Architectural Implications of Spatial Thermal Filtering

Karthik Sankaranarayanan
Intel Labs
Hillsboro, OR, 97124 USA
karthik.sankaranarayanan@intel.com

Brett H. Meyer
Department of Electrical and Computer Engineering
McGill University
Montreal, QC, H3A 2A7 Canada
brett.meyer@mcgill.ca

Wei Huang[‡], Robert Ribando[*], Hossein Haj-Hariri[*], Mircea R. Stan[†], and Kevin Skadron[‡]
Departments of [*]Mech. and Aerospace Engineering, [†]Elec. and Comp. Engineering, and [‡]Computer Science
University of Virginia
Charlottesville, VA 22904 USA
{wh6p, rjr, hh2b, mircea}@virginia.edu, skadron@cs.virginia.edu

***Abstract*—Process technology scaling, lagging supply voltage scaling, and the resulting exponential increase in power density, have made temperature a first-class design constraint in today's microprocessors. Prior work has shown that the silicon substrate acts as a spatial low-pass filter for temperature. This phenomenon, *spatial thermal filtering*, has clear implications for thermal management: depending on the size of dissipators, either design-time strategies, such as dividing and distributing functionality spatially, or runtime strategies, such as isolating functionality temporally (duty cycling), may be the most effective way to control peak temperature. To assist designers with such trade-offs, we have performed extensive analysis and simulation to evaluate the extent and effect of spatial filtering on thermal management in a number of microarchitecture design scenarios.**

We begin our exploration of spatial filtering with an analytical study of the heat conduction problem, followed by a series of studies to validate the effect and extent of spatial filtering under realistic system assumptions. In particular, we investigate the effect of power dissipator size, location, and aspect ratio in the context of high-performance computing. We then extend these experiments with two microarchitectural studies. First, we perform a study of spatial filtering in many-core architectures. Our results show that as cores shrink, the granularity of effective thermal management increases to the point that even turning cores on and off has a limited effect on peak temperature. Second, we investigate spatial filtering in caches. We discover that despite the size and aspect ratio of cache lines, pathological code behavior can heat caches to undesirable levels, accelerating wear-out.

I. Introduction

As semiconductor manufacturing processes scale to smaller and smaller feature sizes, operating temperatures in silicon are emerging as a critical design constraint. Higher power densities—which are due in large part to parameters such as V_{DD} and V_{Th} not scaling ideally—create several problems requiring costly solutions. First, since higher power densities translate to greater silicon heating, cooling costs, which increase super-linearly with temperature, are rising. Maintaining low system temperatures is important, since hotter systems are less reliable: many important wear-out induced permanent failure mechanisms are exponentially dependent on temperature [1]. Including redundancy or additional design margin to mitigate these temperature-dependent system reliability problems is also expensive, requiring considerable design effort and increases in per-chip costs.

Multicore architectures, which naturally parallelize and distribute computer system functionality across the die, give designers unprecedented freedom to cleverly organize systems and the policies that determine how they are used to address a variety of emerging design challenges. As temperature is closely related to power density, distributing functionality to reduce power density can be an effective way of addressing design concerns such as system lifetime. Such strategies are important to high-performance computing because of the pressing need to reduce package and cooling costs.

In this context, there are two basic approaches to system-level temperature management:

1) *distributing functionality in time at runtime*, for example, through the selection of scheduling policies that duty-cycle hot components or functional units; and,
2) *distributing functionality in space at design-time*, for example, through the selection of system components and their location in the system floorplan, separating hot cores or functional units with cool components.

The task of the computer system designer is, given the

Most of this work was completed while both Karthik Sankaranarayanan and Wei Huang were with the University of Virginia. Wei is now with IBM Research–Austin in Austin, TX, USA.

expected system cooling solution, to determine (1) the most effective combination of design-time and runtime techniques to distribute functionality and minimize peak system temperature, and (2) the level of design abstraction at which these optimizations should be applied. For example, should *functional units* be carefully scheduled (or placed in the layout) to reduce peak temperature, or, alternatively, is it more effective to turn *entire cores* on and off (or judiciously place them relative to one another)?

To support designers in this crucial design activity, we have performed extensive analysis and simulation to determine what factors contribute to how heat is dissipated in multicore computer systems, and as a result, how these systems should be architected to ensure low operating temperatures.

A. Spatial Filtering

It has been previously observed that the relative physical dimensions of cores and the silicon within which they are implemented play a critical role in how heat is transferred between different regions of the silicon system [2], [3]. In particular, if power-dissipating silicon blocks are small enough, they do not cause hot spots. In effect, the silicon filters out these spatial high frequencies, disproportionately reducing the silicon heating of small power dissipators. This phenomenon, known as *spatial filtering*, is illustrated in Figure 1.

Figure 1(a) shows a silicon die with two square power dissipating blocks. Both blocks have the same power density, but the area of the block on the left is four times that of the one on the right. Although the power densities are the same, it can be seen that the larger block on the left is significantly hotter than the smaller block on the right. This relationship between block-size and peak temperature is illustrated in Figure 1(b). It is a classic Bode plot frequency response of a low-pass filter with the spatial frequency (1/m) plotted on the x-axis and the peak temperature on the y-axis. Both of the axes are on a logarithmic scale. It shows a clear cut-off frequency beyond which the temperature falls rapidly.

Spatial filtering suggests that

1) when blocks of functionality are smaller than the cut-off size, the local thermal effect of their power dissipation is negligible;
2) when blocks of functionality are near the cut-off size, both duty-cycling its use as well as sub-dividing and spreading its power dissipation over several disjoint blocks (and thereby crossing the cut-off size) will significantly reduce temperature; and,
3) when blocks of functionality are much larger than the cut-off size, only duty-cycling is expected to result in significant temperature reductions, unless the blocks can be sub-divided to such an extent that the size of the subdivisions falls below the cut-off size.

B. Architectural Implications of Spatial Filtering

Understanding the architectural implications of spatial filtering is critical to cost-effective temperature-aware design. In this paper, we have conducted extensive analysis and simulation to explore the extent, as well as the implications, of spatial filtering in silicon. We begin by exploring spatial filtering in the context of an analytical solution to the three dimensional heat conduction problem in silicon, and observe that when power dissipating blocks are small, lateral heat conduction dominates vertical heat conduction, reducing the impact of hot spots. This makes substantial temperature reduction possible if blocks can be shrunk across the cut-off size, *e.g.*, through manufacturing process scaling or by replacing larger processors with a collection of smaller processors.

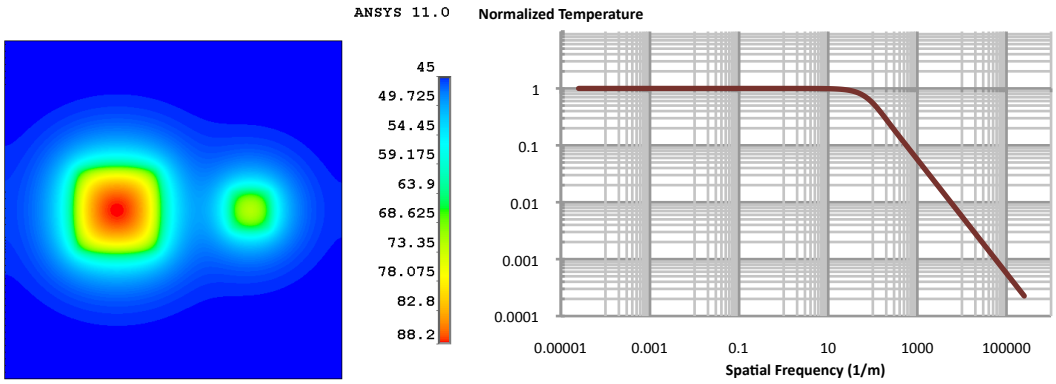
We then conduct a series of empirical studies to validate this observation for high-end computing platforms. First, we look at the impact of the size, location and aspect ratio of a single power dissipator in silicon. In general, we observe that the heat spreaders and sinks in high-end systems conduct heat laterally to compensate for local imbalances in power density. This weakens the relative effect of spatial filtering, but as a result, it is actually possible to modestly reduce the peak temperature of systems with large blocks by shrinking components, contrary to the theoretical result (which predicts that changes that don't cross the threshold of the low-pass filter are ineffective).

Second, we extend these experiments to tiled multicore architectures, and explore the impact of the number of cores, die thickness, and the ratio of the area devoted to processors to that devoted to on-chip cache. We also explore the effect of dynamic thermal management policies operating at different levels of granularity, *e.g.*, disabling a microprocessor functional unit or an entire core. We observe that as the size of cores shrinks in high-performance systems, system-level thermal management must operate at higher granularity to be effective. As functional units, and, ultimately cores, shrink below the cut-off size, changes in the use of these structures are spatially filtered, with the result that duty-cycling entire cores or groups of spatially contiguous cores eventually becomes the only effective approach.

In our third study, we apply our findings on spatial filtering to explore the extent to which malicious code may create hot spots in and ultimately damage on-chip caches. We observe that in performance-optimized caches, which partition the SRAM array to reduce access delay, repeatedly accessing a small number of contiguous cache lines can significantly increase the temperature of those cache lines, resulting in accelerated aging.

