

Exploring the Thermal Impact on Manycore Processor Performance

Wei Huang^{1,4*}, Kevin Skadron¹, Sudhanva Gurumurthi¹, Robert J. Ribando², and Mircea R. Stan³

¹ Department of Computer Science, University of Virginia, Charlottesville, VA

² Department of Mechanical and Aerospace Engineering, University of Virginia, Charlottesville, VA

³ Department of Electrical and Computer Engineering, University of Virginia, Charlottesville, VA

⁴ IBM Austin Research Laboratory, Austin, TX

* Wei Huang, whuang@virginia.edu

Abstract

Performance of processors with many simple parallel cores is limited by the serial part of the workload, requiring an asymmetric core organization with one or more aggressive “primary” cores for better serial performance. A primary core introduces power-hungry microarchitectural structures and usually causes severe local hot spots. This paper explores the thermal impact on manycore processor architecture and evaluates its performance. Preliminary results show that thermal constraints reduce performance as expected, but also make performance almost insensitive to the complexity of the primary core across a diverse degrees of parallelism, which greatly reduces design complexity.

Keywords

Thermal impact, thermal constraint, manycore, processors, performance.

1. Introduction

CMOS semiconductor technology scaling and the diminishing return of extracting performance from traditional instruction-level parallelism (ILP) have caused a paradigm shift toward explicitly parallel processor design with many, possibly simple, processor cores. For example, Intel has announced an 80-core chip [1], NVIDIA has graphics chips with 240 parallel cores [2], and AMD's graphics chips have up to 800 streaming processing units [3]. On the other hand, the speedup of parallel computing is asymptotically limited by the serial fraction of the workload [4]. Therefore, single-thread serial performance is still crucial for a majority of applications. This favors an asymmetric (i.e. heterogeneous) manycore architecture, which consists of many basic computing cores handling massively parallel threads and one aggressive, often much more complex “primary” core dealing with the serial threads [5]. In addition, for general-purpose computing, it is important that the design can handle applications with a variable amount of parallelism. To achieve this, reconfigurable chip multiprocessors that can dynamically combine multiple simple cores into a primary core with variable complexities have been proposed [6, 7].

At the same time, non-ideal CMOS technology scaling also causes power density to increase from generation to generation [8]. With the shift toward manycore processors, it is likely that the common-case total power will also rise as technology scales. For example, the new Sun Niagara chip multiprocessor already consumes 250 Watts [9]. This is caused by the increased number of cores, the improved circuit delay (hence frequency),

the relatively constant die area required for yield, and the lack of supply voltage scaling. As a result, we are likely to face a serious thermal challenge in the manycore era. As a matter of fact, we are quickly approaching the limit of air cooling, which has been predicted to be difficult for significant improvement. For example, [10] states that the air cooling limit is around an average chip power density of 1.5W/mm². Even novel cooling techniques can only delay the onset of severe thermal difficulties by a few years.

Asymmetric manycores are particularly at a disadvantage from a thermal point of view, because the primary core usually creates local hot spots as a result of the extra power-hungry structures added to improve single-thread ILP. Thus it has at least the same, if not more, power density as the simpler cores. Previous work has shown that with the same power density, large cores are hotter than small cores [11] due to a thermal spatial low-pass filtering effect that reduces the equivalent thermal resistance of smaller cores. To make the situation even worse, speed boosting techniques, such as the “turbo mode” of Intel Nehalem processors [12], are used to further increase the single-core performance by increasing its frequency and supply voltage when other cores are idle, with the side effect of increasing its power density and temperature. Clearly, there is a tradeoff between the higher single-thread performance and the performance penalty caused by local hot spots in the primary core of an asymmetric architecture.

This paper explores the first-order impacts of thermal constraints on performance as well as the core organization by comparing with a first-order manycore performance analysis [13]. With a focus on asymmetric manycores, we find that *thermal constraints make performance almost insensitive to the primary core's complexity, and it is possible that dynamic core configurations may not be necessary in order to satisfy applications with a large diversity of parallelism.*

2. Related Work

There is some existing work considering manycore (or multicore) design from a thermal point of view. Li et al.[14] and Monchiero et al. [15] consider the thermal constraints in multicores at a more detailed microarchitecture level with comprehensive architecture simulations for multi-programmed and multi-threaded workloads, respectively. Huang et al. [11] look at a heat-spreading floorplanning approach to increase the power envelope of symmetric manycores without thermal violations. Donald et al. [16] and Chaparro et al. [17] investigate thermal management techniques for symmetric multi-

cores. None of these previous studies consider asymmetric or dynamic multicores and the impact of heterogeneity on temperature, power and performance. Most of them focus on multi-programmed workloads running on symmetric complex cores, whereas thread-level parallelism is not fully considered. The goal of this paper is to find the overall trends and provide high-level insights for temperature-aware manycore design that are tedious and time-consuming to explore using detailed microarchitectural simulations. In addition, we mainly focus on asymmetric manycores with thread-level parallelized workloads. Our observations in this paper are orthogonal to the existing temperature-aware multicore and manycore work.

