NORTHWESTERN
UNIVERSITY

Electrical Engineering and Computer Science Department

Technical Report
NWU-EECS-10-05
March 4, 2010

Power Scaling: the Ultimate Obstacle to 1K-Core Chips

Nikos Hardavellas[*], Michael Ferdman[†‡], Anastasia Ailamaki[‡], Babak Falsafi[‡]

[*]Northwestern University, Department of Electrical Engineering and Computer Science
[†]Carnegie Mellon University, Department of Electrical and Computer Engineering
[‡]École Polytechnique Fédérale de Lausanne, School of Computer and Communication Sciences

ABSTRACT

As Moore's Law continues for at least another decade, the number of cores on chip and the on-chip cache size will continue to grow at an exponential rate. While workloads with limited parallelism pose performance challenges with multicore processors, server workloads with abundant parallelism are believed to be immune, capable of scaling to the parallelism available in the hardware. However, despite the inherent scalability in threaded server workloads, increasing core counts cannot directly translate into performance improvements because chips are physically constrained in power and off-chip bandwidth.

In this work, we explore the design space of physically-constrained multicore chips across technologies and show that, even with conservative estimates, chips will not scale beyond a few tens of cores due to physical power and off-chip bandwidth constraints, potentially leaving the die real-estate underutilized in future technology generations. We observe that customized heterogenous multicores can leverage die area to overcome the initial power barrier, resulting in bandwidth constrained designs. Overcoming the bandwidth wall, e.g. through the use of large multi-gigabyte 3D-stacked caches, fully exposes multicore designs to the power wall, requiring innovation in low-power interconnects and on-chip hierarchies to further improve the performance of future servers.

Keywords: multicore, process technology, performance, power, area, bandwidth, physical constraints, cache, interconnect, 3D, heterogeneous multicore.

Power Scaling: the Ultimate Obstacle to 1K-Core Chips

Nikos Hardavellas*, Michael Ferdman†‡, Anastasia Ailamaki‡, Babak Falsafi‡
*Northwestern University, Department of Electrical Engineering and Computer Science
†Carnegie Mellon University, Department of Electrical and Computer Engineering
‡École Polytechnique Fédérale de Lausanne, School of Computer and Communication Sciences

ABSTRACT

As Moore's Law continues for at least another decade, the number of cores on chip and the on-chip cache size will continue to grow at an exponential rate. While workloads with limited parallelism pose performance challenges with multicore processors, server workloads with abundant parallelism are believed to be immune, capable of scaling to the parallelism available in the hardware. However, despite the inherent scalability in threaded server workloads, increasing core counts cannot directly translate into performance improvements because chips are physically constrained in power and off-chip bandwidth.

In this work, we explore the design space of physically-constrained multicore chips across technologies and show that, even with conservative estimates, chips will not scale beyond a few tens of cores due to physical power and off-chip bandwidth constraints, potentially leaving the die real-estate underutilized in future technology generations. We observe that customized heterogenous multicores can leverage die area to overcome the initial power barrier, resulting in bandwidth constrained designs. Overcoming the bandwidth wall, e.g. through the use of large multi-gigabyte 3D-stacked caches, fully exposes multicore designs to the power wall, requiring innovation in low-power interconnects and on-chip hierarchies to further improve the performance of future servers.

1 INTRODUCTION

Shortcomings of existing architectures, along with the continued rise in the number of transistors available on chip, have encouraged a switch to multicore (CMP) architectures. CMPs avoid an increase in core complexity, and instead integrate multiple processors on a single die, relying on the parallelism exposed within the workload. While the availability of parallelism in desktop and engineering applications is limited, there is a general belief that server workloads, where parallelism is abundant [17], can scale by taking advantage of the multicore hardware. Thus, vendors and researchers have pursued designs with high core counts,

maximizing the number of lean on-chip cores [24] and threads [26,29], with projections of growing to 100s or 1000s of cores in the future [7,37].

However, multicores are not a panacea for server processor designs. While Moore's Law enables more transistors on chip [6], the static power consumption of the additional transistors can no longer be mitigated through circuit-level techniques [11]. Although a trade-off exists between cache performance and leakage power, the cache latency cannot be sufficiently reduced to deliver reasonable performance and simultaneously keep at bay the leakage power of exponentially growing caches. Additionally, the multiplying core counts and thread contexts constitute a substantial fraction of the chip's transistors, steadily raising both static and dynamic core power consumption. While voltage-frequency scaling may lower the dynamic power of the cores and enable more cores on chip, static power dissipation and performance requirements impose a limit. Future multicore designs are therefore rapidly approaching the power wall.

Even if the power limitation can be temporarily elided through highly efficient core designs or low-operational-power transistors, the rising core and thread counts will drastically increase pressure on the limited and non-scalable off-chip memory bandwidth, encountering the bandwidth wall [36]. Traditional approaches to alleviate off-chip bandwidth pressure call for larger on-chip caches, which further drive up the chip's power consumption, reducing the power available to the cores. Thus, despite the abundant parallelism present in server workloads, without a technological miracle, the number of cores in future CMPs will be severely limited by the chip power envelope and the constrained off-chip bandwidth.

To understand the CMP characteristics necessary to attain peak performance while staying within the physical constraints of power and bandwidth, it is imperative to jointly optimize all design parameters. To date, there has been no objective and comprehensive study that examines how multicore trends affect the performance of server workloads. In this work, we consider a large array of design parameters and construct detailed models which conform to ITRS projections of future manufacturing technologies. We jointly optimize supply and threshold voltage, on-chip clock frequency, core count, manufacturing process, on-chip cache size, and memory technology to draw the following conclusions for future multicore server trends:

- CMPs will not scale beyond a small number of cores. Power constraints force under-utilization of the die area and require reduction in cache and core performance to reduce leakage power and meet the power budget. Simultaneously, high thread counts and limited cache overload off-chip bandwidth. Coupled with Amdahl's law providing diminishing returns with each additional core, physically-con-

strained designs having a large number of power- and bandwidth-limited cores are at best competitive with designs having a smaller number of fast cores.

- The die real-estate can be effectively used by power-efficient customized heterogenous cores implemented with low-operational-power transistors, with all but the most application-specific hardware disabled. Specialized heterogenous designs can achieve peak bandwidth-bound performance with a small number of cores of the highest possible single-thread performance, maximizing the power available for large caches that reduce off-chip bandwidth pressure.
- Large multi-gigabyte 3D-stacked-DRAM caches can effectively overcome the bandwidth wall. 3D-stacked caches allow higher performance through increased parallelism to all multicore designs, with the largest gains available to heterogenous designs that become primarily limited by the workload parallelism.

The rest of this paper is organized as follows. Section 2 presents a forecast of the analysis. Our approach is illustrated intuitively in Section 3. Section 4 evaluates a large and diverse CMP design space to determine trends and projections of future server multicores. We summarize the related work in Section 5 and conclude in Section 6. Finally, Section 7 presents our analytical models and the empirical results for the validation of our workload-specific parameters.

2 ANALYSIS FORECAST

The desire for higher performance for server workloads suggests CMPs with high core counts. Yet, physical limitations restrict the number of cores that can be practically employed. We forecast that CMP designs that balance the on-chip resources to obtain peak performance will employ only a modest amount of cores. Increasing the core count will force the chip to run slower, so it can remain within the power and bandwidth envelope, yielding a suboptimal design point.

In an attempt to break past the power wall, we analyze a range of techniques that minimize power consumption, from voltage-frequency scaling, to using low-power or customized cores, to employing low-operational-power transistors even for time-critical chip components. However, even with conservative estimates and the utilization of all these techniques, we find that CMPs attaining peak performance will still employ only a modest amount of cores. While these techniques lower power consumption and allow for higher core counts, the increasing number of cores pushes CMPs against the bandwidth wall. We observe

that 3D-stacked DRAM caches effectively mitigate the off-chip bandwidth constraints, making power the ultimate limiter to CMP scaling. With techniques to lower core power and chip leakage already under way, large-scale multicores require innovation in light-weight on-chip interconnects and memory hierarchies.