II. Related Work

Other research has examined spatial filtering from the viewpoint of accurate thermal *modeling*. Etessam-Yazdani *et al.* [4] perform a two-dimensional Fast Fourier Transform (FFT) analysis to determine the spatial cut-off frequency beyond which high power density has less impact on temperature. Our prior research has also explored the effect of spatial filtering on temperature modeling [5], [6], with a particular focus on the relationship between power dissipator size, aspect ratio and modeling accuracy. This paper extends the prior research by investigating the impact of spatial filtering on temperature *management*. Unlike prior approaches, we begin by studying the thermal conduction problem directly, applying sinusoidal



(a) (b)

Fig. 1: Illustration of spatial thermal filtering. (a) Though each block has the same power density, different peak temperatures result. (b) Low-pass filter "Bode plot" frequency response. Cut-off occurs at a normalized half-size of 1.

power density distributions to find an analytical solution in *three* dimensions. We then investigate the effect of spatial filtering in a variety of scenarios, including mircoarchitectural studies, in order to explore its implications for design- and run-time techniques that reduce system temperature. Among the important factors not investigated in the prior work, we explores in detail the effect of the package on spatial filtering.

Our prior research has also explored the effect of spatial filtering on the thermal efficiency of many-core processors [2]. This paper extends our prior research by considering the effect of the package, chip thickness, and L2 area on spatial filtering. Unlike our prior work, this paper considers the broader implications of spatial filtering on thermal management, such as the appropriate granularity for thermal management policies as a function of dissipator size, and sensor accuracy.

Dadvar and Skadron [7] have previously raised concerns about the possible security risks involved in software-controlled thermal management. Ku *et al.*'s research [8] have investigated techniques for reducing the power density of caches by keeping the live lines as far apart from each other as possible. In effect, they exploit spatial filtering by increasing the spatial frequency. Kong *et al.* [9] have investigated the effect of malicious code on instruction caches, and present methods for protecting cores accordingly. In this paper, we apply our findings on spatial filtering to analyze the likelihood that pathological code could damage data caches. By exploring the relationship between spatial thermal filtering and cache heating, we provide a framework to reason about such thermal management schemes as appear in the literature.

III. An Analytical Approach

The core of thermal filtering is the relationship between heat conduction in the lateral direction along the die and the vertical direction through the die, heat spreader, and heat sink. In order to understand this better, we first approach the problem from an analytical perspective, solving the three-dimensional heat conduction problem in silicon using simplified boundary conditions. Because the geometry and boundary conditions of an actual chip and package are quite complex, we subsequently perform detailed Finite Element Model (FEM) simulations of a realistic chip and package configuration to validate our analysis of spatial filtering for high-end computing platforms with sophisticated cooling solutions.

A. Analytical Solution of a Simplified Heat Conduction Problem

Figure 2 illustrates a simplified version of the heat conduction that occurs in the lateral and vertical directions of a chip package. It shows a single layer of silicon with thickness l convectively cooled at the top surface ($z = l$) by a heat transfer coefficient h to an ambient temperature at $0°C$. Power is dissipated at the bottom surface ($z = 0$) with a peak power density q. We have observed in the past that the secondary heat conduction path through the package and circuit board does not contribute significantly to the cooling of high-performance computers [10]. We therefore assume in our analytical model that the secondary path is insulated, and that all thermal conduction occurs through the interface at the top surface.

In order to capture the spatial thermal behaviour in the frequency domain, we let the power density vary as a *spatial sinusoid* with spatial frequencies in the x and y directions. In this context, small dissipators have higher spatial frequency, while large power dissipators have lower spatial frequency.

The silicon block is assumed to extend infinitely in the lateral (x and y) directions, and we set the spatial frequencies in the x and y directions to be equal to $\frac{1}{4a}$ and $\frac{1}{4b}$ respectively. To model such infinite lateral dimensions, since the supplied power density is periodic in space, it is sufficient to consider just a single period along each lateral axis ($x = -2a$ to $x = +2a$ and $y = -2b$ to $y = +2b$), as shown in the figure. The boundary conditions can then be repeated periodically in each lateral axis by requiring that the values at the boundaries be equal, *e.g.*, requiring that the temperatures at $x = -2a$ and $x = +2a$ be the same.

In order to determine the steady state temperature distribution $T(x, y, z)$ for the simplified geometry described above, we must solve the steady state heat conduction equation [11]

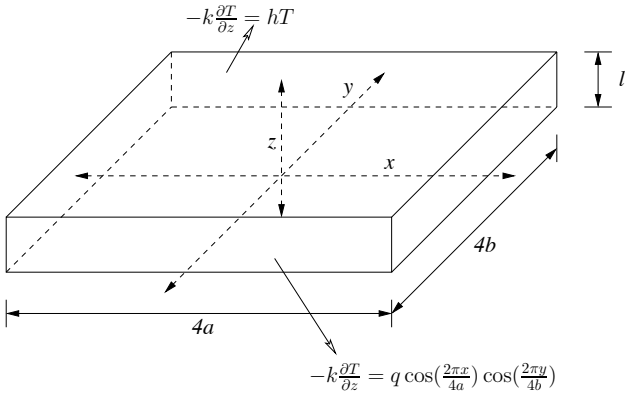


Fig. 2: A simplified three-dimensional heat conduction problem. Heat is generated at the bottom surface, and removed at the top surface.

below in three dimensions:

$$\nabla^2 T = 0 \quad (1)$$
$$(-2a \le x \le 2a)$$
$$(-2b \le y \le 2b)$$
$$(0 \le z \le l)$$

If k is the thermal conductivity of silicon, Fourier's law [11] states that *power density* $= -k\nabla T$. The power density cooled by convection is equal to $h(T - T_{amb})$ where T_{amb} is the ambient temperature (which simplifies to hT for an ambient of $0°C$). Moreover, the periodicity of the supplied power density indicates that the temperature and power density at $x = -2a$ ($y = -2b$) and $x = +2a$ ($y = +2b$) are equal. Thus, when the supplied sinusoidal power density is $q\cos(\frac{2\pi x}{4a})\cos(\frac{2\pi y}{4b})$, the boundary conditions for our problem are:

$$-k\frac{\partial T}{\partial z}|_{z=0} = q\cos(\frac{2\pi x}{4a})\cos(\frac{2\pi y}{4b}) \quad (2a)$$
$$-k\frac{\partial T}{\partial z}|_{z=l} = hT \quad (2b)$$
$$T|_{x=-2a} = T|_{x=+2a} \quad (2c)$$
$$-k\frac{\partial T}{\partial x}|_{x=-2a} = -k\frac{\partial T}{\partial x}|_{x=+2a} \quad (2d)$$
$$T|_{y=-2b} = T|_{y=+2b} \quad (2e)$$
$$-k\frac{\partial T}{\partial y}|_{y=-2b} = -k\frac{\partial T}{\partial y}|_{y=+2b} \quad (2f)$$

The analytical solution of this boundary value problem can be obtained using classic PDE-solving techniques [11]–[13]. Due to space considerations, we provide only the closed-form solution of the function T here, in Equation (3). Please refer to the appendix for the detailed derivation.

Several important observations can be made from the analytical solution. First, we note that, for given spatial frequencies $\frac{1}{4a}$ and $\frac{1}{4b}$, the peak temperature T_{peak} occurs at $(0, 0, 0)$, *i.e.*, at the center of the sinusoidal power dissipator[1]. Second, the absolute maximum peak temperature (for all spatial frequencies) occurs when $a = b = \infty$, *i.e.*, when there is uniform power dissipation. Let us call this temperature T^{∞}_{peak}. $T^{\infty}_{peak} = \frac{ql}{k} + \frac{q}{h}$ [11].

We can define the normalized peak temperature T^{norm}_{peak} as $\frac{T_{peak}}{T^{\infty}_{peak}} = \frac{T(0,0,0)}{\frac{ql}{k}+\frac{q}{h}}$. It is this T^{norm}_{peak}, whose value ranges between 0 and 1, that has been plotted against the reciprocal of spatial frequency in Figure 1(b). For very large spatial frequencies ($\lambda_c l \gg 1$), T^{norm}_{peak} can be calculated from Equation (3) as $\frac{1}{1+\frac{k}{lh}}\frac{1}{\lambda_c l}$. Setting this expression to the maximum value of $T^{norm}_{peak} = 1$, the cut-off frequency point is given by the relation $\lambda_c l = \frac{1}{1+\frac{k}{lh}}$. From the expression for T^{norm}_{peak} we observe that at very large frequencies, the gain of the low-pass filter falls by 10 dB per decade.

Our analysis reveals a trade-off with implications for thermal management. On the one hand, Figure 1(b) illustrates the exponential relationship between the size of a power dissipator and its peak temperature. In this case, spatial filtering suggests that when the size of the power dissipating block is close to the cut-off frequency, reducing component size can have a significant benefit over duty-cycling because of the exponential relationship between size and temperature, as opposed to the linear relationship between activity and temperature. On the other hand, Equation (3) shows the linear relationship between the power density q and peak temperature. In this case, even when the block is so large that even its partitions are much larger than the cut-off frequency, reducing the power density might be a better choice since the benefit from size reduction becomes negligible.