There are also several existing studies regarding first-order performance analysis of manycores, but none considers the design implications of thermal constraints. Hill and Marty [13] provide guidance for performance analysis of symmetric and asymmetric manycores. Following that, the Hill-Marty work has been further improved with power and energy considerations [18, 19]. Furthermore, Loh [20] refines the performance model in [13] with the cost of “uncore” components, such as last-level caches and on-chip interconnects. Here, we add yet another important aspect of manycore design—the thermal constraints—to existing studies, and use the Hill-Marty model [13] as a base case for comparison. In addition, we discover the insensitivity of performance to the primary core’s complexity in the thermally-limited case. The thermal analysis also provides crucial design insights that are missed by other related studies.

3. Thermal Extension of Manycore Amdahl’s Law

In the computer architecture community, there is an important rule known as Amdahl’s Law [4], which states that more parallelism results in more performance, but the maximum performance is asymptotically bounded by the performance of the serial part. Full comprehension and consideration of Amdahl’s Law are crucial to successful computer architecture designs. In this section, we first list the assumptions that we adopt from the Hill-Marty manycore extension [13] to Amdahl’s Law. We also add further assumptions that are specific to the thermal aspects. We then show the resultant thermal extension of the manycore Amdahl’s Law.

3.1. General assumptions

We normalize the performance metric to that of a basic “unit core”. In this paper, a unit core is the fundamental processing unit that more complex cores are normalized to. We also assume a constant die area of 20mm×20mm, which is the typical size a lithographic reticle during the chip fabrication process [8]. As for the number of unit cores accommodated on a die, we choose a realistic configuration with $n=256$ unit cores as in the Hill-Marty model. A smaller number would possibly miss interesting tradeoffs that only exist in large number of cores, but the analysis in the paper also applies to other values of n . To increase the single-thread performance, especially that of the serial threads, we also need large cores with a complexity r times larger than unit cores. We call the more complex and large core in an asymmetric architecture a “primary” core. We assume the primary core area scales linearly with r . If the performance of a unit core is 1, we assume the performance of the

primary core is \sqrt{r} , according to the widely-accepted Pollack’s Rule [21]. The parallelized fraction of the workload is denoted by f , and hence the serial fraction is $1 - f$. Furthermore, like in [13], the “uncore” overhead from on-chip networks and low-level caches is not considered. With these assumptions, we can reach the following equations as in the Hill-Marty model [13].

For a symmetric architecture with only the unit cores:

$$\text{Speedup}_{\text{sym}}(f, n, r) = \frac{1}{(1-f) + \frac{f}{n}} \quad (1)$$

where 1 in the numerator represents the original execution time; $(1-f)$ in the denominator is the execution time of the serial part that cannot be improved by adding more simple cores; f/n is the execution time of the parallel part of the workload.

For a symmetric architecture with many homogeneous complex cores, each has a complexity of r unit cores and a speedup of \sqrt{r} :

$$\text{Speedup}_{\text{sym}}(f, n, r) = \frac{1}{\frac{1-f}{\sqrt{r}} + \frac{f \cdot r}{\sqrt{r \cdot n}}} \quad (2)$$

For an asymmetric architecture with $n - r$ unit cores and one primary core of a complexity r :

$$\text{Speedup}_{\text{asym}}(f, n, r) = \frac{1}{\frac{1-f}{\sqrt{r}} + \frac{f}{\sqrt{r+n-r}}} \quad (3)$$

3.2. Thermal-specific assumptions

The Hill-Marty model is simple but adequate for high-level performance analysis. Unfortunately, modern high-performance design is also limited by thermal constraints. An optimization neglecting thermal constraints will not yield realistic designs [14]. To add the thermal constraints, we make the following additional thermal-specific assumptions. All the specific values we use in the following assumptions are typical values that are derived from actual designs or existing studies. We also tried other reasonable values and found similar trends.

