128K	22	2048	16	64K	32K	512K
7	256	128	64K	32K	128K	23	256	128	64K	32K	512K
8	2048	128	64K	32K	128K	24	2048	128	64K	32K	512K
9	256	16	32K	64K	128K	25	256	16	32K	64K	512K
10	2048	16	32K	64K	128K	26	2048	16	32K	64K	512K
11	256	128	32K	64K	128K	27	256	128	32K	64K	512K
12	2048	128	32K	64K	128K	28	2048	128	32K	64K	512K
13	256	16	64K	64K	128K	29	256	16	64K	64K	512K
14	2048	16	64K	64K	128K	30	2048	16	64K	64K	512K
15	256	128	64K	64K	128K	31	256	128	64K	64K	512K
16	2048	128	64K	64K	128K	32	2048	128	64K	64K	512K

8 Fetch queue entries, 4 ALU, and 2FLU are fixed.

Table 1: 32 Different Microarchitectural Configurations.

generates both cycle-accurate performance information and power consumption for each module. We perform simulations on 15 SPEC2000 benchmarks (gzip, gcc, bzip2, art, mesa, vpr, ammp, mgrid, equake, applu, swim, apsi, mcf, parser, and vortex) and use the arithmetic mean to obtain the IPC and power consumption for each configuration. We use Hotspot 2.0 [9] as the architectural thermal analyzer. The temperature constraint is 100^oC for the experimental results presented in this section.

To show the importance of thermal-aware floorplanning, we also implement a performance-driven floorplanner which uses the same SA engine as the proposed thermal-aware floorplanner but whose cost function contains only the performance factor. The comparison of these two floorplanners in both performance and temperature aspects are shown in Figure 3. Note that performance is measured as BIPS (billion instructions per second). Our thermal-aware floorplanner generates solutions with much lower temperatures than the performance-driven floorplanner without impact on performance. These results point out two things. First, different chip layout does have significant impact on the die temperature. Second, compared with the performance-driven floorplanner, our floorplanner successfully produces solutions that can satisfy the temperature constraint without performance degradation.

To demonstrate the effectiveness of our method (adaptive cost function combined with heuristic-based perturbation) for satisfying the temperature constraint compared with the traditional method (fixed cost function without heuristic-

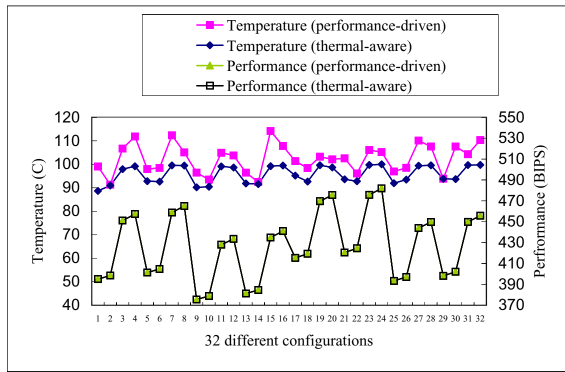

Figure 3: Performance and Temperature Comparison of Performance-driven and Thermal-aware Floorplanning.

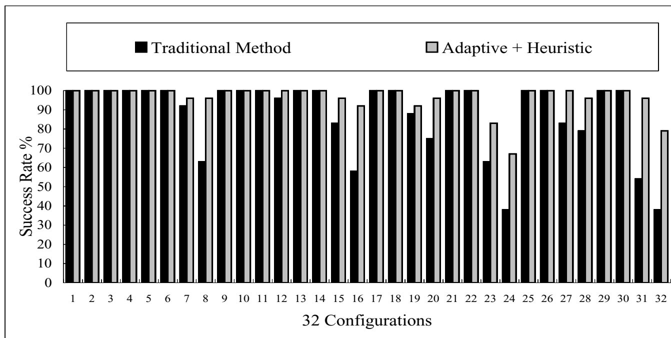

Figure 4: The Success Rate of Traditional Method vs. Adaptive + Heuristic for Each Configuration.

based perturbation), we show the success rate[1] of all 32 configurations. For the traditional method, we set both α and β to 1. Our method achieves higher success rate than the traditional method in the 12 out of 32 configurations. For example, in configuration #31, our method achieves about 96% success rate while the traditional method achieves only 54%.