3 METHODOLOGY

Complexity and run-time requirements make it impractical to rely on full-system simulation for a large-scale design-space exploration study. Instead, we rely on first-order analytical models of the dominant components. Our algorithm uses the analytical models as constraints, always finding the core count and cache size of the peak-performing design. The details of our performance, power, area, and bandwidth models are presented in Section 7.

We use the example in Figure 1 to illustrate our algorithm for finding the peak-performing designs. The "Area" curve in Figure 1 shows the area-constrained core vs cache tradeoff (farthest right curve). The "Power" curve shows the same relationship, but bound only by the chip power budget (lowest and left-most curve). Although potentially hundreds of cores can fit into the available area, only a handful of them can be powered.

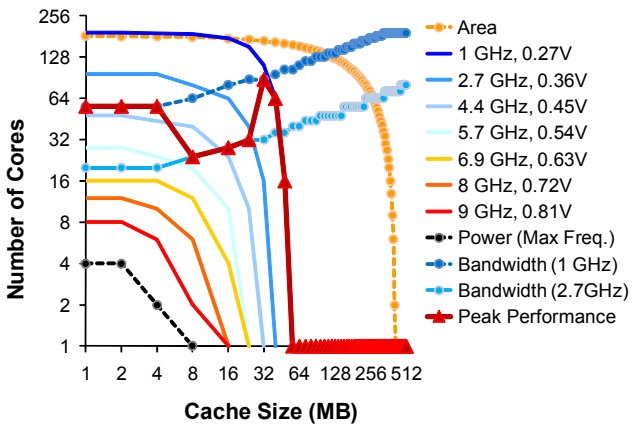

FIGURE 1: Core count-cache size trade-off subject to physical constraints.

Voltage scaling enables lowering the chip power consumption and permitting designs with more cores and cache, albeit at lower performance because the frequency also decreases. The progressive curves between the "Area" and "Power" curves show designs with varying voltages/frequency. For example, it is possible to lower supply voltage to fully populate the chip area, as shown by the intersection of the "Area" and the "1GHz/0.27V" curves; however, at 0.27V, the cores are restricted to 1GHz operation, while the technology supports frequencies in excess of 10GHz. We further overlay the off-chip bandwidth constrained designs on Figure 1, showing core vs cache ratios at the 1GHz and 2.7GHz off-chip bus frequency.

Although bandwidth limitations favor larger caches, power constraints favor smaller caches with lower leakage power. More and faster cores translate into higher performance, but the power and bandwidth walls favor fewer and slower cores. Therefore, selection of the highest performance design must balance conflict-

ing requirements imposed by the physical constraints. The "Peak Performance" curve in Figure 1 shows the progression of our algorithm. The algorithm walks along the most limiting physical constraint, exploring slack of the other constraints to improve performance. In the example of Figure 1, power initially limits the clock rate and voltage to 1GHz / 0.27V. The algorithm proceeds along the bandwidth limit until an 8MB cache, where it switching to 2.7GHz bandwidth with fewer cores. At 32MB cache size, the power wall for 0.36V forces a lower voltage, allowing to power 88 instead of 32 cores. The rest of the candidate designs are power constrained, with the highest performance design at the intersection of the power and bandwidth limits for 2.7GHz / 0.36V.

To more intuitively show the progression of the algorithm, we plot the same design points, replacing core count with application performance on the *Y* axis in Figure 2. The "Area" curve shows area-constrained designs at maximum frequency, assuming unlimited power and bandwidth. The "Power" curve shows power-constrained designs at maximum frequency, assuming unlimited area and bandwidth. The "Area+Power" curve uses voltage-frequency scaling (VFS), showing the highest-performing designs assuming unlimited bandwidth. Finally, the "Bandwidth" curve shows VFS designs subject only to bandwidth constraints. The "Peak Performance" design is shown as initially bounded by bandwidth, eventually reaching the "Area+Power" voltage-scaled power constraint at 32MB of cache.

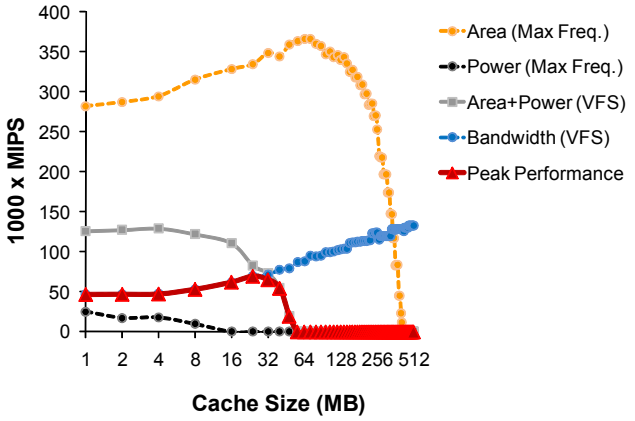

FIGURE 2: Performance of physically-constrained designs.

3.1 Technology Model

We model multicore processors across four fabrication technologies: 65nm (in large-scale commercial production since 2007), 45nm (to be used by the majority of new products by 2010), 32nm (due in 2013) and 20nm (due in 2017). For each technology node, we utilize parameters and projections from the International Technology Roadmap for Semiconductors (ITRS) 2008 Edition [6]. When scaling across technologies, we follow ITRS forecasts on new device types that are expected to replace devices that do not scale beyond a given technology node. In agreement with ITRS, we model bulk planar CMOS for the 65nm and

45nm nodes, ultra-thin-body fully-depleted MOSFETs for 32nm technology [16], and double-gate FinFETs [38] for the 20nm node. Prototypes of these devices are under development at several industrial labs.

3.2 Hardware Model

We model CMPs with cores built in one of three ways: general purpose (GPP), embedded (EMB), or ideal (Ideal-P). GPP cores are similar to the cores in Sun's UltraSPARC [26,29]. We model 4-way multi-threaded scalar in-order cores, as similar cores have been shown to optimize performance for server workloads [15,17]. We calculate that a 4-way multi-threaded core achieves speedup of 1.7x over a single-threaded core when running server workloads, corroborating prior research [17]. Because general-purpose cores consume an inordinate amount of power and area compared to embedded cores, we also evaluate cores similar to the ones in ARM11 MPCore [3,21]. Based on prior research, we conservatively estimate that an EMB core delivers the same performance as a single-threaded GPP core [43,44] when running commercial workloads.

To obtain an upper bound on core performance and power efficiency, we evaluate ideal cores (Ideal-P) that have ASIC-like properties: Ideal-P cores deliver 7x the performance of a GPP core and consume 140x less power [12]. The evaluation of Ideal-P cores is especially relevant to designs in the deep-nanometer regime, where abundant die real-estate enables heterogenous CMPs with cores that are heavily optimized for different functionality. A heterogeneous CMP may enable only the cores that most closely match the requirements of the available work, and use GPP cores only for non-critical or complex/uncommon parts of the program, thereby exhibiting near-ASIC properties for most cores.

Each core is supported by 64KB L1 instruction and 64KB L1 data caches. The CMP employs a shared L2 cache ranging from 1MB to 512MB in size. We optimize each L2 cache configuration for each technology node with CACTI 6.0 [33], and use the tool's average access latency estimate in our models. CACTI 6.0 models the access time, cycle time, area, and power for a wide range of cache organizations (from small conventional caches to large NUCA [25] caches), jointly optimizing the cache organization and aspect ratio, the on-chip interconnect and the wire technology. Each NUCA slice is also independently optimized and multi-banked for performance [10,33]. We do not evaluate deeper on-chip cache hierarchies because prior research shows that a NUCA organization outperforms any multi-level cache design [25]. Relevant CMP parameters are listed in Table 1 (a).