1. *Thermal design power (TDP)*. The baseline TDP we use in this paper is 256 Watts. This makes the power of each unit core 1 Watt. 256W is a high but plausible value for state-of-the-art multicore and manycore chips. For example, the new Sun Niagara chip multiprocessor consumes 250 Watts [9].
2. *Power of the primary core*. We assume that the primary core’s power consumption scales linearly with its complexity and area [21]. In other words, the average power densities of the basic unit cores and the primary core are the same. Because large and complex cores usually include additional power-hungry microarchitecture structures to increase ILP, this is actually a rather conservative assumption from a thermal point of view.
3. *Local high power densities*. As mentioned before, the primary cores usually add power-hungry structures. These structures have much higher local power density than the rest of the core. Our simulated data of a scaled Alpha 21364 (EV6) core at the 130nm technology show that the hot units, such as the integer register file, have power densities that are 3~4 times higher than the processor core’s

average power density (excluding lower-level caches). As technology continues to scale, the power density of local hot spots will get even worse. In this paper, we use “power density ratio”, pd_ratio , to describe the ratio of within-core hot spot power density to the core average power density, i.e.,

$$pd_ratio(r, n) = 1.0 + pd_ratio_0 \cdot (r - 1)/n \quad (4)$$

where pd_ratio_0 is set to a conservative value of 3.0. From Equation (4), we can see that a unit core ($r = 1$) has $pd_ratio = 1.0$, meaning its power is uniform. On the other hand, a primary core that occupies the entire die ($r = n$) has $pd_ratio = 4.0$, meaning the hottest unit within the core has a power density that is 4 times higher than the average core power density. In addition, based on our area estimation for the EV6 processor at 130nm technology, we further assume that 10% of the core area has higher local power densities that lead to hot spots (excluding lower-level caches), i.e. $a_factor=0.1$.

4. *Performance boosting techniques.* In multicores and manycores, if there are inactive cores, the performance of active cores can be boosted by reclaiming the unused power of the inactive cores and dynamically increase the supply voltage and operating frequency of the active cores. This technique has been adopted by the Intel Nehalem architecture [12]. In this paper, we assume similar techniques are used. In other words, the primary core’s supply voltage increases during the execution of serial threads when the parallel unit cores are idle. If more power is applied to the primary core, the primary core’s supply voltage and performance can be approximated by

$$Vdd_{new} = \sqrt[3]{\frac{Power_{new}}{Power_{old}}} \cdot Vdd_{old} \quad (5)$$

$$Perf_{new} = \sqrt[3]{\frac{Power_{new}}{Power_{old}}} \cdot Perf_{old} \quad (6)$$

This is because, to the first order, frequency increases linearly with supply voltage, and according to $P = CV^2f$, power increases cubically with respect of supply voltage (neglecting leakage power for simplicity). There should also be a limit of how high supply voltage can be. In this paper, we set the dynamic range of supply voltage between the nominal Vdd and 1.3Vdd—a conservative limit on what typical processes allow.

5. *The impact of hot spot size on temperature.* Previous studies [11, 22] indicate that the size of a hot unit plays an important role in its temperature. For the same power density, the temperature rise of a small unit can be much less than that of a large unit. This is because the lateral heat spreading within silicon is more predominant for small heat sources, resulting in lower equivalent thermal resistance from the hot spot to the ambient. Here, we adopt the model in [11] and express the thermal resistance reduction ratio (Rth_ratio) as follows:

$$Rth_ratio = \frac{1}{\sqrt{1 + (\frac{t}{4s})^2}} \quad (7)$$

where s is the size of the hot spot, and t is the silicon thickness from the die surface to the package surface, which corresponds to the isothermal surface defined in [11]. With this, the actual thermal resistance from the hot spot in the primary core to the isothermal surface can be written as

$$R_{th,actual} = R_{th,lumped} \cdot Rth_ratio = \frac{t \cdot Rth_ratio}{k_{si} A_{hotspot}} \quad (8)$$

where k_{si} is the thermal conductivity of silicon, and $A_{hotspot}$ is the hot spot area, which is 10% of the primary core area.

6. *Maximum die temperature and package thermal resistance.* We pick 85°C as the critical silicon temperature and set the ambient temperature to be 25°C. For a chip dissipating 256W with typical silicon and package configurations, that leads to a package convection thermal resistance about $R_{pack}=0.17K/W$, which is a reasonable value for advanced air cooling solutions [8].

3.3. Temperature-aware manycore performance model

With the above assumptions, we can derive a first-order temperature-aware manycore performance model.

With the presence of local hot spots when running serial threads in an asymmetric manycore, the actual power dissipated in the hot spot within the primary core can be calculated as

$$Power_{hotspot} = \frac{T_{max} - T_{ambient}}{R_{th,actual} + \frac{R_{pack}}{a_factor \cdot pd_factor}} \quad (9)$$

We can further calculate the actual power allowed to be dissipated by the primary core with thermal constraints as

$$Power_{primary} = \frac{Power_{hotspot}}{a_factor} \cdot \frac{1}{pd_ratio} \quad (10)$$

When executing serial threads, only the primary core is active, making the primary core power, which is also the total instantaneous chip power in this case, less than TDP. On the other hand, the primary core consumes more power than it does when all cores are active. This is possible because the elevated hot spot temperature is counteracted by the lower total chip power, keeping the maximum chip temperature within thermal constraints.