We also observe in Equation (3) that, similar to the spatial frequency, the *aspect ratio*, *i.e.*, the ratio between the spatial frequencies in the x and y directions, also has an exponential relationship with the peak temperature. This relationship implies that *tall and skinny* power dissipators have a thermal advantage over *short and stout* ones.

B. Spatial Filtering in Practice

In the context of spatial filtering, there are two possible thermal management schemes:

1) distributing functionality in time, reducing the temperature of a hot block by reducing its time-average computational activity and therefore its power density; and,
2) distributing functionality in space, manipulating the size of the power dissipating block (*e.g.*, by partitioning the block and separating the partitions by a safe distance) and hence exploiting this spatial filtering effect to reduce peak temperature.

To begin exploring trade-offs between these approaches in real systems, we relaxed the assumptions of the analytical model and examined spatial filtering in a series of experiments using a commercially available FEM solver, ANSYS 12.1 [14]. First, we repeated the analytical experiment and explored the impact of sinusoidal power dissipator size in finite silicon systems. Next, we extended this experiment to explore the impact of power dissipators of various aspect ratios.

[1]Considering peak temperature is sufficient for our purposes, as a thermal emergency is defined as an event in which the temperature anywhere in the system exceeds some pre-defined limit.

$$T(x,y,z) = \frac{ql}{k}\frac{1}{\lambda_c l}\left[\frac{(k\lambda_c+h)e^{\lambda_c(l-z)}+(k\lambda_c-h)e^{-\lambda_c(l-z)}}{(k\lambda_c+h)e^{\lambda_c l}-(k\lambda_c-h)e^{-\lambda_c l}}\right]\cos(\lambda_a x)\cos(\lambda_b y) \quad (3)$$
$$\textit{where, } \lambda_a = \frac{2\pi}{4a}, \lambda_b = \frac{2\pi}{4b} \textit{ and } \lambda_c = \sqrt{\lambda_a^2+\lambda_b^2}$$

1) Impact of Size: In these experiments, we relax several important assumptions of the analytical solution. The analytical model in the previous section assumed a single vertical conduction layer and infinite lateral dimensions. However, realistic packages include multiple vertical layers and finite lateral dimensions. We relax the assumptions of the analytical model in three steps. First, we consider a finite geometry with insulated lateral extremities (*i.e.*, no heat is conducted out of the sides of the die). Next, we include the effect of multiple package layers but without any lateral spreading beyond the lateral dimensions of the die (*i.e.*, all the vertical layers are of the same lateral dimensions as the die). Finally, we include lateral spreading beyond the die across the heat spreader and the heat sink, as both are typically larger than the die.

For the first step, we consider a 10 mm x 10 mm silicon die that is 1 mm thick, whose lateral surfaces are insulated. The top surface is cooled by convection to an ambient temperature of $0°C$ with a heat transfer co-efficient equivalent to a 0.1 $\frac{K}{W}$ thermal resistance (*i.e.*, for a 100 mm^2 area, it is $0.1\frac{W}{mm^2K}$). At the bottom surface of the die, we supply a sinusoidal power density with an amplitude of $1\frac{W}{mm^2}$ and equal spatial frequency in the x and y directions (*i.e.*, in the terminology of the previous section, $a = b$; alternatively, the aspect ratio is 1). The frequency of the sinusoid is varied and the temperature at the center of the bottom surface is measured. The maximum center temperature for this experiment occurs when the supplied power density is uniform (*i.e.,* the frequency of the sinusoid is zero) and its value is 20 degrees higher than the ambient. We call this experiment *1layer* to denote the single layer of silicon.

Next, we repeat the same experiment with the die attached to three other layers—a layer of Thermal Interface Material (TIM) followed by a layer of copper (the heat spreader), and then a second layer of copper (the heat sink). As before, all the layers are insulated at their extremities. To separate the effect of lateral spreading beyond the die shadow from the presence of multiple layers, we restrict the lateral dimensions of each of the layers to 10 mm x 10 mm. The vertical thickness of each layer is set in such a manner that the *proportion* of the thicknesses is similar to what is seen in a typical package and that the total *equivalent thickness* in silicon terms (*i.e.*, the sum of the thicknesses of the individual layers weighted by the ratios of their thermal conductivities to that of silicon) is still 1 mm. As a result, the maximum center temperature is still $20°C$. The actual thicknesses of the silicon, TIM, spreader and heat sink layers are 0.0571 mm, 0.0076 mm, 0.3810 mm and 2.6286 mm respectively. The respective thermal conductivities are 100, 4, 400 and 400 $\frac{W}{mK}$. The convection thermal resistance at the top surface remains the same as before and the spatial frequency of the power dissipator is varied as before. We call this experiment *4layers-equal* to denote the equal lateral size of the four layers.

Finally, to examine the effect of spreading beyond the die shadow, and to model a typical package in which the heat spreader and heat sink are larger than the die, we extend the *4layers-equal* configuration to make the lateral sizes of the heat spreader and the heat sink layers to be 20 mm x 20 mm and 40 mm x 40 mm respectively. We call this configuration *4layers-spread*. The convection thermal resistance remains the same as before. It is to be noted that the maximum center temperature in this case will be lower than the $20°C$ observed in the previous cases because of the lateral spreading beyond the die shadow.

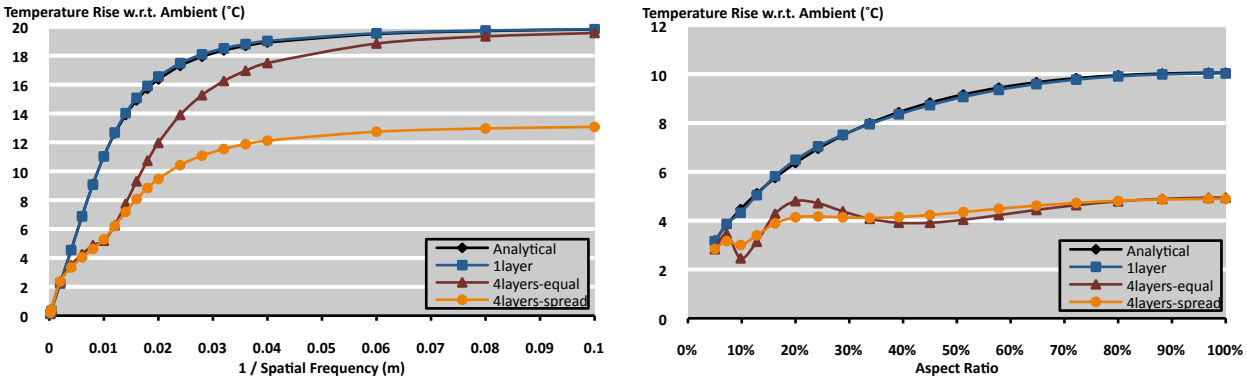
All the above experiments are run under ANSYS 12.1 with a lateral grid size of 100 x 100 in the silicon layer. The lateral dimensions of the grid cells are the same in the other layers as well. For the *1layer* configuration, the silicon die is divided vertically into ten layers. For the other two configurations, the die is modeled as three layers, the TIM as a single layer and the spreader as four layers. For the *4layers-equal* scenario, the sink is modeled as thirteen layers while for the *4layers-spread* case, it is modeled as four layers. Figure 3(a) shows the results of these experiments. For each configuration, it plots the temperature at the center of the dissipator surface (the location of the peak temperature, as noted above) as a function of the reciprocal of its spatial frequency (dissipator size). For validation, it also plots the temperature at the origin computed by the analytical model, Equation (3) (*Analytical* in the graph).

First, it can be seen from the graph that in spite of the approximations (infinite boundaries and periodicity at the lateral extremeties), the analytical model captures the heat conduction in a single layer very well, practically matching the *1layer* curve at all points. Point-by-point, *analytical* differs from *1layer* by just 4% on average.

However, when considering more realistic systems with multiple layers, the similarity between the analytical and simulated models breaks down, to the advantage of system designers. Like the *1layer* configuration, the *4layers-equal* configuration clearly displays the steep exponential relationship between dissipator size and temperature as suggested by the analytical equation, though the relationship between dissipator size and temperature increase isn't quite as strong as predicted by the analytical model. As a result, *analytical* differs from *4layer-equal* by 22% on average.

4layers-spread, on the other hand, shows a reduced exponential, indicating that less filtering is occurring; the difference between *analytical* and *4layer-spread* is 34% on average. This is the result of heat spreading beyond the die shadow; lateral spreading in the spreader and sink clearly play a crucial role in determining the extent of spatial thermal filtering.

Furthermore, in contrast to what is predicted by the analytical model, in both cases (*4layer-equal* and *4layer-spread*), scaling large components in the presence of heat spreaders and sinks provides modest benefits. As a result, a wider



(a) (b)

Fig. 3: Results of FEM simulation. (a) shows the impact of power dissipator size and (b) shows the effect of aspect ratio. Note that in both cases, *Analytical* and *1layer* are nearly coincident.

variety of systems are expected to benefit from design-time thermal management strategies that partition and distribute power dissipators.