To demonstrate the efficiency of our method in exploring combined design spaces, we show that with shorter running time, the solution quality obtained by our approach is the same as the bruteforce method which generates the best floorplan for each configuration individually. Table 2 lists the configuration selected. The performance (in terms of BIPS), and running time for the bruteforce method, the method used in [6] (random), and our approach (heuristic). Both performance and running time are normalized to the bruteforce method. We see that our approach finds the same solution in #24 as the bruteforce method with only 13% of its running time while the random method can not generate the optimal solution and is not as efficient as our approach (2.5 times the running time 0.33 compared to 0.13).

[1]Percentage of runs that satisfy the temperature constraint.

Method	Configuration	Performance	Running time
Bruteforce	#24	1	1
Random	#19	0.97	0.33
Heuristic	#24	1	0.13

Table 2: Solution Quality & Run Time Comparison

5. CONCLUSION

This paper presents a thermal-aware micro-architectural floorplanning framework that allows an architect to perform efficient design space exploration considering the interaction between the physical and architectural designs. The goal of this framework is to find an architecture configuration and the corresponding chip layout which maximizes performance while satisfying the temperature constraint. We adopt the simulated annealing floorplanning method with adaptive cost function and heuristic-guided perturbation. Our floorplanner is able to obtain significant thermal gains compared with the traditional performance-driven floorplanner without impact on performance. We are also able to search the huge combined solution spaces more effectively than a bruteforce approach.

6. REFERENCES

[1] D. Brooks, V. Tiwari, and M Martonosi, "Wattch: A Framework for Architectural-Level Power Analysis and Optimizations," *In Proceedings of the 27th International Symposium on Computer Architecture (ISCA)*, Vancouver, British Columbia, June 2000.

[2] The international technology roadmap for semiconductors(ITRS), 2003

[3] D. Brooks and M. Martonosi, "Dynamic Thermal Management for High-Performance Microprocessors," *In Proceedings of the 7th International Symposium on High Performance Computer Architecture (HPCA)*, pp.171-182, Jan. 2001.

[4] Y.-C. Chang, Y.-W. Chang, G.-M. Wu, and S.-W. Wu, "B*-trees: A New Representation for Non-Slicing Floorplans," *In Proceedings of the 37th Conference on Design Automation (DAC)*, pp. 458–463, June 2000.

[5] J. Cong and D. Pan, "Interconnect Estimation and Planning for Deep Submicron Design," *In Proceedings of the 36th Conference on Design Automation (DAC)*, pp. 507–510, June 1999.

[6] J. Cong, A. Jagannathan, G. Reinman, and M. Romesis, "Microarchitecture Evaluation with Physical Planning," *In Proceedings of the 36th Conference on Design Automation (DAC)*, pp. 32–35, June 2003.

[7] M. Ekpanyapong, J. R. Minz, T. Watewai, H.-H. S. Lee, and S. K. Lim, "Profile-Guided Microarchitectural Floorplanning for Deep Submicron Processor Design," *In Proceedings of the 36th Conference on Design Automation (DAC)*, pp. 634–639, June, 2004.

[8] S. Gunther, F. Binns, D. M. Canmean, and J. C. Hall, "Managing the Impact of Increasing Microprocessor Power Consumption," *Intel Technology Journal*, Q1 2001.

[9] Hotspot tool suit http://lava.cs.virginia.edu/HotSpot/

[10] W. Huang, M. R. Stan, K. Skadron, K. Sankaranarayanan, S. Ghosh, and S. Velusamy, "Compact Thermal Modeling for Temperature-Aware Design," *In Proceedings of the 36th Conference on Design Automation (DAC)*, pp. June 2004.

[11] S. Kirkpatrick, C. D. Gelatt, and M. P. Vecchi, "Optimization by Simulated Annealing," *Science*, vol. 220, no. 4598, pp.671–680, May, 1983.

[12] S. Wilton and N. Jouppi, "Cacti: An Enhanced Cache Access and Cycle Time Model," *IEEE Journal of Solid-State Circuits*, May 1996.

[13] S. Gupta, S. W. Keckler, and D. Burger, "Technology Independent Area and Delay Estimates for Microprocessor Building Blocks," *Technical Report 2000-05, Department of Computer Sciences, The University of Texas at Austin*, 2000.

[14] Y. W. Wu, C. L. Yang, P. H. Yuh, and Y. W. Chang, "Joint Exploration of Architectural and Physical Design Spaces with Thermal Consideration," *Technical Report 05-04, National Taiwan University*, 2005.