By dissipating more power on the primary core in the serial mode of execution, we can improve the serial performance by using performance boosting techniques. The performance gain can be calculated similar to Equation (6):

$$Perf_{primary_serial} = \sqrt[3]{\frac{Power_{primary_serial}}{Power_{primary_parallel}}} \cdot Perf_{primary_parallel} \quad (11)$$

where $Perf_{primary_parallel}$ is \sqrt{r} . Of course, this performance gain is bounded by the maximum supply voltage that is allowed, which is 1.3Vdd in our analysis.

Therefore, the performance of an asymmetric manycore with thermal constraints is:

$$Speedup_{th,asym}(f, n, r) = \frac{1}{\frac{1-f}{Perf_{primary_serial}} + \frac{f}{\sqrt{r+n-r}}} \quad (12)$$

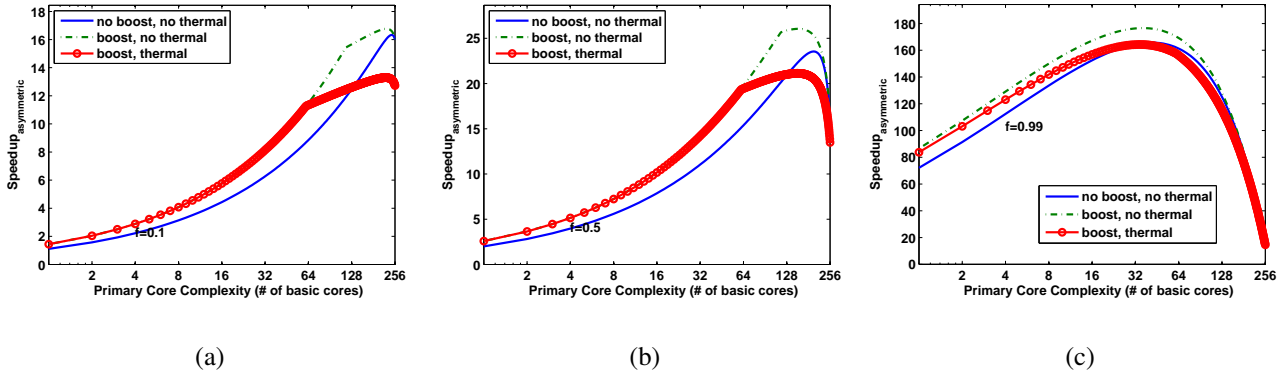


Figure 1. Speedup vs. primary core complexity for asymmetric manycores. (a) $f=0.1$; (b) $f=0.5$; (c) $f=0.99$

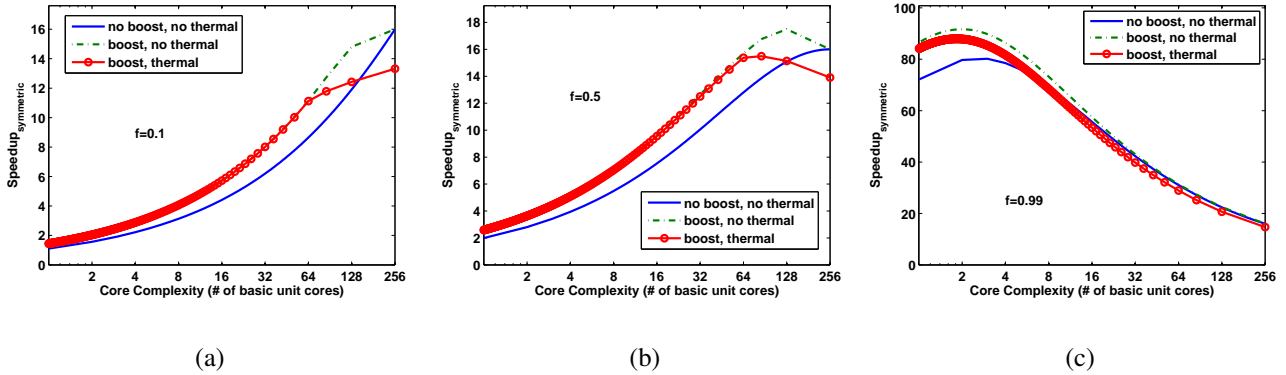


Figure 2. Speedup vs. homogeneous core complexity for symmetric manycores. (a) $f=0.1$; (b) $f=0.5$; (c) $f=0.99$