4 ANALYSIS

4.1 Impact of Workload Parallelism on CMP Core Count

We analyze the impact of varying degrees of application parallelism on the CMP core count by employing our models to devise peak-performance designs for three CMP configurations: two with GPP cores and one with EMB cores. The details of the CMP designs are explained in Section 4.2 and Section 4.3. Figure 3 shows the core counts of the resulting peak-performing designs. We observe that over a wide range of application parallelism, the core count of the peak-performing CMP designs remains within a narrow band for all except the 100%-parallel workload. Thus, the core count of peak performing designs is largely independent of the parallelism available in the workload, except for workloads with near-perfect parallelism (over 99.5%). Unless otherwise noted, the remainder of our analysis assumes workloads with 99% parallelism.

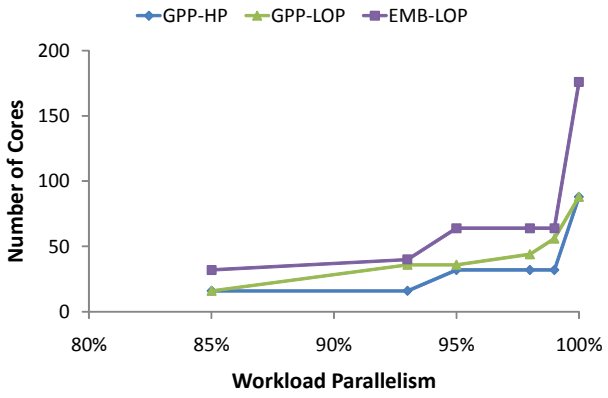

FIGURE 3: Peak performance designs at 20nm for varying application parallelism (DSS).

4.2 Physically-Constrained Designs Across Technologies

To mitigate the power wall, some processors utilize high-V_{th} transistors for non-time-critical components to lower the leakage current. Such low-operational-power (LOP) transistors achieve orders of magnitude lower subthreshold leakage current, while retaining 54%-68% of the switching speed of high-performance (HP) transistors [6]. Caches are a prime candidate for using LOP transistors, as their activity level is significantly lower than the cores' and the high transistor density of caches results in high aggregate leakage.

To evaluate the impact of device-level power savings on CMP design, we run our models across technologies for CMPs with GPP cores that utilize (i) HP transistors for the entire chip, (ii) HP/LOP transistors for the cores/cache respectively, and (iii) LOP transistors for the entire chip. A detailed description of the transistors we evaluate appears in [6]. We present the design-space exploration results in Figure 4, and the core counts of peak-performing designs across technologies in Figure 5. In the interest of brevity, Figure 4 shows results only for OLTP at 20nm; the trends are the same across technologies and workloads.

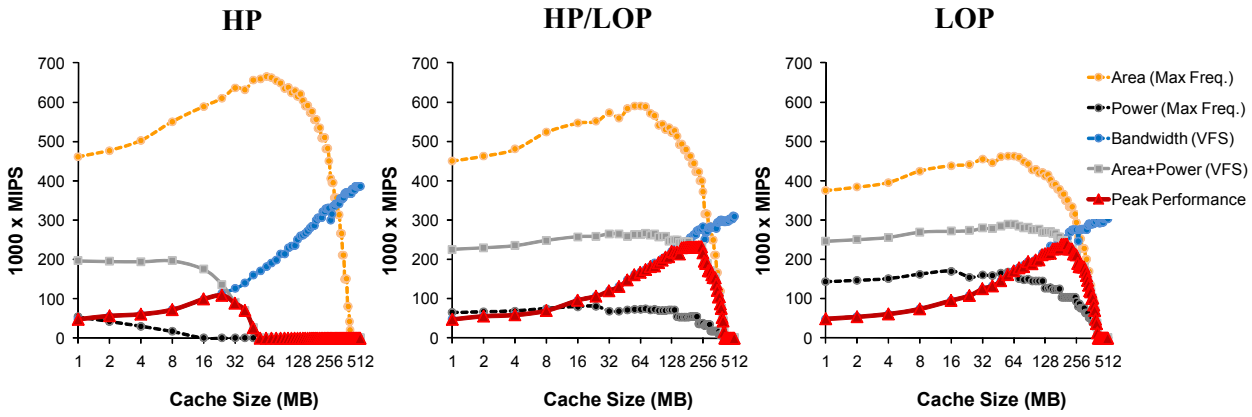

FIGURE 4: Performance of GPP CMPs using HP, HP/LOP and LOP transistors at 20nm running OLTP.

Because full-HP designs are severely power-limited across technologies (Figure 4, left), only a small number of cores can be powered. Although the die at 20nm can fit 180 cores, HP designs can power only approximately 32 cores (Figure 5). Utilizing LOP transistors for the cache enables larger caches that can support more cores and yield higher performance (Figure 4 and Figure 5, middle). At 20nm, HP/LOP designs support approximately 64 cores, twice the HP core count, with approximately 20% of the chip power dissipated due to leakage in the cores.

Implementing cores with LOP transistors can eliminate core leakage, at the cost of per-core performance. However, due to power constraints, the peak-performing HP designs must employ on-chip clocks at least 43% lower than the maximum frequency supported by the technology. Although LOP transistors are slower than HP transistors, they retain 54%-68% of the maximum switching speed at an optimal clock rate. As such, we find that LOP devices can be used to implement the cores as well as the cache, obtaining similar CMP performance as HP/LOP designs (Figure 4, right) while achieving 25% higher performance per watt

As expected, workloads with less parallelism benefit from designs built with HP transistors rather than LOP transistors. However, the crippling effect of the power wall limits the core count and clock frequency

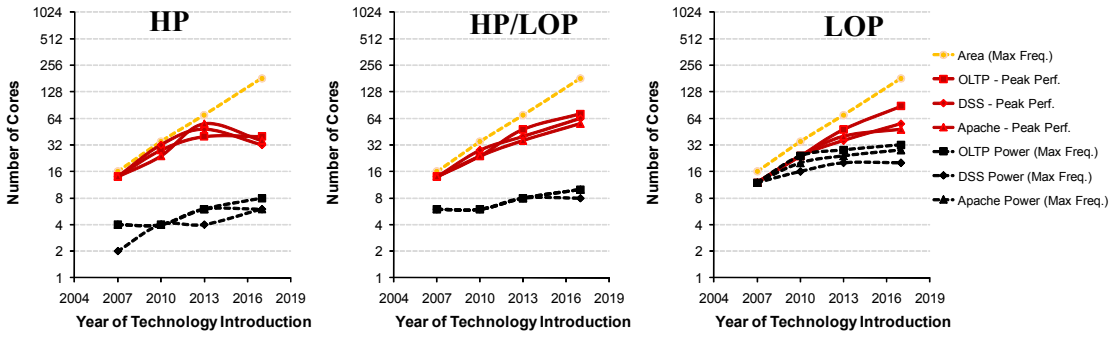

FIGURE 5: Core count for peak-performance GPP CMPs using HP, HP/LOP and LOP transistors.

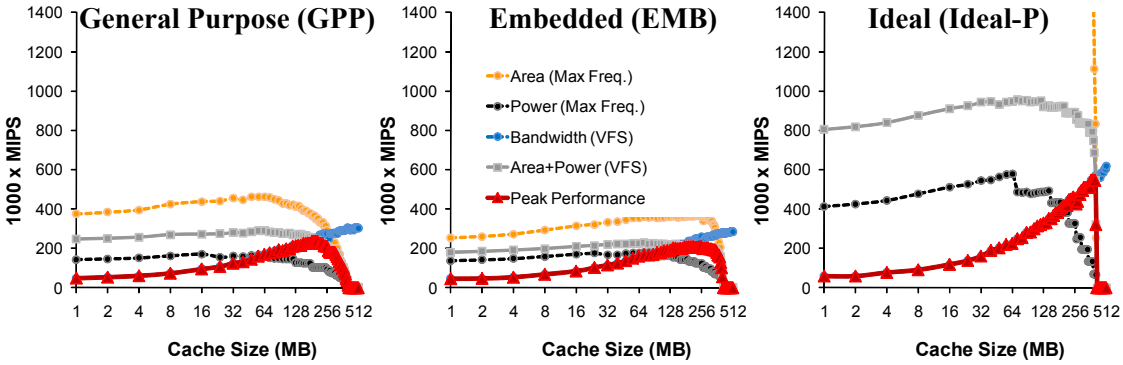

FIGURE 6: Performance of GPP, EMB, and Ideal-P CMPs using LOP transistors at 20nm running OLTP.

of HP designs. As a result, LOP designs with higher core counts outperform HP designs down to workloads with only 30% available parallelism, at which point a 4-core 9GHz HP design offers a marginal absolute performance benefit over a 24-core 6.7GHz LOP design. LOP designs are competitive with HP/LOP designs up to 97% parallelism, at which point the higher core count of LOP designs overtakes the higher per-core performance of HP/LOP designs, enabling LOP designs to exhibit greater absolute performance and higher performance per watt. For the remainder of the paper we focus on LOP designs.