C. Impact of Aspect Ratio

We next study the effect of dissipator aspect ratio on temperature. Understanding this relationship is essential for understanding the thermal behaviour of high aspect ratio microarchitectural sub-blocks such as cache lines.

In these experiments, the aspect ratio is defined as the ratio between the spatial frequencies in the x and y directions. In order to keep the dissipator area roughly constant (while still supplying a sinusoidal power density), the product ab is held constant at 5 mm^2, while the aspect ratio is varied from 1:20 to 1:1. This dissipator size corresponds to 0.009 m on the 1/Spatial Frequency scale of Figure 3(a). A small dissipator was selected in order to allow us to explore a wide variety of aspect ratios within the boundaries of the previously defined silicon system.

Figure 3(b) plots the relationship of aspect ratio and temperature for the above configurations. Again, we observe in Figure 3(b) that the analytical model approximates the behaviour of *1layer* very closely; point-by-point, *analytical* differs from *1layer* by less than 0.5% on average. We furthermore observe that while aspect ratio also has a clear exponential relationship with temperature, for the small dissipator under consideration, this exponential is shallow, and nearly linear under both *4layer* scenarios. We observe that even at an aspect ratio of 1, the dissipator size is small enough that there is sufficient lateral spreading in copper, even just under the die shadow (as in *4layer-equal*) to mitigate any temperature increase. For this reason, in this case *4layers-spread* and *4layers-equal* are nearly coincident, differing by only 6% on average. While the exponential is expected to be stronger for larger dissipators, it is obvious yet again that heat spreading is dampening the spatial filtering that occurs in silicon alone.

D. Summary

We have observed that when considering systems with a single power dissipator, dissipator size and aspect ratio have an exponential relationship with system temperature. Furthermore, we have observed that the presence or absence of a sophisticated cooling solution including a heat spreader and sink substantially affects how temperature changes when these parameters change. However, in high-end computers, reducing the size of large components modestly reduces system temperature, contrary our analytically derived result.

IV. Microarchitectural Examples

Next we explore the effect of spatial thermal filtering in a variety of detailed microarchitectural studies. To investigate how our observations in the previous section may be applied to real systems, we first perform experiments on tiled many-core processors. Second, we investigate whether cache lines, high aspect ratio sub-blocks within caches, can become hot spots due to pathological code behaviour.

A. Many-Core Processors

1) Thermal Benefit of Spatial Filtering: We have observed that as the size of a power dissipator shrinks, temperature is sharply reduced. It follows that separating cores from each other with caches could substantially reduce temperature. Not only do caches have lower power densities than cores, reducing the size of contiguous power dissipators increases the spatial frequency and takes advantage of spatial thermal filtering. We conducted a series of experiments investigating the temperature of tiled many-core processors organized in the form of a checkerboard, illustrated in Figure 4(a). In the figure, darkly shaded blocks are processors while the lightly shaded blocks are caches. When organized in this way, the system exploits spatial filtering by maximizing the spatial frequency of the power density distribution. We specifically investigate the effect of spatial filtering on temperature as a function of (a) the number of cores, (b) and core orientation and occupancy (the ratio of on-chip cores to on-chip cache), and (c) die thickness.

Number of Cores. Assuming that the silicon die is arranged in a checkerboard as shown in Figure 4(a), we vary the number of cores by varying the size of the checkerboard (from 2 x 2 to 20 x 20) and study its effect on peak temperature. To isolate the thermal effect of spatial filtering from other circuit and microarchitectural factors affecting temperature we assume that as the number of cores increases, the power density remains constant, as may possibly be achieved by replacing complex cores with simpler cores, or through manufacturing scaling. In practice, increasing the number of cores by reducing core complexity may reduce power density. On the other hand, increasing the number of cores through transistor scaling alone is likely to increase power density. Since the thickness of the die does not scale with the feature size for reasons of mechanical strength, it is assumed to be constant as well. Later, we examine the effect of this assumption by performing a sensitivity study that varyies the die thickness.

We assume a silicon die 16 mm x 16 mm x 0.15 mm in size. In the base case (denoted *regular-50* and illustrated in Figure 4(a)), cores occupy 50% of the total die area; the rest of the die is devoted to L2 cache. The rest of the package is composed of a TIM $20\mu m$ thick and having the same lateral dimensions as the die, a copper heat spreader of size 3 cm x 3 cm x 1 mm and a copper heat sink of size 6 cm x 6 cm x 6.9 mm. The heat sink is cooled by convection with a co-efficient of heat transfer equivalent to a 0.1 $\frac{K}{W}$ thermal resistance. The thermal conductivities of the materials are as in Section III-B. The uniform power density on the cores is set to be 1 $\frac{W}{mm^2}$, while that on the L2 cache banks is set to be 0.1 $\frac{W}{mm^2}$. We also explore the case of infinite cores as a limit study. When the number of cores is infinite, the power density on the entire die is uniform with a value equal to the average of the power density on the cores and that on the cache banks, weighted by their respective area occupancies (50% in this case).

We performed a steady state thermal analysis of this setup using ANSYS 11.0 FEM solver. We used a lateral grid size of 48 x 48 (due to software license restriction on the number of nodes). The lateral dimensions of the grid cells in the other layers are the same as in silicon. The vertical modeling resolution is identical to that of the *4layers-spread* configuration in Section III-B.

regular-50 in Figure 5(a) plots the relationship between the number of cores and peak temperature rise over the ambient for the base case. Both of the axes are logarithmic (except for the infinity point on the x-axis). Since the number of cores is inversely related to the size of a core, it is similar in sense to spatial frequency. Hence, the plot is analogous to a Bode plot.

Compared with the analytical model we derived above, the spatial filtering response is quite shallow for *regular-50*: temperature reduction is flatter earlier, and far less total temperature reduction is possible. Moving from 2 to 32 cores reduces the peak temperature only 7% in this case; in the limit, peak temperature can be reduced by 17%. Heat spreading in copper clearly smoothes out the response of the low-pass filter.

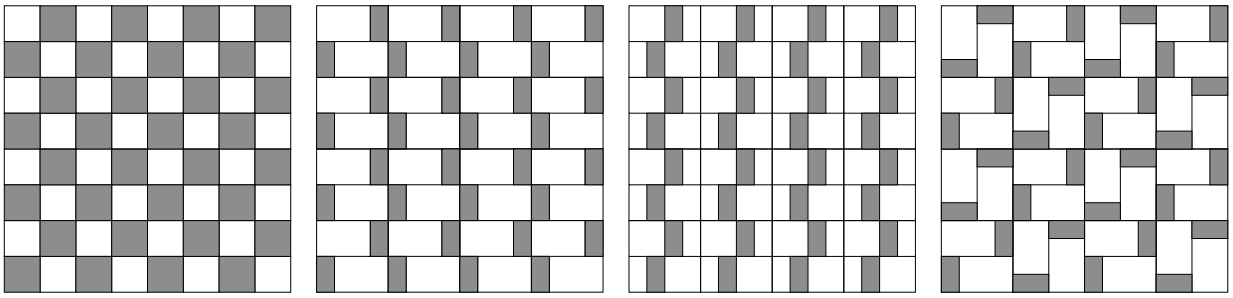
Core occupancy. We next conduct experiments to explore the effect of spatial filtering on systems with different core occupancy, since different systems may have different ratios of area devoted to core and cache. When the die occupancy is different from 50%, alternative checkerboard configurations are possible. Figure 4 illustrates a few such configurations for an occupancy of 25%. While we do not explicitly model the effect of a 2:1 aspect ratio on performance, we do not expect that modest changes in aspect ratio have a significant effect, especially in the context of increasing core count by trading high-speed, high-complexity cores (designed for single-threaded performance) for low-speed, low-complexity cores (designed for high throughput). *regular-25* (Figure 4(b)) is a derivative of *regular-50*. *alt-25* (Figure 4(c)) exposes the effect of chip boundaries by moving cores away from chip edges and corners. *rotated-25* (Figure 4(d)) additionally exposes the effect of core orientation (when cores are not square) by rotating some cores by 90°. We also explore a die occupancy of 75%; *regular-75* is the complement of *regular-25*. Likewise, *rotated-75* is the complement of *rotated-25*. We do not examine an *alt-75* configuration; this would divide some cores core, placing each half on either side of a cache bank, imposing a prohibitive performance penalty. The peak temperature for these arrangements are presented in Figure 5(a).

We observe that die occupancy influences the peak temperature more strongly than any other variable we've considered so far, with changes in occupancy resulting in the most significant observed changes in temperature for a fixed number of cores. As occupancy increases, so does power dissipation and therefore peak temperature. Moreover, when the die occupancy is lower, the spatial duty cycle is lower, too. For example, when cores are separated by cache, from a power perspective two small high power density areas are separated by a large low power density area. The low-pass behaviour of spatial filtering therefore further reduces peak temperature when occupancy is low. On the other hand, the duty cycle is higher for higher die occupancy. For these reasons, we observe at least three distinct lines (one for each occupancy level) in Figure 5(a).