Similarly, we can also reach the temperature-aware performance of a symmetric manycore with multiple homogeneous complex cores:

$$\text{Speedup}_{\text{th,sym}}(f, n, r) = \frac{1}{\frac{1-f}{\text{Perf}_{\text{primary_serial}}} + \frac{f \cdot r}{\sqrt{r \cdot n}}} \quad (13)$$

Equations (12) and (13) are derived from Amdahl's Law and reasonable thermal-related assumptions based on actual design data. Therefore, the model should be able to catch first-order thermal impact on manycore performance. Detailed architectural simulations are also needed to verify this first-order model and will be our future work.

4. Results and Discussions

With Equations (12) and (13), we are equipped to analyze the temperature-aware speedup of both asymmetric and symmetric manycores. In particular, we are interested in aspects where adding thermal constraints change the results in the Hill-Marty model. Detailed results and discussions are presented in this section.

4.1. Asymmetric is still better

Without thermal constraints, conventional wisdom such as Amdahl's Law, tells us that the asymmetric manycores should always have more speedup than the symmetric ones. However, asymmetric manycores introduce more power non-uniformity, hence more severe local hot spots and more performance penalty. On the other hand, symmetric manycores might be

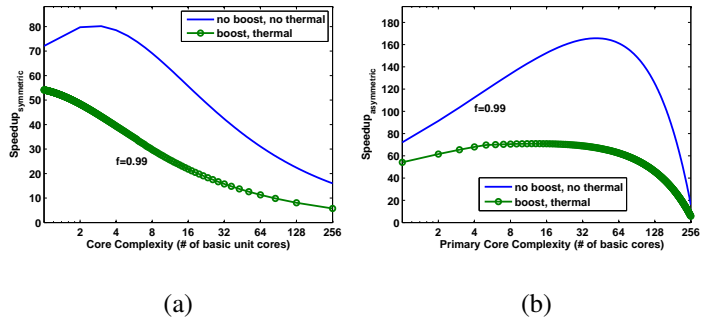


Figure 3. Speedup vs. complexity of the primary core(s) at $f=0.99$, with higher package thermal resistance $R_{\text{pack}}=5\text{K/W}$. (a) Symmetric manycore (multiple primary cores with the same complexity); (b) Asymmetric manycore (one primary core and many simple basic unit cores).

designed with only simpler cores that are less thermally constrained, but have lower serial performance. Therefore, it is interesting to see whether there is a turning point where thermal constraints are so predominant that asymmetric manycores perform worse than symmetric ones.

Fig. 1 and Fig. 2 plot the speedup as a function of the primary core's complexity (i.e. number of equivalent basic unit cores) with $f=0.1, 0.5$, and 0.99 for asymmetric and symmetric manycores, respectively. f indicates the fraction of parallelized threads in the workloads or applications. There are three curves in each figure. The thin solid line is the Hill-Marty model (no

boost, no thermal); the thin dotted line is the Hill-Marty model with performance boosting techniques (boost, no thermal); the circled thick line is our model with boosting techniques and thermal constraints (boost, thermal). The sudden change in the slopes of the curves is caused by the relaxation of the 1.3V_{dd} constraint that is not needed for more complex primary cores with dynamic voltage scaling.

As we can see, the performance boosting techniques improve the speedup significantly, especially for more parallelized workloads (e.g. $f=0.5$ and 0.99). However, with thermal considerations, the improvement is reduced for poorly parallelized workloads (e.g. $f=0.1$ and 0.5).

Comparing asymmetric to symmetric manycores, it is apparent that, for workloads that are highly parallelized ($f=0.99$, Fig. 1(c) and Fig. 2(c)), the asymmetric manycores have much higher speedup than symmetric manycores (160 vs. 87). Even for workloads with $f=0.5$, the advantage of asymmetric manycores is still noticeable (21 vs. 16). For poorly parallelized workloads (e.g. $f=0.1$), the two are almost the same. This tells us that 1) it is crucial to always keep Amdahl's Law in mind, because improvement in serial performance is important to achieve further speedup, especially for highly parallelized applications; 2) the performance penalty caused by the thermal constraints is not significant enough to fully cancel the performance benefit of an asymmetric manycore.