4.3 Multicore Processors With milliWatt Cores

Lean cores deliver high performance when running commercial server workloads at reasonable power consumption (e.g., Sun UltraSPARC T1 consumes less than 2W per thread [29]). However, embedded systems are dominated by milliWatt cores that deliver reasonable performance at orders of magnitude lower power. For example, the ARM1176JZ(F)-S consumes 279mW with an 8-stage, scalar, out-of-order pipeline, dynamic branch prediction, separate Ld/St and arithmetic pipelines, a SIMD unit, and a vector floating-point co-processor [4]. Prior research has indicated that simple in-order cores are preferable for commercial server workloads [15,17]. These workloads typically exhibit tight data dependencies and adverse memory

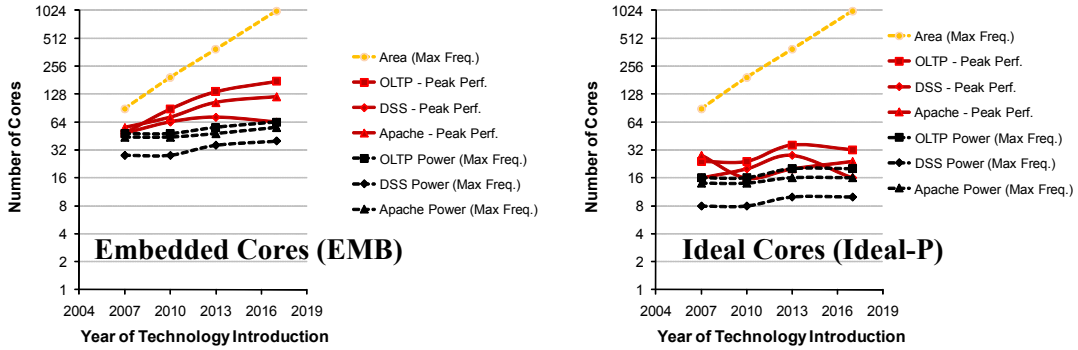

FIGURE 7: Core count of CMPs with embedded (EMB) and ideal (Ideal-P) cores using LOP transistors.

access and sharing patterns that are impervious to most architectural optimizations. Thus, simple efficient cores present a viable building block for server multicores.

We find that EMB-based multicores generally exhibit trends similar to GPP-based multicores. The peak-performing designs are bandwidth-constrained at small cache sizes, becoming power-constrained for larger caches, with the highest performing designs at the intersection of the constraints (Figure 6). Both GPP and EMB designs require similar-sized caches to remain within the bandwidth envelope.

However, to reach peak performance, EMB multi-cores require double the core count compared to GPP multicores (Figure 5 right and Figure 7 left). Although additional cores deliver significantly higher performance in today's 65nm technology (Figure 11), at smaller technologies with higher core counts, additional cores provide a marginal performance benefit due to Amdahl's Law. While the best 20nm EMB design allows for 176 cores compared to 88 GPP cores, the EMB design trails 13% in absolute performance with a 99% parallel workload, achieving a speedup over GPP designs only with 99.6% or higher workload parallelism. Furthermore, higher core counts require larger interconnects, dissipating 68% more power than the interconnect of the GPP design (Figure 8). The EMB performance per watt is therefore similar to GPP designs, with power efficiency of EMB cores outweighed by the power consumption of the interconnect. We evaluated multi-threaded EMB cores, but due to the increased power and bandwidth requirements, we observe minimal differences compared to EMB designs

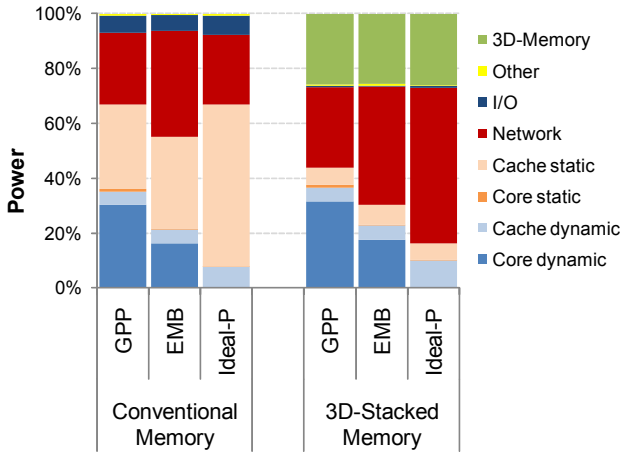

FIGURE 8: Power breakdown of conventional and 3D-memory CMPs using LOP transistors at 20nm (OLTP).

4.4 Multicore Processors with Ideal Cores

Amdahl's Law prohibits large core counts from delivering high aggregate performance (except for embarrassingly parallel applications). An alternative design is to deliver higher performance with fewer cores. We evaluate an extreme application of this approach by considering heterogeneous computing, where a multicore chip may contain hundreds of diverse powered down cores, enabling only those cores that are most useful to a given application's performance. Low core count reduces the impact of Amdahl's Law,

while matching of specialized cores to an application's requirements enables high performance at high power efficiency.

We explore heterogeneous designs by evaluating multicores built with Ideal-P cores exhibiting ASIC-like properties: Ideal-P cores deliver 7x performance of a single-threaded GPP core at 1/140th the power. Although it remains questionable whether cores may ever achieve these goals, the inclusion of reconfigurable logic and spatial computing [12] may approximate these assumptions in some cases. Thus, we consider our analysis a first step towards a feasibility study, rather than an accurate performance estimator.

Superior power and performance characteristics of Ideal-P cores push the power envelope much further than possible with other core designs (Figure 6). As a result, Ideal-P multicores attain roughly 2x speedup over the GPP and EMB designs. Unlike GPP and EMB designs that are ultimately power-limited, Ideal-P designs are primarily constrained by off-chip bandwidth. Bandwidth constraints force Ideal-P designs to hundreds of megabytes of cache, dominating the power budget with cache leakage.

Superior single-core performance of Ideal-P, along with the limitations imposed by Amdahl's Law on massive parallelism, allows small-scale CMPs to achieve higher performance than GPP or EMB-based designs with four times more cores. Although almost a thousand cores can fit in a 20nm chip, the optimal (bandwidth-limited) Ideal-P designs are at 16 to 32 cores, with remaining die area used for reducing off-chip bandwidth requirements through a larger on-chip cache. The observation of low core-count Ideal-P designs exhibiting peak performance is especially true for workloads with smaller fractions of parallelism, and holds up to 99.9% parallelism for the workloads we studied.

4.5 Effect of Multi-Gigabyte On-Chip Caches

Advances in fabrication technology have resulted in techniques that enable stacking multiple chip substrates on top of each other [32]. Communication between the substrates is performed through vertical buses which can deliver terabytes per second of bandwidth [32]. Although stacking multiple processor chips may have prohibitive thermal implications, stacking memory on top of processing cores results only in a small increase in temperature (10^{o}C for 9 additional layers [32]) while offering unprecedented bandwidth to the memory arrays. The resulting 3D-stacked memory can be used as a large "in-package" cache, that can ease the burden imposed by the high core counts on the off-chip pins.