In Figure 5(a)), the temperature difference between occupancy levels is far greater than the difference due to changes in the number of cores or core orientation. On average, increasing occupancy from 50% to 75% increases peak temperature by 43% under the *regular* floorplan. Likewise, decreasing occupancy from 50% to 25% decreases peak temperature by 42% under the *regular* floorplan. On the other hand, when occupancy is fixed at either 25 or 75%, peak temperature changes on average less than 1% when the floorplan is changed. Clearly, placing cores in the corners of the chip, where less lateral heat conduction is possible, has little negative effect in the presence of a heat spreader and sink.

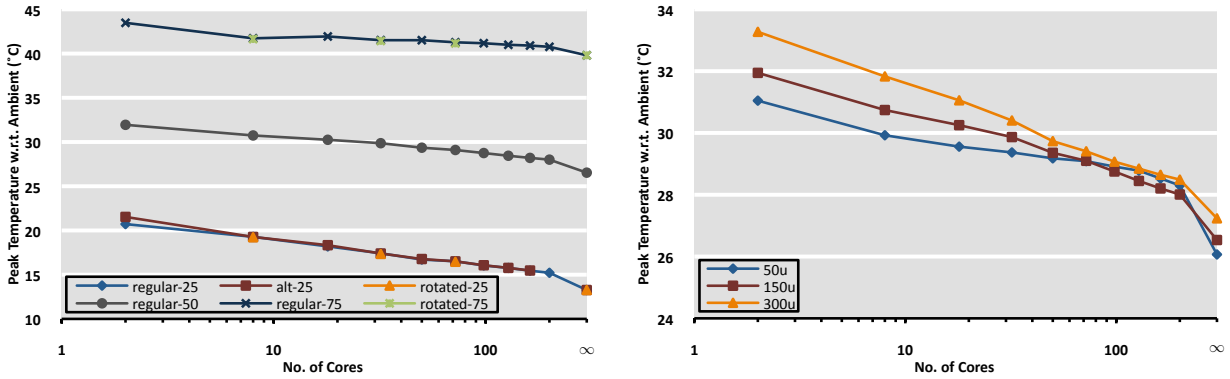
Die thickness. We next explore the effect of the die thickness on spatial filtering in both desktop and mobile configurations. Figure 5(b) plots the peak temperature for variable core counts using die thicknesses of $50\mu m$, $150\mu m$ (equivalent to *regular-50*), and $300\mu m$.

We observe that the previously observed trend in spatial filtering, *i.e.*, a shallow response, remain the same across different die thicknesses. Increasing die thickness to $300\mu m$ on average decreases temperature by less than 1%, while decreasing die thickness to $50\mu m$ on average increases temperature by 2%. As before, the presence of a heat spreader and sink



(a) regular-50 (b) regular-25 (c) alt-25 (d) rotated-25

Fig. 4: Illustration of the several checkerboard configurations studied. Darkly shaded areas are cores and the unshaded areas are lower level cache banks. "-xx" indicates the percentage of die area occupied by the cores. Each figure above shows a many-core die with 32 cores.



(a) Effect of core count, occupancy and orientation (b) Effect of die thickness

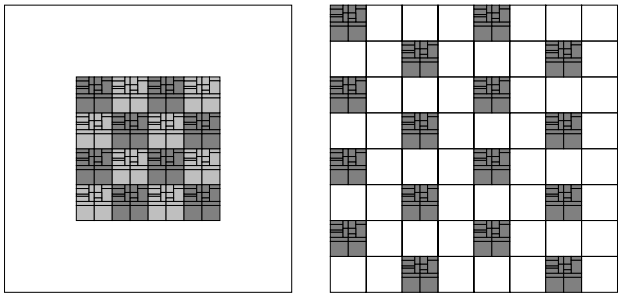
Fig. 5: Results of the checkerboard experiments. Note the overlap of the curves *rotated-25 and regular-25* (*rotated-75/regular-75*) in (a). *alt-25 and regular-25* are also coincident in (a).

minimizes the effect of changes in lateral heat conduction or vertical thermal resistance as die thickness changes.

Other experiments. In addition to the exploration of the above-mentioned parameters, we conduct experiments to determine the sensitivity of peak temperature to cache bank power density and the convection heat transfer coefficient. The effect of these parameters on peak temperature tracks the trends observed above, with a weak filtering response in systems with sophisticated cooling solutions.

2) Local vs. *Global Thermal Management:* Spatial filtering has clear implications for dynamic thermal management. We have observed that in some cases the location and size of power dissipators has a significant influence on peak temperature. It follows that when dynamically duty-cycling components the size and location of the selected component may also significantly influence peak temperature. We have therefore performed experiments investigating the relationship between core size and the opportunity for thermal management at different management granularities.

Experimental setup. We simulate two different scenarios marking the two extremes of core arrangement with respect to spatial filtering. In the first, denoted *center*, all the cores are arranged close to each other at the center of the die with the second level cache surrounding them. The shaded areas denote the cores while the unshaded areas denote the L2 cache (the



(a) Center (b) Checker

Fig. 6: Illustration of the two different many-core scenarios simulated. The unshaded areas are L2 cache. The shaded areas are cores resembling the Alpha 21364.

cores are shaded in alternating light and dark shades for better visibility). In the second, denoted *checker*, the cores and the L2 cache blocks are arranged in a checkerboard-like fashion similar to the previous section. In both cases, each core is modeled as an Alpha 21364 as in [15], and the L2 cache is assumed to occupy 75% of the total die area, all in a 130 nm process technology.

As in the previous section, we vary the number of cores, keeping the power density in each core (and hence the total

power on the entire die) constant. In order to examine the effectiveness of DTM, out of all the n cores, we select a single core that is closest to the center of the die and manage its temperature by turning off computational activity in regions of varying sizes, and observe the resulting change in the steady state temperature. Since we are interested in the best-case DTM potential of a region, we only study the change in steady state temperature resulting from its deactivation. If turning a region off does not reduce the peak temperature in the steady state case, it is unlikely that turning the same region off will provide any benefits in the transient case.

The regions are turned off at three levels of granularity: locally, at the functional unit level; semi-locally, at the sub-domain level; and globally, at the core level. At the functional unit level, each of the 15 architectural units are turned off separately, independently of each other. The unit that provides the greatest temperature reduction is chosen as the candidate for comparison against other levels of granularity. At the sub-domain level each sub-domain is also turned off independently of each other and the best-case peak temperature reduction is chosen. We consider four non-overlapping sub-domains:

- *Fetch engine*: I-cache, I-TLB, branch predictor and decode logic.
- *Integer engine*: Issue queue, register file and execution units.
- *FP engine*: Issue queue, register file and execution units.
- *Load-store engine*: Load-store ordering queue, D-cache and D-TLB.

At the core level, we turn off a single core closest to the center of the die. We do not consider coarser granularity (such as groups of cores) here; this is the subject of future work.

We experiment with the SPEC2000 benchmark suite [16] and determine steady state temperatures using a combination of the SimpleScalar performance model [17], Wattch power model [18] and HotSpot 4.1 thermal model [6]. We simulate each benchmark for an interval of 500 million instructions, identified using the SimPoint [19] tool. All the cores are assumed to run the same benchmark and hence the power numbers are replicated across them. The thermal model parameters are set to the defaults of HotSpot 4.1 except the die size, which is set to 12.4 mm x 12.4 mm, and the convection resistance of the package, which is set to $0.75\frac{K}{W}$. We use the grid-based model of HotSpot with a resolution of 256 x 256 grid cells on the die. Of the 26 SPEC2000 benchmarks, only 11 (7 int (bzip2, crafty, eon, gcc, gzip, perlbmk, vortex) and 4 fp (art, galgel, mesa, sixtrack)) have peak steady state temperatures above the thermal emergency threshold of $85°C$. Since only these benchmarks need DTM, the results presented here are averages over these 11 benchmarks.

Results. Figure 7 plots the difference between the peak and the ambient temperatures for the various levels of DTM granularity as a function of the number of cores averaged over the 11 benchmarks listed above. Figure 7(a) plots the results for the *center* scenario with the cores at the center while Figure 7(b) plots the same for the *checker* scenario. The *All-on* curves denote the case with no DTM.

We first observe that without DTM, using the *checker* floorplan instead of *center* reduces peak temperature by $5°C$ (11%, relative to the ambient) on average. This is due to spatial filtering: in *checker* there is a much wider spatial distribution of heat because the cores are separated by L2.

The *Unit-off*, *Domain-off* and *Core-off* curves plot the best-case (lowest) peak temperatures for the three levels of DTM granularity. Since the power density is kept constant across the number of cores, the total power dissipated in a single core decreases with increasing cores. Hence the thermal benefit of shutting down a single core also decreases. The *Unit-off* and *Domain-off* curves on the other hand show both increasing and decreasing behaviors. This is a function of two competing factors: on the one hand, the peak temperature (even with all the units turned on) diminishes due to spatial filtering, and on the other, the power dissipated (and saved) per unit (or per sub-domain) decreases due to core size scaling. In the curves for *center*, decreases in power dissipation dominate up to 9 cores, after which reductions due to spatial filtering dominate.