To be complete, we also consider manycore designs with poor cooling package, where thermal constraints are more severe. For example, in some mobile applications, chips are packaged without heatsinks, resulting in a much higher package thermal resistance. We use $R_{pack}=5K/W$, and reach the results in Fig. 3(a) and (b), for symmetric and asymmetric manycores, respectively. We only show the results with $f=0.99$. As we can see, with a poor thermal package, the allowed power dissipation is much less than the original TDP, leading to lower temperature-aware speedup. In addition, the difference in speedup between the asymmetric and symmetric manycores becomes small (the circled thick lines). Also notice that the temperature-aware speedup in this case is lower than the Hill-Marty model, especially for asymmetric manycores (see Fig. 3(b)).

Because asymmetric manycores are usually better than symmetric manycores even with thermal constraints, we focus on asymmetric manycores in the following analysis.

4.2. Performance loss due to thermal constraints

From Fig. 1, we can also look at the performance penalty induced by thermal constraints. For asymmetric manycores, we can see that thermal constraints push down the *boost, no-thermal* curve significantly, especially for poorly parallelized workloads (up to 20% for $f=0.1$ and a good cooling solution). For these workloads, thermal constraints make the speedup less than what the original Hill-Marty model predicts, even with the performance boosting techniques. For highly parallelized workloads (e.g. $f=0.99$ or higher), because the optimal complexity of the primary core is less, the performance penalty caused by the local hot spots in the primary core becomes lower.

In order to reclaim the thermally-induced performance loss for poorly parallelized workloads that do not require high parallel speedup, there are a couple of possible solutions: 1) build

a smaller chip with less unit cores to bring down the total power and hence the hot spot temperature; 2) use a low-power process for the parallel unit cores (such as high- V_{th} transistors) to have lower total power and hence lower hot spot temperature in the primary core. These will be interesting future work.

Also notice in Fig. 1, if we choose the optimal primary core complexity that has the maximum speedup in the Hill-Marty model, although it is sub-optimal on the temperature-aware curve, the resulting actual speedup is still close to the maximum speedup in the temperature-aware case.

4.3. Shift in optimal primary core complexity

One of the most important decisions to make in the design of an asymmetric manycore is the optimal complexity of the primary core. From Fig. 1, we can see that the temperature-aware optimal primary core complexity shifts to the left (less complexity) with respect to the Hill-Marty model (no boost, no thermal) and the boosted Hill-Marty model (boosted, no thermal). The shift is most dramatic for poorly parallelized workloads. For example, with $f=0.5$, the shift with respect to the Hill-Marty model is from 198 unit cores to 143 unit cores, whereas for $f=0.99$, the shift is from 47 unit cores to 30 unit cores (Notice the horizontal axis is in logarithmic scale). This can be explained by the fact that a more complicated core has more severe local hot spots. The performance penalty caused by these hot spots makes such cores less attractive, and it is possible that higher speedup can be achieved with a less complicated core.

4.4. Relaxed dynamic core combination

The most interesting observation from Fig. 1 is the flatness of the temperature-aware speedup curves around the optimal primary core complexity. This means that a large difference in the primary core complexity makes only a small difference in speedup. For example, with $f=0.5$, a primary core with complexity of 64 unit cores has a normalized speedup of 19.4, a primary core with complexity of 220 unit cores also has a normalized speedup of 19.4, whereas the maximum normalized speedup of 21.3 happens at the complexity of around 150 unit cores. This is significantly different from the Hill-Marty model, where the speedup is very sensitive to the primary core's complexity. Similar results apply to other values of f .

This phenomenon can be explained as follows. As the primary core gets more complicated, its serial performance also increases. However, due to the more severe hot spot with a more complicated primary core, the thermally induced performance penalty also increases. These two factors counteract each other over a wide range of primary core complexity, resulting in a flat speedup curve.

Fig. 4 illustrates the degree of flatness for different f . Here, we plot the range of primary core complexity where the speedup is greater than or equal to 95% of the maximum speedup at each f (from 0 to 1.0). On the y-axis is the complexity of the primary core; f is on the x-axis. There are two sets of curves, one set corresponds to the Hill-Marty model (Fig. 4(a)), the other set corresponds to the temperature-aware model (Fig. 4(b)). Within each set, there are three curves—top, middle and bottom. The middle curve shows the optimal primary core complexity for each f in each model. The top curve shows the complexity of a more complicated primary core that achieves 95% of the