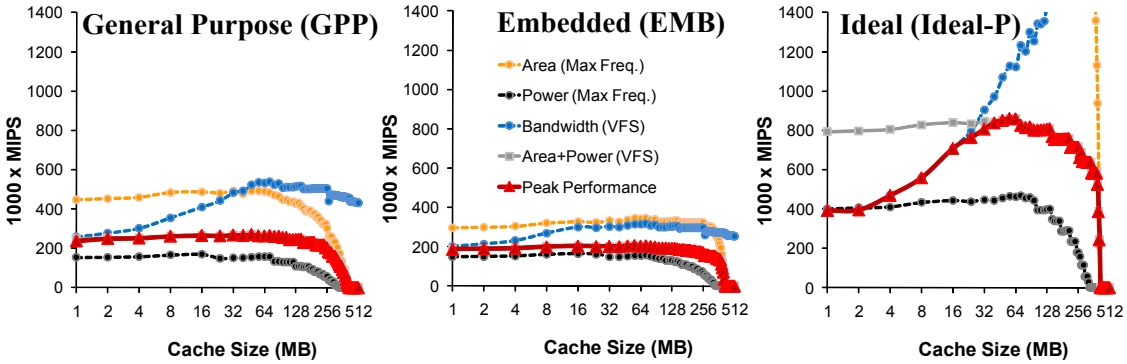

FIGURE 9: GPP, EMB and Ideal-P CMPs using LOP transistors and 3D-memory at 20nm (OLTP).

The designs considered in previous sections are limited by bandwidth and power, two interrelated constraints. Bandwidth considerations result in designs with large caches, which limit power available to add more cores or to allow for faster cores. We evaluate 3D-stacked CMP designs across technologies, where the stacked memory is used as a cache and present our results in Figure 9 and Figure 10. We find that 3D-stacked memory pushes the bandwidth constraint beyond the power constraint in most cases (Figure 9). This leads to peak-performance designs that are only power-constrained and achieve higher performance than their conventional-memory counterparts. Figure 11 shows the speedup of each design, with and without a 3D-stacked cache, averaged over all our workloads.

Although 3D-memory delivers a modest performance improvement in GPP or EMB multicore processors (less than 35%), reduction in off-chip bandwidth requirements results in almost 2x speedup when used with Ideal-P cores. Figure 12 shows the relationship of available parallelism and the average Ideal-P core count of peak performance designs across our workloads. A 3D-stacked cache eliminates the bandwidth wall, enabling a small on-die cache to realize high performance; in the case of perfectly scalable (100% parallelism) applications, only 16MB of cache is needed, with the majority of the die populated by cores. However,

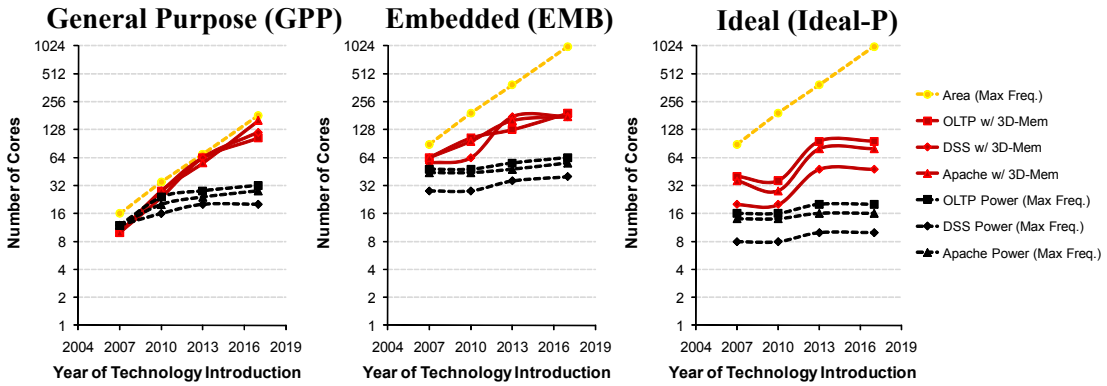

FIGURE 10: Core counts for GPP, EMB and Ideal-P CMPs using LOP transistors and 3D-stacked memory.

Amdahl's Law results in diminishing returns from high core counts, favoring fewer cores and larger caches for peak-performance designs at lower available parallelism.

With all core designs we evaluate, the use of a large 3D-stacked memory alleviates the off-chip bandwidth wall for most memory accesses, allowing for more cores on chip compared to bandwidth-limited designs (about two to three times more cores at 20nm—Figure 10). We observe that the higher core counts in these designs result in CMPs where the network subsystem dominates chip power (Figure 8) and becomes the new bottleneck.

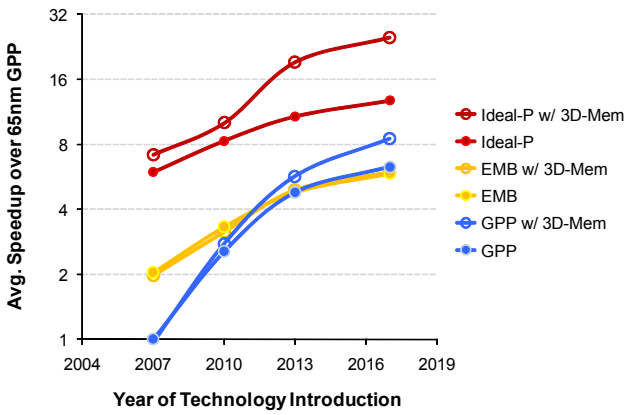

FIGURE 11: Speedup of GPP, EMB and Ideal-P CMPs using LOP transistors and conventional or 3D-stacked memory.

5 RELATED WORK

Hartstein *et al.* [19] evaluate the nature of cache misses for a variety of workloads and validate the square-root rule-of-thumb for cache misses. Rogers *et al.* [36] extend this work to CMPs and conclude that miss rates follow a simple power law. In this work, through robust fitting of hundreds of candidate functions, we find that x-shifted power laws accurately describe the cache miss behavior of commercial server workloads, while simpler power laws may generate relative errors in excess of 50%. Hill and Marty [20] analytically explore how different levels of software parallelism and core asymmetry affect the performance of multicore processors, while our model focuses on the trade-offs between physical constraints and performance.

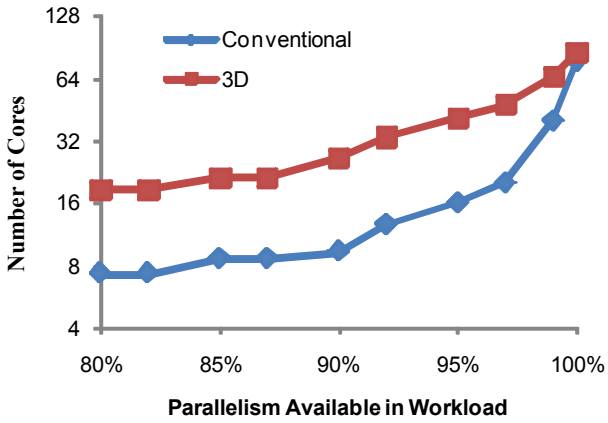

FIGURE 12: Number of cores of peak-performance 20nm designs with and without a 3D-stacked cache.

Rogers *et al.* [36] model the effect of die area allocation to cores and caches on the on-chip memory traffic in current and future technology generations to conclude that bandwidth is the primary performance constraint. However, Rogers *et al.* do not consider power implications on performance and leverage the assumption that modern multicores are already bandwidth constrained, which contradicts prior research

[9]. We agree with the observation that 3D-stacked memory does not alleviate the bandwidth wall when a single stacked layer is considered, but we find that the addition of multiple layers of dense DRAM arrays [32] effectively mitigates the bandwidth wall for all designs across technologies.

[15] explored the design space of CMPs, focusing on throughput as the primary performance metric to compare server workload performance across chip multiprocessors with varying processor granularity. However, this study stops short of a detailed performance characterization and breakdown of where time is spent during execution. Similarly, Huh *et al.* [23] explore the design space of CMPs to determine the best configuration and extrapolate SPEC CPU results to server workloads, but do not consider the power implications of CMP designs, focusing the study on smaller systems where bandwidth and power are less critical. Moreover, the performance model in [23] employs private L2 caches per core, which greatly increase the data sharing overhead and off-chip miss rate.