We observe that local thermal management (at the functional unit level or at the sub-domain level) ceases to be effective after 9-16 cores: In the case of *center*, at 36 cores sub-domain management reduces peak temperature by just $0.8°C$, or 2%. In the case of *checker*, at 36 cores sub-domain management in *checker* reduces peak temperature by just $1.2°C$, or 4%. At 36 cores, turning off an entire core is more beneficial, reducing temperature by $2.4°C$ (7%) and $4.4°C$ (14%) in *center* and *checker* respectively.

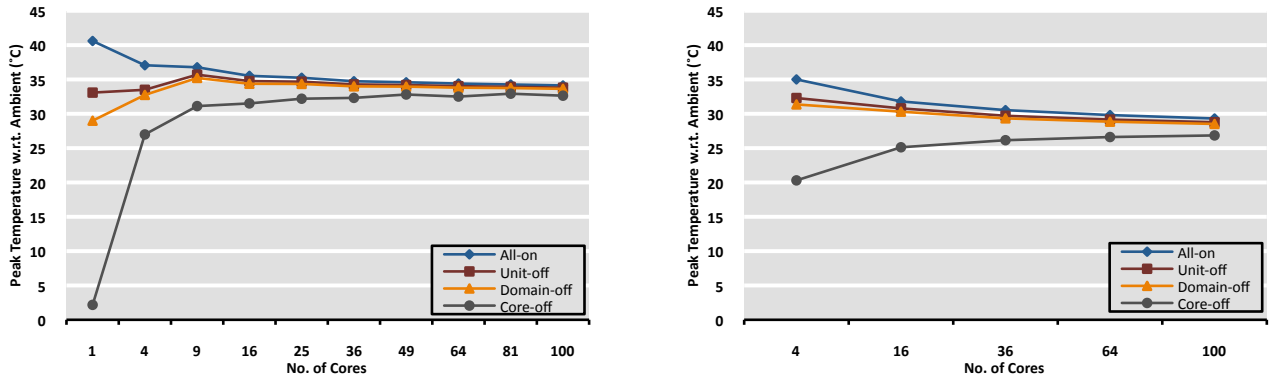
However, when there are 100 cores, even turning off an entire core produces little change: the changes in power density are being spatially filtered. In this case, core-level management reduces temperature by $1.5°C$ (4%) and $2.5°C$ (8%) in *center* and *checker* respectively. This suggests that as a result of spatial filtering, effective DTM policies must consider toggling groups of contiguous cores in order to significantly affect system temperatures.

We also performed sensitivity studies that varied the ratio of the L2 cache area and the package characteristics by removing the heat sink and spreader. The results of these studies were not substantially different from what has been presented above.

B. Sub-blocks with High Aspect Ratio

In our final set of experiments, we investigate whether or not malicious code behavior can damage cores by causing them to overheat in the context spatial temperature filtering.

Prior research has raised concerns about the possibility of pathological code damaging a chip [7]. Several microarchitectural sub-blocks, such as register file entries and cache lines, can be controlled by software either directly (*e.g.*, registers) or indirectly (*e.g.*, cache lines). Under typical program behaviour, the data array of a cache is not a source of hot spots. This is because, over the time scales at which silicon heats up, the accesses to the data array are usually well-distributed in space. Also, on every cache access, only the line that is addressed dissipates dynamic energy. On the other hand, the periphery is usually the hot spot in the cache since the address/data drivers, sense amps, pre-decoders, *etc.* dissipate dynamic energy every time the cache is accessed. However, under malicious program behaviour, a single cache line can potentially become a hotspot if the program directs all cache accesses to it.



(a) Center (b) Checker

Fig. 7: Results of the local *vs.* global thermal management study.

Structures such as register file entries and cache lines are characterized by very high aspect ratios. We saw in section III-C that the aspect ratio of a block and its peak temperature are exponentially related. As aspect ratio increases, so does spatial filtering. This is turn increases lateral thermal conductance, and limits the peak temperature of the block. In this section, we investigate whether malicious code behavior, by concentrating activity in specific sections of a data cache, can raise system temperatures to undesirable levels.

1) Approach: To determine if pathological code can cause cache lines to overheat, we perform simulations using a processor similar to the Alpha 21364 in [15] but scaled to 90 nm. We first collect a representative set of per-unit power consumption values. We do so by simulating the SPEC2000 benchmark suite over a modeling setup similar to that in section IV-A2. From these simulations, we select the benchmark *bzip2* as the candidate for further exploration and illustration, since it has the highest average data cache temperature.

Next, we subdivide the data cache power consumption into those of the individual sub-blocks within the data cache. This is accomplished using Cacti 5.3 [20], which models the performance, dynamic power, leakage power and area of caches. We use Cacti to model an Alpha-like data cache (64 KB, 2-way, 64-byte lines) and separate the data array into the active lines that dissipate dynamic power and passive lines that dissipate leakage. We then aggregate the periphery of each sub-array into a single sub-block (including the row and column decoders, sense amps, muxes, comparators, *etc.*). The address-input/data-output drivers and the request/reply networks are aggregated into the outside mat sub-block. For ease of modeling, the tag and the data portions of the sub-blocks are aggregated into one. Among these sub-blocks, only the active lines (and hence the passive lines as well) can be directly controlled in such a way that program behavior may easily influence the power density and location of power dissipation within them. The other sub-blocks are typically a lot less amenable to program control. Moreover, their thermal behavior is fairly similar across all accesses, independent of the address/location of the line accessed.

Table I shows the area, power consumption and power density of each of these sub-blocks for the *bzip2* benchmark.

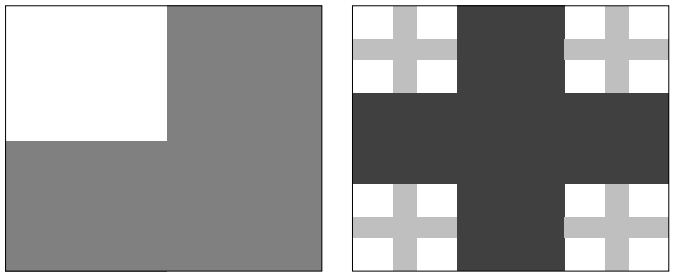
TABLE I: Data cache sub-block area, power and power density executing *bzip2*

Region	Area %	Power %	Power Density Ratio
Outside Mat	56.2	61.4	1.09
Passive Lines	25.7	7.0	0.27
Sub-array Periphery	18.1	22.6	1.24
Active Line	0.03	9.0	358.4

The area and power numbers are expressed as a percentage of the total cache area and power respectively while the power density is represented as a ratio with the average cache power density. We observe that much of the cache area is occupied by the address/data drivers and request/reply networks. The SRAM array occupies only about a quarter of the cache area and most of it is passive at any given time. The power density in these passive lines (due to leakage) is only about a quarter of the average cache power density. On the other hand, the power density at the sub-array periphery is about 24% more than the average. On any given cache access, the power dissipated in the active line is about 9% of the total cache power. For the 64K cache with 64-byte lines, since there are 1024 lines, the area of an active line is a mere 0.03% of the total cache area. Hence, the power density of an active cache line is two orders of magnitude greater than the average.

In general, however, such high power density is not sustained for the time scale at which silicon heats up (tens of thousands of cycles), since the accesses to the cache lines are usually distributed in space. Spatial filtering exposes a trade-off influencing the extent to which pathological code, which can concentrate activity in a single cache line to maintain high power density, can increase cache temperature to undesirable levels: though the power density in the active line is high, the small area and high aspect ratio of the cache line increase lateral heat transport.

2) Experimental setup: In order to study the effect of a cache's layout on its thermal profile, we investigate two different cache arrangements, illustrated in Figure 8. The first is a simplified arrangement with a single sub-array. The shaded portion includes both the "outside mat" sub-block and the sub-array periphery. The second is a placement optimized for performance with the sub-arrays sized in such a manner (using



(a) Placement 1 (b) Placement 2

Fig. 8: Two different placements studied for the data cache. The unshaded areas denote the SRAM sub-arrays while the shaded portions indicate the periphery and routing.

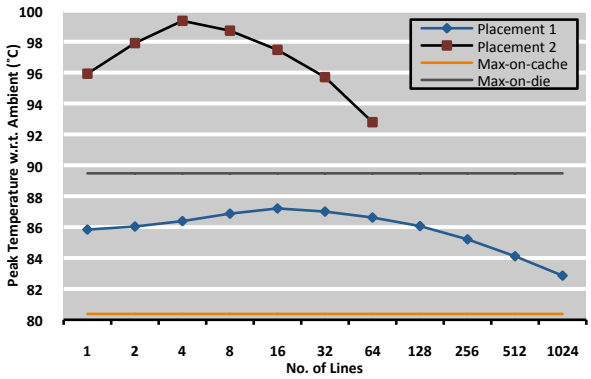


Fig. 9: The peak temperature of the active lines as a function of the number of lines that are accessed contiguously.