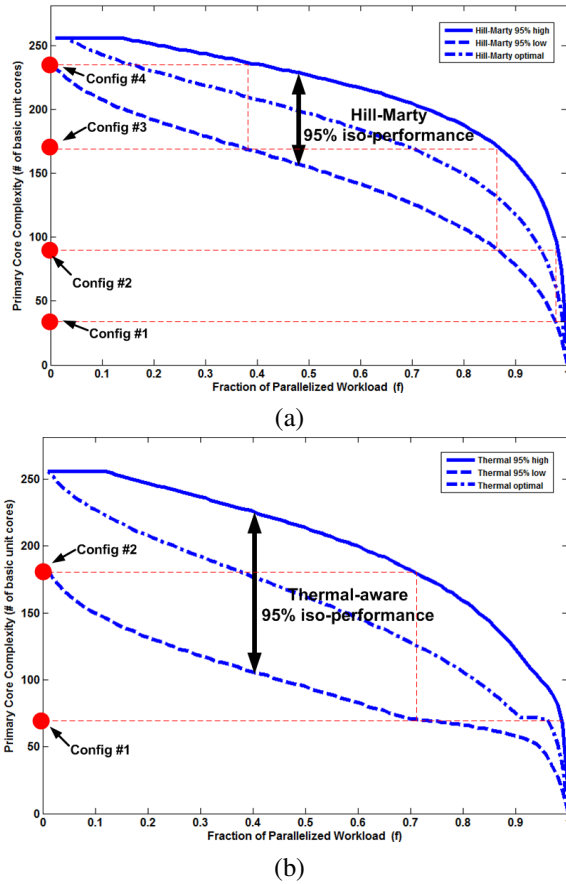


Figure 4. Range of primary core complexity that achieves 95% iso-performance of the optimal speedup in an asymmetric manycore. (a) The Hill-Marty model. (b) Our temperature-aware model. In the temperature-aware analysis, only two primary core configurations are enough to cover a software diversity of $f=0\sim 0.99$, whereas the Hill-Marty model shows four dynamic configurations are needed. If 90% performance is allowed, dynamic core combination becomes unnecessary.

maximum speedup (i.e. the 95% point on the right side of the optimal point on the speedup vs. complexity curve in Fig. 1). The bottom curve shows the complexity of a less complicated primary core that also achieves 95% of the maximum speedup (i.e. the 95% point on the left side of the optimal point on the speedup vs. complexity curve). The width of the two bands in Fig. 4(a) and (b) can be viewed as an indicator of the flatness of the speedup-complexity curves in Fig. 1.

As can be seen, for the Hill-Marty model, the band is narrow for very small f , indicating a very complicated primary core is needed for poorly parallelized workloads. The band grows wider as f increases, indicating more flexibility in choosing the optimal primary core complexity for a relatively large f . As f approaches 1.0, the band becomes narrow again, indicating a symmetric-like architecture where homogeneous simplest basic cores are preferred.

On the other hand, the width of the band is different when thermal constraints are considered. For very small f , because

of the aforementioned flatness from thermally-induced performance loss with large cores, the band is wider than the Hill-Marty model from the beginning, and gets gradually narrower as f approaches 1.0.

One concern in general-purpose manycore design is to have the processor be able to handle applications with all possible degrees of parallelism (i.e. different values of f). To achieve this, techniques such as core fusion [6], core federation [7], or dynamic manycores [13] have been proposed to dynamically combine multiple cores into one primary core depending on the workload. These approaches add significant design complexities and overhead. It is also a non-trivial work to find the number of levels of complexity that is needed to deal with all possible workloads.

However, when thermal constraints are considered, the requirement of dynamic combination of multiple cores is greatly relaxed. For example, in Fig. 4(b), for a primary core with a complexity of 180 unit cores, one can handle f from 0 to 0.68 with at most 5% performance loss comparing to the optimal speedup. With another primary core configuration of 70 unit cores, the design can cover f from 0 to 0.99. In contrast, if thermal constraints are not considered and only the original Hill-Marty model is used (Fig. 4(a)), one would put much more effort on designing an asymmetric manycore that should be able to dynamically combine 230 unit cores into one primary core to deal with applications with $f=0\sim 0.38$; 170 unit cores for $f=0.38\sim 0.86$; 90 unit cores for $f=0.86\sim 0.97$; 30 unit cores for $f=0.97\sim 0.99$, and so on, with at least 95% of the maximum performance.

In fact, if we relax the performance requirement from 95% to 90%, the temperature-aware analysis shows that we can get away with only one primary core configuration at $r=110$ unit cores for $f=0\sim 0.99$, without dynamic core combination at all. This is assuming we use a simple workload model, in which workloads are evenly distributed across f . On the other hand, if thermal constraints are not considered, we still need two dynamic primary core configurations (210 and 100 unit cores) to cover the same range of parallelism.

5. Conclusions and Future Work

In this paper, we add thermal constraints to the manycore Amdahl's Law model, and find that asymmetric manycores are still better than symmetric manycores in term of performance, although the thermal constraints degrade the performance of the primary core. We also find that the optimal primary core complexity becomes less due to the impact of thermal constraints. A more interesting observation is that the optimal speedup can be achieved with a wide range of primary core complexity, and this is true for applications with different fractions of parallelized threads. The implication of this observation is that the design complexity for dynamic combination of multiple cores to deal with different levels of parallelism is greatly relaxed.