Li *et al*. [31] present a comprehensive study of the CMP design space subject to physical constraints and jointly optimize across a large number of design parameters. However, they investigate only SPEC CPU benchmarks and a single technology node (65nm), while we focus on commercial server workloads across technologies. Moreover, [31] assumes that cache latency remains constant when scaling the cache size, which does not allow for an accurate exploration of a wide range of cache sizes.

Alameldeen [2] studies how compression improves processor performance, and develops an analytic model to balance cores, caches, and communication. In contrast, we explore how physical constraints determine the configuration of CMPs across technologies and evaluate different devices, core technologies, and memory technologies to extend the constraints. Kumar *et al*. [27] present a performance evaluation of a heterogeneous CMP, but focus on a CPU-intensive diverse workload rather than a homogeneous commercial server workloads targeted by our study.

6 CONCLUSIONS

As Moore's Law continues for at least another decade, the number of cores on chip and the on-chip cache size will continue to grow at an exponential rate. Although server workloads have abundant parallelism, they are not capable of scaling to the parallelism available in the hardware because increasing core counts cannot directly translate into performance improvements, as chips are physically constrained in power, bandwidth, and area.

In this work, we developed models to explore the design space of physically-constrained CMPs across technologies. Through this analysis, we showed that CMPs will not scale beyond a few tens of cores due to physical power and off-chip bandwidth constraints, potentially leaving the die real-estate underutilized in future technology generations. We evaluated embedded and ASIC-like cores, and found that scalability using embedded cores is limited, particularly in future fabrication technologies, while heterogeneous multicores and voltage and frequency scaling can leverage die area to overcome the initial power barrier. We observed that 3D-stacked caches can mitigate the bandwidth wall, resulting in power-constrained designs, requiring innovation in low-power interconnects and on-chip hierarchies to further improve the performance of future servers.

7 FIRST-ORDER ANALYTICAL MODEL

As Moore's Law continues and the number of transistors on chip rises exponentially, there is sufficient die real-estate to fabricate large-scale CMPs with hundreds of cores. Under ideal conditions, such CMPs are able to execute several billion instructions per second. Unfortunately, this massive processing capability is throttled by the latency gap between the memory subsystem and the processor. For many commercial server applications, only a fraction of the peak performance can be achieved [1,9,15,17]. Growing the on-chip cache allows for more data to be serviced from the faster cache rather than the slower main memory, but cache and cores compete for die area. At the same time, power and thermal considerations limit the number of cores that can run concurrently, while leakage current limits the amount of cache that can be employed. Although scaling the supply voltage allows for lower overall power consumption, it does so only at the expense of performance. Concurrently, memory bandwidth constraints impede the ability to feed all cores with data, raising yet another wall that CMP designs must take into account [36].

In this work, we use construct first-order models relating the effects of technology-driven physical constraints to the application performance of commercial workloads running on future CMPs. The goal of this effort is not to provide absolute values on how many cores or how much cache a CMP should have. Rather, our intent is to uncover trends in future CMP designs.

7.1 Area Model

We model 310 mm^2 dies [6] and proportionally scale the cores and cache area [15], allocating 72% of the die for cores and cache, with the remaining area allocated to the on-chip interconnect, memory controllers,

and system-on-chip components. We estimate the core area by scaling existing designs. For GPP cores, we scale the cores of the Sun UltraSPARC T1 processor [29], using 13.67 mm^2 per core (including L1 caches) in a 65nm node. For EMB cores, we scale the ARM11 MPCore, using 2.48 mm^2 at 65nm. We estimate the area of Ideal-P cores to be equal to EMB cores. We estimate the area required for the L2 cache by calculating the area of ECC-protected tag and data arrays following ITRS projections. Finally, we scale the cores and caches across technologies in accordance to ITRS guidelines on transistor size, logic and SRAM density, area efficiency, and SRAM cell area factor for each technology.

7.2 Performance Model

It is important to consider Amdahl's Law when investigating massive parallelism, as even a small serial portion can severely limit the speedup obtained by employing more cores. For each CMP configuration, workload, and process technology, we calculate the performance of the CMP using Amdahl's Law. We model applications with parallelism ranging between 80-100%. For example, a 99% parallelizable application on 128 cores yields a speedup of 56x, while on 1024 cores it achieves only 91x (an order of magnitude less than linear speedup). Even at 99.9% parallelism, 1024 cores barely reach 507x speedup.

We estimate the performance of a single core by calculating the aggregate number of user instructions committed per cycle (*IPC*), as this metric is proportional to overall system throughput [41]. To make comparisons across different cores, we consider the relative performance between GPP, EMB and Ideal-P cores presented in Section 3.2. We estimate *IPC* by calculating the expected number of cycles an instruction needs to execute, accounting for the probability this instruction accesses the L2 cache or main memory (typically, the L1 hit latency is not exposed to the application). The probability that an instruction accesses the L2 cache (hit or miss) is proportional to the fraction of dynamic load/store instructions in the application and the L1 miss rate. Because both the fraction of load/store instructions and the L1 miss rate are characteristic of the application and do not depend on the CMP configuration, we empirically measure them for each application using the FLEXUS [18,41] full system simulator. Table 1 (a) and (b) presents the CMP system and application parameters used in the simulations.

To calculate the cycles per instruction spent on L2 and memory accesses, we estimate the L2 access latency using CACTI 6.0, and calculate the memory access latency by scaling DRAM latencies by 7% each year, starting with a DRAM latency of 53ns in 2007 at the 65nm node (e.g., PC-667). For the calculation of the

Table 1: (a) CMP parameters. (b) Workload parameters

CMP cores	16 cores, in-order; UltraSPARC III ISA 4-way fine-grain multithreading
L1 Caches	split I/D, 64KB 2-way set-associative 64-byte blocks, 16-entry victim buffer
L2 Cache	1-512MB, 16-way set-associative, 64-byte blocks
Interconnect	2D-Torus

OLTP – Online Transaction Processing	
OLTP	*TPC-C v3.0 on IBM DB2 v8 ESE,* 100 warehouses (10 GB), 64 clients, 2 GB buffer pool
Web Server	
Apache	***SPECweb99 on*** *Apache HTTP Server v2.0.* 16K connections, fastCGI, worker threading model
DSS – Decision Support Systems	
DSS	*TPC-H Throughput Test on IBM DB2 v8 ESE,* 480 MB buffer pool, 1GB database, 16 clients, Queries 2,6,13,16

memory latency, we model an on-chip memory controller with a 2-cycle latency, with the DRAM modules located on the board about 5-20cm from the processor chip.

7.3 L2 Cache Miss Rate and Dataset Evolution Models

7.3.1 L2 Cache Miss Rate Model

We use a combination of modeling and full-system simulation to estimate the L2 cache miss rate. We simulate and measure each workload's miss rate as a function of the L2 size, varying the L2 size from 256KB to 64MB at coarse steps. We then curve-fit [42] the miss rate measurements to find a function that best approximates the measured miss rates. Using an automated system, we evaluated over 900 functions with at most three coefficients to control smoothness and over-fitting. The functions included polynomials, logarithmic functions, exponentials, hyperbolas, x- and y-shifted power laws, reciprocal functions, and functions prominent in the scientific literature (e.g., Weibull, Steinhart-Hart). The parameters of each function are individually fitted to provide the lowest sum of absolute values of relative errors. The function that most accurately predicts the cache miss rates for our workloads is an x-shifted power law:

$$y = \alpha(x + \beta)^{\gamma}$$

where y is the target miss rate, x is the size of the cache in MB, and α, β, γ are the fitted parameters.

Table 2: Miss Rate Model Parameters.