Cacti) that the delays along the two dimensions are balanced. It subdivides the SRAM array into 16 sub-arrays, each with 64 lines. The unshaded areas denote the SRAM sub-arrays. The lightly shaded areas denote the periphery of each sub-array and the darkly shaded area denotes the "outside mat" sub-block. The aspect ratio of a single cache line in *Placement 1* is higher than that of a line in *Placement 2*. Since the power densities of the sub-blocks are the same in both the layouts, in considering the two placements, we are actually investigating the impact of sub-block size and aspect ratio on temperature.

In modeling the thermal distribution of these cache arrangements, a challenge arises because of the vastly differing sizes of the sub-blocks. While the other sub-blocks of the cache are comparable in size, the active cache line is different in size and aspect ratio by about three orders of magnitude. Modeling the entire die at the resolution of the single cache line is prohibitive. In FEM parlance, doing so would entail millions of FEM nodes. Hence, we perform thermal modeling at two distinct resolutions. First, we model the case with all the sub-blocks except the active line, *i.e.,* all other cache lines are passive, dissipating only leakage power. This is done using the grid-based model of the HotSpot tool at a grid size of 256 x 256 for the entire die (the data cache alone occupies a sub-grid of about 30 x 35). The package parameters are set to their default values except the convection resistance of the package, which is set to $0.75\frac{K}{W}$. Next, we model the active cache line alone (with the rest of the die dissipating zero power). This is done using the ANSYS tool for a die and package configuration identical to the first step above. Due to the small size of the cache line, the thermal distribution of the die in this case is independent of the location of the line within the cache. Hence, we assume that the cache line is at the center of the die and exploit the symmetry of such a setup about the two axes, thereby reducing the number of required nodes by a factor of four. Furthermore, ANSYS employs a non-uniform mesh to model this setup and hence is able to model it with tens of thousands of FEM nodes. Finally, for combining these two steps, since thermal conduction is a linear phenomenon, we use the principle of superposition and sum up the temperatures. It should be noted that since we are interested in the worst-case, we only consider steady-state thermal behaviour here.

3) Results: Malicious code intent upon heating up the cache can do so either by (a) concentrating activity on a single cache line, or (b) accessing n contiguous cache lines in a round-robin fashion, thereby preventing spatial filtering due to the small size and large aspect ratio of a single cache line. Increasing the size of the target region reduces average power in each of the targeted lines since it accesses every cache line only once every n cycles. Thus the round-robin technique is actually a trade-off between power density and spatial filtering.

Figure 9 plots the temperatures that result from round-robin access of different numbers of cache lines. It plots the peak temperature within the active lines for both the simplified (*Placement1*) and performance-optimized (*Placement2*) cache placements, and several other temperatures for reference, including (1) the peak temperature within the cache when no cache lines are active (*Max-on-Cache*), and (2) the peak temperature within the die when no cache lines are active (*Max-on-Die*). The peak temperature of a single, square cache line (*square-aspect*) is $141°C$ (not plotted).

For the simplified placement, a round-robin technique can access up to 1024 contiguous lines. For the performance-optimized placement, it is restricted to 64 lines since each sub-array only has that many lines. Accessing the lines of another sub-array reduces the advantage of contiguity. The trade-off between power density and spatial filtering is evident in both the *Placement 1* and *Placement 2* curves: they both increase up to a point and then decrease. The maximum point is when the reduction in power density starts to outweigh the benefit from size and aspect ratio.

As mentioned above, the thermal behaviour of an active cache line has two mitigating factors that counterbalance its high power density: its size and its aspect ratio. It can be seen from the graph that the *Placement* * curves are far below the *square-aspect* temperature of $141°C$. This means that aspect ratio significantly affects the peak temperature of a cache line. In fact, this is the reason why the *Placement 2* curve is higher than *Placement 1* (since the aspect ratio of a cache line in *Placement 1* is higher than that of *Placement 2*).

It can be seen that for *Placement 1*, the steady-state temperature rise due to worst-case code behaviour is only $6.8°C$ above the peak temperature within the cache, a 9% increase in temperature relative to the ambient. This is not sufficient to make the cache lines the hottest spots within the die.

On the other hand, for *Placement 2*, the maximum rise is $19°C$ with respect to the rest of the cache, an increase in temperature of 24%. This is significant enough to make it $9.9°C$, or 11%, hotter than the hottest region of the die. As we saw above, this difference is due to the aspect ratio of cache lines in *Placement 1 vs. Placement 2*. As a result, it can be reasonably expected that concentrated accesses in a performance-optimized cache such as the one illustrated in Figure 8(b) would result in accelerated aging and early performance degradation, and ultimately failure, due to sustained use at high temperature. While in practice the extent to which malicious code could accelerate aging is dependent on a variety of factors, it is clear that such attacks must be taken into consideration in the design of caches and the overall thermal design of the system.

V. Conclusions

Higher operating temperatures will pose an increasing design challenge as manufacturing processes scale. Substantial research effort has been devoted to managing temperature at run-time by *distributing functionality in time*, by performing scheduling, duty-cycling, *etc.*, to reduce peak system temperature. In this paper, we have shown that there is also substantial opportunity to manage system temperature at design time by *distributing functionality in space* by carefully selecting and organizing system components to take advantage of *spatial thermal filtering*. To explore the extent of spatial filtering in silicon and its implications for computer architecture, we performed detailed analytical modeling and extensive simulation.

First, we performed analytical modeling to investigate the relationship between changes in various aspects of a single power dissipator and the resulting increase in system temperature. When considering a single layer of silicon, our analytical solution to the three-dimensional heat conduction problem is nearly a perfect match with finite-element simulations. We further observed that in theory as well as in practice that there is an exponential relationship between system temperature and both dissipator size and aspect ratio; in particular, when dissipators are large, or have aspect ratios near unity, changes in size or aspect ratio have little effect temperature. The presence of a sophisticated cooling solution, however, including copper heat spreader and sink, significantly dampens the spatial thermal filtering effect. There is a caveat, however. The analytical model predicts that small changes in the size of large dissipators should have no effect on system temperature. However, because in practice the relationship between temperature and size is so much more shallowly exponential in the presence of heat spreading, reductions in dissipator size produce modest reductions in temperature, nearly regardless of the initial size of the dissipator.

This has a number of important implications for computer architecture. First, we observed that core count, core orientation, and die thickness have little effect on peak temperature. Scaling cores such that the number of cores in the system changes from 2 to 200, thereby significantly increasing the spatial frequency of power dissipation (power density was assumed to remain the same), reduces peak temperature by only 14%. Likewise, changing the orientation of cores results in a 0.5% change in peak temperature.

We also observed that as the number of cores on-chip increases, the granularity of effective thermal management also increases, to the point that only turning off entire cores produces any substantial reduction in peak temperature for systems with 100 cores. In this case, turning off a single core reduces peak temperature by $2.5°C$, while a single functional unit achieves only a $0.5°C$ reduction: the circuitry and software required to fine tune power dissipation is unlikely to justify the gains, because the resulting changes in power density are spatially filtered. However, it should be noted that the most important thermal management technique we explored was affected at design time rather than runtime: tiling cores in a checkerboard rather than clustering them in the center of the die reduced peak temperature by $5°C$ on average.

We concluded our experimentation with a microarchitectural case study in spatial filtering. We observed that cache lines, when accessed repeatedly, can be heated to $19°C$ hotter than the rest of the surrounding cache and $10°C$ hotter than the next hottest area of the die. Despite the fact that the high aspect ratio of cache lines significantly improves their lateral heat conductance, this increase in temperature is substantial enough to significantly accelerate wear-out.

While higher operating temperatures are unavoidable, they can be managed with a variety of run- and design-time techniques. In our exploration of spatial filtering in silicon, we have demonstrated that careful design that considers the various relationships between temperature, cooling solution, and different aspects of component selection and system organization, can result in significant reductions in peak temperature.

Acknowledgments

We would like to thank Jessica Gullbrand for the discussion about periodic boundary conditions, which contributed significantly to the development of the presented analytical model. This work was supported in part by the NSF under grant nos. EHS-0509245 and CRI-0551630, and by a research grant from Intel Research.

References

[1] J. Srinivasan, S. V. Adve, P. Bose, and J. A. Rivers, "The impact of technology scaling on lifetime reliability," in *the Proceedings of the 2004 International Conference on Dependable Systems and Networks, DSN'04*, June 2004.

[2] W. Huang, M. R. Stan, K. Sankaranarayanan, R. J. Ribando, and K. Skadron, "Many-core design from a thermal perspective," in *the Proceedings of the 45th Annual Conference on Design Automation, DAC'08*, June 2008, pp. 746–749.

[3] K. Etessam-Yazdani, M. Asheghi, and H. F. Hamann, "Investigation of the impact of power granularity on chip thermal modeling using white noise analysis," *IEEE Trans. Compon., Packag., Technol.*, vol. 31, no. 1, March 2008.

[4] K. Etessam-Yazdani, H. Hamann, and M. Asheghi, "Impact of power granularity on chip thermal modeling," in *the Proceedings of the Tenth Intersociety Conference on Thermal and Thermomechanical Phenomena in Electronics Systems, ITHERM '06*, May 2006, pp. 666–670.