Because this work is based on a very simple workload and performance model, one should look at the predicted trends rather than specific data points by this model, and apply such temperature-aware analysis early in the design process of future manycore chips in the thermally-limited era. More detailed simulations and experiments are required in order to reach an

actual optimal design point.

Interesting directions of future work include detailed simulations to calibrate the findings in this paper; more extensive exploration of the design space with wider range of design variables such as TDP, cooling solutions, microarchitecture options for the primary core and the simple cores; detailed modeling of the overhead of dynamic core combination and the uncore components; and power and energy efficiency of the temperature-aware manycores.

Acknowledgement

This work is funded by NSF grants CRI-0551630 and EHS-0509245, as well as a grant from Intel Research.

References

1. S. R. Vangal et al. An 80-tile sub-100-w TeraFLOPS processor in 65-nm CMOS. *IEEE Journal of Solid-State Circuits*, 43(1):29–41, January 2008.
2. http://www.nvidia.com/object/product_tesla_c1060_us.html.
3. <http://ati.amd.com/products/radeonhd4800/specs.html>.
4. G. M. Amdahl. Validity of the single-processor approach to achieving large-scale computing capabilities. In *Proc. of AFIPS*, 1967.
5. R. Kumar, D. M. Tullsen, N. P. Jouppi, and P. Ranganathan. Heterogeneous chip multiprocessing. *IEEE Computer*, 38(11):32–38, November 2005.
6. E. Ipek, M. Kirman, N. Kirman, and J.F. Martinez. Core fusion: Accommodating software diversity in chip multiprocessors. In *Proc. of ISCA*, 2007.
7. D. Tarjan, M. Boyer, and K. Skadron. Federation: Repurposing scalar cores for out-of-order instruction issue. In *Proc. of DAC*, 2008.
8. The International Technology Roadmap for Semiconductors (ITRS), 2007.
9. G. K. Konstadinidis et al. Architecture and physical implementation of a third generation 65 nm, 16 core, 32 thread chip-multithreading sparc processor. *IEEE Journal of Solid-State Circuits*, 44(1):7–17, January 2009.
10. P. Zhou, J. Hom, G. Upadhy, K. Goodson, and M. Munch. Electro-kinetic microchannel cooling system for desktop computers. In *Proc. of SEMI-THERM*, 2004.
11. W. Huang, M. Stan, K. Sankaranarayanan, R. Ribando, and K. Skadron. Many-core design from a thermal perspective. In *Proc. of DAC*, 2008.
12. Intel Turbo Boost Technology in Intel Core Microarchitecture (Nehalem) Based Processors. (<http://download.intel.com/design/processor/applnots/320354.pdf>) November 2008.
13. M. D. Hill and M. R. Marty. Amdahl's law in the multicore era. *IEEE Computer*, 41(7):33–38, July 2008.
14. Y. Li, B. Lee, D. Brooks, Z. Hu, and K. Skadron. CMP design space exploration subject to physical constraints. In *Proc. of HPCA*, 2006.
15. M. Monchiero, R. Canal, and A. Gonzalez. Design space exploration for multicore architectures: A power/performance/thermal view. In *Proc. of ICS*, June 2006.
16. J. Donald and M. Martonosi. Techniques for multicore thermal management: Classification and new exploration. In *Proc. of ISCA*, June 2006.
17. P. Chaparro, J. Gonzalez, G. Magklis, Q. Cai, and A. Gonzalez. Understanding the thermal implications of multi-core architectures. *IEEE Transactions on Parallel and Distributed Systems*, 18(8):1055–65, 2007.
18. D. H. Woo and H. S. Lee. Extending amdahl's law for energy-efficient computing in the many-core era. *IEEE Computer*, 41(12):24–31, December 2008.
19. S. Cho and R. G. Melhem. Corollaries to amdahl's law for energy. *IEEE Computer Architecture Letters*, 7(1):25–28, January-June 2008.
20. G. H. Loh. The cost of uncore in throughput-oriented many-core processors. In *Proc. of Workshop on Architectures and Languages for Troughput Applications (ALTA)*, 2008.
21. S. Borkar. Thousand core chips—a technology perspective. In *Proc. of DAC*, 2007.
22. K. Etessam-Yazdani, H. F. Hamann, and M. Asheghi. Impact of power granularity on chip thermal modeling. In *Proc. of 10th Intersociety Conference on Thermal and Thermomechanical Phenomena in Electronics Systems (ITHERM)*, June 2006.