X-Shifted Power Law: $y = \alpha (x + \beta)^{\gamma}$					
	α	**β**	**γ**	mean error	max error
OLTP	0.5785	0.4750	-0.589	1.3%	8.2%
DSS	0.5925	0.5154	-0.327	0.5%	6.5%
Apache	1.0081	2.1104	-0.503	1.2%	4.9%

The fitted parameters for each workload are shown in Table 2, along with the average and maximum errors of the fitted function. The average error of the x-shifted power law is less than 1.3% across our workloads, with a maximum error of 8.2%. In contrast, a power law of the form $y = \alpha x^{\gamma}$, used in prior studies to model miss rates [36], fails to capture accurately the miss rate characteristics of commercial server workloads as it attains average errors of 11% for Apache and 6.4% for OLTP, with max-

imum errors of 58% and 23.6% respectively. It is worth noting that the commonly-used rule-of-thumb that quadrupling the cache size halves the miss rate is subsumed by the traditional power law. Although the rule-of-thumb is among the most accurate functions with one coefficient, the additional degrees of freedom offered by the x-shifted power law allow a more accurate estimate of the miss rate.

7.3.2 Dataset-Adjusted Cache Miss Rate Model

Just as fabrication technology advances over time, the application datasets experience exponential growth. Therefore, predicting a workload's cache miss rate across technologies requires to consider the evolution of the application datasets. We estimate the growth of application datasets for commercial server workloads by measuring the dataset growth of Transactional Processing Council's TPC-A, -B, -C, and -E workloads since 1994 [14]. These workloads are updated regularly to represent the computational demands of online transactional processing and data warehousing in large-scale database management systems. Our measurements indicate that TPC benchmark datasets grow by 29.13% per year, corroborating Myhrvold's Law [28]. Based on the projected evolution of datasets, we adjust our cache miss rate models by lowering the cache's effective size by the expected growth of the application's dataset for each technology node.

7.4 Power Model

The reference cores used for power estimates are the same as those used for the area model (Section 3.2). The total chip power is the sum of the dynamic and static power of the individual components (cores, cache, interconnect, I/O, and the miscellaneous system-on-chip components). The maximum allowable power for air-cooled chips with heatsink is estimated for each technology by ITRS.

7.4.1 Dynamic Power

We obtain the core dynamic power consumption by scaling the dynamic power of the reference core proportionally to the gate capacitance of the target technology, the target frequency, and the square of the supply voltage. We estimate the power for various levels of supply voltage ranging from the nominal voltage defined by ITRS for each technology, down to 2.3x [5] the threshold voltage for the same technology. This allows us to model the effects of voltage-frequency scaling, trading off clock frequency for lower power. We account for the non-linear relationship between supply voltage and frequency by fitting published data [13]. The voltage scaling is quantized in steps of 10% reduction over the nominal voltage.

We estimate the L2 dynamic power by scaling published data for the UltraSPARC T1 cache [29]. In addition to scaling the cache power across technologies in a fashion similar to the core dynamic power, we adjust the cache power proportionally to the cache activity (access rate). We calculate the activity factor from the relative L1 miss rates for our designs and workloads, the number of cores on chip, and the relative performance of the cores. We compute the network dynamic power based on network activity, scaled over the same reference design. The activity of the network is equal to the activity of the cache, adjusted by the average hop count of each message on a 2D-torus on-chip interconnect.

We calculate the power of the I/O subsystem proportionally to the reference GPP core design, the L2 access rate by all participating cores, and the L2 miss rate, and we scale it across technologies similarly to cores. Because bandwidth is limited, the worst case power is expended when all I/O pins are fully utilized. To account for this limitation, we cap the bandwidth to the maximum allowed by the packaging process and I/O technology, as predicted by ITRS. We model the power consumption of system-on-chip components similarly to the active power of the core, scaling the reference design power across technologies.

7.4.2 Static Power

We model only the sub-threshold leakage and ignore the gate and junction leakage based on the results of prior research [35]. The static power of the cache is proportional to its size, the supply voltage, the transistor width, and the leakage current at the corresponding temperature. We estimate an average ratio of gate length to gate width of 3.0 across technologies [6] and obtain gate lengths from ITRS. We scale the leakage current for different temperatures similar to [22,30], fitting the sub-threshold coefficient for a target temperature, calculating the thermal voltage from Boltzmann's constant and the electron's electrical charge, and fitting data for the threshold voltage temperature coefficient. We calculate leakage at 66C, a typical operating temperature of today's CMPs [29].

We therefore core leakage as part of the chip power model. We calculate the leakage of the cores by estimating the number of transistors in a core using ITRS logic transistor density projections, assuming that, on any given cycle, one half of the core bits remain unchanged.

7.5 Off-Chip Bandwidth and 3D-Die Stacking Models

We model the chip bandwidth requirements by estimating the relative off-chip activity rate, scaled from the off-chip activity rate measured in simulation of the application on the reference GPP design. The off-chip

bandwidth is proportional to the L2 miss rate, the number of cores, and the activity of the cores (i.e., clock frequency and their relative performance).

In addition to evaluating a conventional memory system, we evaluate CMPs that use 3D-stacked memory [32] as a high-capacity high-bandwidth L3 cache. We model a 3D-stacked memory where each layer has a capacity of 8Gbits at 45nm technology [32]. The worst-case power consumption for each 8Gbit layer is 3.7W [32]. We model 8 layers, for a total of 8GB, with an additional layer used to host controllers and logic. The 9 stacked layers increase the average temperature of the chip by an estimated 10C [32]. Thus, when we evaluate 3D-stacked designs, we account for the effects of the increased temperature on power. Communication from the cores/cache layer to the 3D-stacked memory is performed through vertical buses. Stacking of memory layers above logic enables single-cycle vertical communication to the memory die by traversing a small number of layers a few microns thick. We estimate the area for a 1Kbit vertical bus at 45nm at 0.32 mm^2 [32], and model 8 such buses in our designs for a total of 2.56mm^2 area.

We treat 3D-stacked memory as a large L3 cache because the memory it houses is not sufficient for a full large-scale server software installation. When using a 3D-stacked cache, our models include two memory subsystems: one that extends from the L2 cache to the 3D-stack, and one that extends from the 3D-stack to the memory modules on board. For the 3D-stack on chip, we estimate that memory access time improves an additional 32.5% due to more efficient communication between the cores and the memory in the 3D-stack [32]. We model the miss rate of the 3D-stack using the same x-shifted power law as for the L2.

REFERENCES

[1] A. Ailamaki, D. J. DeWitt, M. D. Hill, and D. A. Wood. DBMSs on a modern processor: Where does time go? In *The VLDB Journal*, Sept. 1999.

[2] A. R. Alameldeen. *Using compression to improve chip multiprocessor performance*. PhD thesis, University of Wisconsin at Madison, Madison, WI, USA, 2006. Adviser-Wood, David A.

[3] ARM. ARM MPCore. http://www.arm.com/products/CPUs/ARM11MPCoreMultiprocessor.html.

[4] ARM. ARM1176JZ(F)-S: enhanced security and lower energy consumption for consumer and wireless applications. http://www.arm.com/products/CPUs/ARM1176.html.

[5] S. I. Association. The international technology roadmap for semiconductors (ITRS), process integration, devices, and structures. http://www.itrs.net/, 2002 Update.

[6] S. I. Association. The international technology roadmap for semiconductors (ITRS). http://www.itrs.net/, 2008 Edition.

[7] M. Azimi, N. Cherukuri, D. N. Jayasimha, A. Kumar, P. Kundu, S. Park, I. Schoinas, and A. S. Vaidya. Integration challenges and tradeoffs for tera-scale arch. *Intel Technology Journal*, Aug. 2007.

[8] J. Balfour and W. J. Dally. Design tradeoffs for tiled CMP on-chip networks. In *Proceedings of the 20th Annual International Conference on Supercomputing*, 2006.