[5] W. Huang, M. R. Stan, K. Skadron, K. Sankaranarayanan, and S. Ghosh, "Hotspot: A compact thermal modeling method for CMOS VLSI systems," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 14, no. 5, pp. 501–513, May 2006.

[6] W. Huang, K. Sankaranarayanan, K. Skadron, R. J. Ribando, and M. R. Stan, "Accurate, pre-rtl temperature-aware design using a parameterized, geometric thermal model," *IEEE Transactions on Computers*, vol. 57, no. 9, pp. 1277–1288, September 2008.

[7] P. Dadvar and K. Skadron, "Potential thermal security risks," in *the Proceedings of the IEEE/ASME Semiconductor Thermal Measurement, Modeling, and Management Symposium, SEMI-THERM'05*, March 2005.

[8] J. C. Ku, S. Ozdemir, G. Memik, and Y. Ismail, "Thermal management of on-chip caches through power density minimization," in *the Proceedings of the 38th annual IEEE/ACM International Symposium on Microarchitecture, MICRO-38*, November 2005, pp. 283–293.

[9] J. Kong, J. K. John, E.-Y. Chung, S. W. Chung, and J. Hu, "On the thermal attack in instruction caches," accepted for publication in *IEEE Transactions on Dependable and Secure Computing*, vol. 7, no. 2, April-June 2010.

[10] W. Huang, K. Skadron, S. Gurumurthi, R. J. Ribando, and M. R. Stan, "Differentiating the roles of ir measurement and simulation for power and temperature-aware design," in *ISPASS'09*, 2009.

[11] H. S. Carslaw and J. C. Jaeger, *Conduction of Heat in Solids: Second Edition*. Oxford, UK: Oxford University Press, 1986, c1959.

[12] G. E. Myers, *Analytical Methods in Conduction Heat Transfer*. Schenectady, NY, USA: Genium Pub. Corp., 1987.

[13] E. Kreyszig, *Advanced Engineering Mathematics: 8th Edition*. New York, NY, USA: John Wiley & Sons, Inc., 1998.

[14] "Ansys 12.1," http://www.ansys.com.

[15] K. Skadron, M. R. Stan, W. Huang, S. Velusamy, K. Sankaranarayanan, and D. Tarjan, "Temperature-aware microarchitecture," in *Proceedings of the 30th Annual ACM/IEEE International Symposium on Computer Architecture*, April 2003, pp. 2–13.

[16] Standard Performance Evaluation Corporation, "SPEC CPU2000 Benchmarks," http://www.specbench.org/osg/cpu2000.

[17] T. Austin, E. Larson, and D. Ernst, "Simplescalar: An infrastructure for computer system modeling," *IEEE Computer*, vol. 35, no. 4, pp. 59–67, February 2002.

[18] D. Brooks, V. Tiwari, and M. Martonosi, "Wattch: A framework for architectural-level power analysis and optimizations," in *Proceedings of the 27th Annual ACM/IEEE International Symposium on Computer Architecture*, June 2000, pp. 83–94.

[19] T. Sherwood, E. Perelman, G. Hamerly, and B. Calder, "Automatically characterizing large scale program behavior," in *Proceedings of the Tenth International Conference on Architectural Support for Programming Languages and Operating Systems, ASPLOS-X*, October 2002, pp. 45–57.

[20] S. Thoziyoor, N. Muralimanohar, J. H. Ahn, and N. P. Jouppi, "Cacti 5.1," HP Laboratories, Palo Alto, Tech. Rep. HPL-2008-20, April 2008.

APPENDIX
SOLUTION OF THE HEAT EQUATION

For the convenience of the reader, we re-state the boundary value problem from Section III (Equations (1)-(2)) here:

$$\frac{\partial^2 T}{\partial x^2} + \frac{\partial^2 T}{\partial y^2} + \frac{\partial^2 T}{\partial x^2} = 0 \quad (1)$$
$$(-2a \le x \le 2a)$$
$$(-2b \le y \le 2b)$$
$$(0 \le z \le l)$$

This is to be solved subject to the boundary conditions:

$$-k\frac{\partial T}{\partial z}|_{z=0} = q\cos(\frac{2\pi x}{4a})\cos(\frac{2\pi y}{4b}) \quad (2a)$$
$$-k\frac{\partial T}{\partial z}|_{z=l} = hT \quad (2b)$$
$$T|_{x=-2a} = T|_{x=+2a} \quad (2c)$$
$$-k\frac{\partial T}{\partial x}|_{x=-2a} = -k\frac{\partial T}{\partial x}|_{x=+2a} \quad (2d)$$
$$T|_{y=-2b} = T|_{y=+2b} \quad (2e)$$
$$-k\frac{\partial T}{\partial y}|_{y=-2b} = -k\frac{\partial T}{\partial y}|_{y=+2b} \quad (2f)$$

The standard practice for solving problems such as equation (1) is to seek solutions of a *separable* form *i.e.*, assuming $T(x,y,z) = X(x)Y(y)Z(z)$. In that case, the steady state heat equation $\nabla^2 T = 0$ reduces to the form $\frac{X''}{X} + \frac{Y''}{Y} + \frac{Z''}{Z} = 0$. In this equation, each of the three terms on the left side is a function of a different independent variable. If their sum is to be zero for all x, y and z, then each term has to separately be equal to a constant (*i.e.*, it cannot be a function of x, y or z). Since temperature is a real-valued function, these constants have to be real as well. So that their sum would be non-trivially zero, at least one of them has to be positive and at least one negative. Then, depending on our boundary conditions, we typically have two types of equations to solve. The first case is of the form $X'' = \lambda^2 X$ and the second is of the form $X'' = -\lambda^2 X$. The solution to the former is of the form $C_1 e^{\lambda x} + C_2 e^{-\lambda x}$ where C_1 and C_2 are arbitrary constants. This can be seen from the fact that differentiating this expression twice with respect to x results in it being multiplied by λ^2, satisfying $X'' = \lambda^2 X$. Similarly, the solution to the latter case ($X'' = -\lambda^2 X$) is of the form $C_1\cos(\lambda x) + C_2\sin(\lambda x)$. This can also be verified to satisfy $X'' = -\lambda^2 X$ as before. Furthermore, since at least one of the above-mentioned constants is positive and at least one negative, it can be seen that at least one of the functions ($X(x)$, $Y(y)$ or $Z(z)$) takes the former form (comprised of exponentials), while at least one takes the latter form (comprised of sines/cosines).

Considering the supplied power density as the *input* and the resultant temperature as the *response*, since our problem is linear, the response must have the same frequency as the input. Additionally, since there is no *phase shift* (*i.e.*, the location of the peak temperature coincides with that of the peak power density), $X(x)$ must take the form $C_1\cos(\lambda_a x)$ where $\lambda_a = \frac{2\pi}{4a}$. Similarly, $Y(y)$ must take the form $C_2\cos(\lambda_b y)$ where $\lambda_b = \frac{2\pi}{4b}$. Therefore, by the argument in the previous paragraph, $Z(z)$ must take the form $D_1 e^{\lambda_c z} + D_2 e^{-\lambda_c z}$ where $\lambda_c = \sqrt{\lambda_a^2 + \lambda_b^2}$. For notational convenience, expressing $Z(z)$ as a function of $(l - z)$, we get

$$T(x,y,z) = (Ae^{\lambda_c(l-z)} + Be^{-\lambda_c(l-z)})\cos(\lambda_a x)\cos(\lambda_b y) \quad (3)$$

where A and B are arbitrary constants. Now, applying the boundary condition from Equation (2b) to Equation (3), we dervice the relationship between A and B as

$$B = A\frac{(k\lambda_c - h)}{(k\lambda_c + h)}. \quad (4)$$

Then, applying the boundary condition (2a) to Equation (3) and substituting for B from Equation (4), we get:

$$A = \frac{ql}{k}\frac{1}{\lambda_c l}\left[\frac{(k\lambda_c + h)}{(k\lambda_c + h)e^{\lambda_c l} - (k\lambda_c - h)e^{-\lambda_c l}}\right] \quad (5a)$$
$$B = \frac{ql}{k}\frac{1}{\lambda_c l}\left[\frac{(k\lambda_c - h)}{(k\lambda_c + h)e^{\lambda_c l} - (k\lambda_c - h)e^{-\lambda_c l}}\right] \quad (5b)$$

Substituting A and B from Equations (5a) and (5b) into

Equation (3), we get the solution of the heat equation:

$$T(x,y,z) = \frac{ql}{k}\frac{1}{\lambda_c l}\left[\frac{(k\lambda_c+h)e^{\lambda_c(l-z)}+(k\lambda_c-h)e^{-\lambda_c(l-z)}}{(k\lambda_c+h)e^{\lambda_c l}-(k\lambda_c-h)e^{-\lambda_c l}}\right] \cdot \cos(\lambda_a x)\cos(\lambda_b y), \quad (6)$$

where $\lambda_a = \frac{2\pi}{4a}, \lambda_b = \frac{2\pi}{4b}$ and $\lambda_c = \sqrt{\lambda_a^2+\lambda_b^2}$.