[9] L. A. Barroso, K. Gharachorloo, and E. Bugnion. Memory system characterization of commercial work-

loads. In *Proc. of the 25th Annual Intl. Symposium on Computer Architecture*, June 1998.
[10] B. M. Beckmann and D. A. Wood. Managing wire delay in large chip-multiprocessor caches. In *Proceedings of the 37th Annual IEEE/ACM International Symposium on Microarchitecture*, 2004.
[11] S. Borkar. Microarchitecture and design challenges for gigascale integration. In *Proceedings of the 37th Annual IEEE/ACM International Symposium on Microarchitecture*, 2004.
[12] M. Budiu, G. Venkataramani, T. Chelcea, and S. C. Goldstein. Spatial computation. In *Proc. of the 11th Intl. Conf. on Architectural Support for Prog. Lang. and Operating Systems*, 2004.
[13] T. Burd, T. Pering, A. Stratakos, and R. Brodersen. A dynamic voltage scaled microprocessor system. In *2000 IEEE International Solid-State Circuits Conference, 2000.*
[14] T. P. P. Council. The TPC Benchmark TM H. *Transaction Processing Performance Council*, 2001.
[15] J. D. Davis, J. Laudon, and K. Olukotun. Maximizing CMP throughput with mediocre cores. In *Proc. of the Thirteenth Intl. Conf. on Parallel Arch. and Compilation Techniques*, 2005.
[16] J. Deng, K. Kim, C.-T. Chuang, and H.-S. P. Wong. Device footprint scaling for ultra thin body fully depleted SOI. In *Proc. of the 8th Intl. Symp. on Quality Electronic Design*, 2007.
[17] N. Hardavellas, I. Pandis, R. Johnson, N. Mancheril, A. Ailamaki, and B. Falsafi. Database servers on chip multiprocessors: Limitations and opportunities. In *3rd Biennial Conf. on Innovative Data Systems Research*, pages 79–87, Asilomar, CA, USA, 2007.
[18] N. Hardavellas, S. Somogyi, T. F. Wenisch, R. E. Wunderlich, S. Chen, J. Kim, B. Falsafi, J. C. Hoe, and A. G. Nowatzyk. Simflex: A fast, accurate, flexible full-system simulation framework for performance evaluation of server architecture. *SIGMETRICS Performance Evaluation Review*, 31(4):31–35, April 2004.
[19] A. Hartstein, V. Srinivasan, T. R. Puzak, and P. G. Emma. Cache miss behavior: is it √2? In *CF '06: Proceedings of the 3rd Conference on Computing frontiers*, 2006.
[20] M. D. Hill and M. R. Marty. Amdahl's law in the multicore era. *Computer*, 41(7):33–38, 2008.
[21] K. Hirata and J. Goodacre. ARM MPCore: The streamlined and scalable arm11 processor core. In *Proceedings of the 2007 Asia and South Pacific Design Automation Conference*, 2007.
[22] H. Hua, C. Mineo, K. Schoienfliess, A. Sule, S. Melamed, and W. Davis. Performance trend in three-dimensional integrated circuits. In *Interconnect Tech. Conference, 2006 International*, 2006.
[23] J. Huh, D. Burger, and S. W. Keckler. Exploring the design space of future CMPs. In *Proc. of the Ninth International Conference on Parallel Architectures and Compilation Techniques*, 2001.
[24] T. Kgil, S. D'Souza, A. Saidi, N. Binkert, R. Dreslinski, T. Mudge, S. Reinhardt, and K. Flautner. Picoserver: using 3d stacking technology to enable a compact energy efficient chip multiprocessor. *SIGOPS Oper. Syst. Rev.*, 40(5):117–128, 2006.
[25] C. Kim, D. Burger, and S. W. Keckler. An adaptive, non-uniform cache structure for wire-delay dominated on-chip caches. *ACM SIGPLAN Not.*, 37(10):211–222, 2002.
[26] P. Kongetira, K. Aingaran, and K. Olukotun. Niagara: A 32-way multithreaded SPARC processor. *IEEE Micro*, 25(2):21–29, 2005.
[27] R. Kumar, D. M. Tullsen, P. Ranganathan, N. P. Jouppi, and K. I. Farkas. Single-ISA heterogeneous multi-core architectures for multithreaded workload performance. In *Proc. of the 31st Annual International Symposium on Computer Architecture*, 2004.
[28] J. Larus. Spending moore's dividend. *Communications of the ACM*, 52(5):62–69, 2009.
[29] A. S. Leon, K. W. Tam, J. L. Shin, D. Weisner, and F. Schumacher. A power-efficient high-throughput 32-thread SPARC processor. *IEEE Journal of Solid-state circuits*, 2007.
[30] B. Li, L.-S. Peh, and P. Patra. Impact of process and temperature variations on network-on-chip design exploration. In *Proc. of the 2nd ACM/IEEE Intl. Symposium on Networks-on-Chip*, 2008.
[31] Y. Li, B. Lee, D. Brooks, Z. Hu, and K. Skadron. CMP design space exploration subject to physical constraints. In *The 12th Intl. Symposium on High-Performance Computer Architecture*, 2006.
[32] G. H. Loh. 3D-stacked memory architectures for multi-core processors. In *Proceedings of the 35th International Symposium on Computer Architecture*, 2008.
[33] N. Muralimanohar, R. Balasubramonian, and N. Jouppi. Optimizing NUCA organizations and wiring al-

ternatives for large caches with CACTI 6.0. In *Proceedings of the 40th Annual IEEE/ACM International Symposium on Microarchitecture*, 2007.

[34] P. Ranganathan, K. Gharachorloo, S. V. Adve, and L. A. Barroso. Performance of database workloads on shared-memory systems with out-of-order processors. In *Proc. of the 8th Intl. Conf. on Architectural Support for Prog. Lang. and Operating Systems*, 1998.

[35] S. Rodriguez and B. Jacob. Energy/power breakdown of pipelined nanometer caches (90nm/65nm/45nm/32nm). In *Proc. of the 2006 Intl. Symposium on Low Power Electronics and Design*, 2006.

[36] B. M. Rogers, A. Krishna, G. B. Bell, K. Vu, X. Jiang, and Y. Solihin. Scaling the bandwidth wall: challenges in and avenues for CMP scaling. In *Proceedings of the 36th Annual International Symposium on Computer Architecture*, 2009.

[37] L. Seiler, D. Carmean, E. Sprangle, T. Forsyth, M. Abrash, P. Dubey, S. Junkins, A. Lake, J. Sugerman, R. Cavin, R. Espasa, E. Grochowski, T. Juan, and P. Hanrahan. Larrabee: a many-core x86 architecture for visual computing. *ACM Trans. Graph.*, 27(3):1–15, 2008.

[38] T. Skotnicki. Materials and device structures for sub-32 nm CMOS nodes. *Microelectronics Engineering*, 84(9-10):1845–1852, 2007.

[39] S. Somogyi, T. F. Wenisch, A. Ailamaki, B. Falsafi, and A. Moshovos. Spatial memory streaming. In *Proc. of the 33rd Annual International Symposium on Computer Architecture*, 2006.

[40] T. F. Wenisch, S. Somogyi, N. Hardavellas, J. Kim, A. Ailamaki, and B. Falsafi. Temporal streaming of shared memory. In *Proc. of the 32nd Annual Intl. Symp. on Computer Architecture*, 2005.

[41] T. F. Wenisch, R. E. Wunderlich, M. Ferdman, A. Ailamaki, B. Falsafi, and J. C. Hoe. SimFlex: statistical sampling of computer system simulation. *IEEE Micro*, 26(4):18–31, Jul-Aug 2006.

[42] ZunZun.com. ZunZun.com online curve fitting and surface fitting web site. http://zunzun.com/.

[43] T. Kgil, S. D'Souza, A. Saidi, N. Binkert, R. Dreslinski, T. Mudge, S. Reinhardt, and K. Flautner. PicoServer: using 3D stacking technology to enable a compact energy efficient chip multiprocessor. In *Proc. of the 12th Intl. Conf. on Architectural Support for Programming Languages and Operating Systems*, 2006.

[44] D. G. Andersen, J. Franklin, M. Kaminsky, A. Phanishayee, L. Tan, and V. Vasudevan. FAWN: a fast array of wimpy nodes. In *Proc. of the 22nd ACM SIGOPS Symposium on Operating Systems Principles*, 2009.