Assessing the Performance Limits of Parallelized Near-Threshold Computing

Nathaniel Pinckney, Korey Sewell, Ronald G. Dreslinski, David Fick, Trevor Mudge, Dennis Sylvester, and David Blaauw
EECS Department University of Michigan, Ann Arbor, MI.
npfet@umich.edu

ABSTRACT

Supply voltage scaling has stagnated in recent technology nodes, leading to so-called “dark silicon.” In this paper, we investigate the limit of voltage scaling together with task parallelization to maintain task completion latency. When accounting for parallelization overheads, minimum task energy is obtained at “near threshold” supply-voltages across 6 commercial technology nodes and provides 4X improvement in overall CMP performance.

Categories and Subject Descriptors

B.8.0 [**Hardware**]: Performance and Reliability – g*eneral.*

General Terms

Performance, Design.

Keywords

Near-Threshold Computing, low-power design, parallelization.

1. INTRODUCTION

Moore’s law [1] has historically enabled increased microprocessor performance with each technology node while maintaining constant power density. However, conventional voltage scaling has slowed in recent years. As transistors have become leakier, it has become more difficult reduce the threshold voltage and hence supply voltage scaling has stagnated as well in order to maintain sufficient overdrive and performance [2]. This deviation from constant-field scaling theory [3], combined with continuing scaling of transistor density has resulted in an increase in power density (W/mm^2) beyond the 130nm node, as shown in Figure 1.

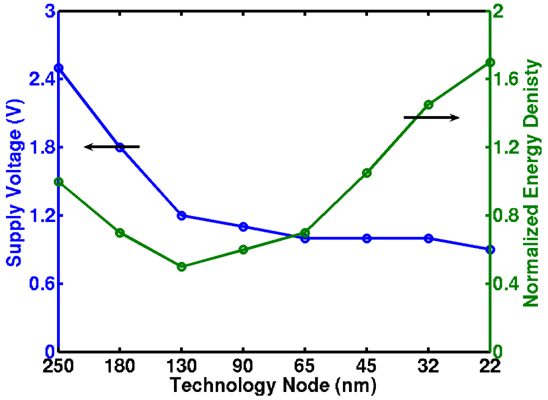

Figure 1: Nominal supply voltage and energy density from 250 nm to 22 nm.

The first consequence of this supply voltage stagnation has been the inability to increase processor frequency while still meeting power density constraints. Instead, processor designs have added more cores without significant increase in their frequency, leading to a prevalence of chip multiprocessor (CMP) [4] in contemporary commercial architectures. However, since the die area of a server class chip has remained approximately constant at ~300–600mm^2, and since the number of cores has been increasing geometrically with each process step, the total chip power has again started to increase, despite relatively flat core frequencies. In practice the maximum allowable power dissipation of a single die is constrained by thermal cooling limits and is roughly 150W without advanced cooling technologies [1]. Hence, the second consequence of supply voltage stagnation is a limit on the number of cores that can be active simultaneously on a die and thus the maximum attainable performance of a modern CMP.

For instance, a 600mm^2 CMP could accommodate 23 Intel Westmere cores [1] in 22nm CMOS, which would dissipate 211W when all simultaneously executing, far exceeding the practical thermal dissipation limit. This would result in 40% of the cores (9 of 23) being idle. The problem of power-constrained core under-utilization has been recently observed in the literature and is sometimes referred to as *dark silicon* [5]. If scaling trends continue to 16nm, a similar CMP would consist of 46 cores, consume a max of 300W, and 50% of the cores (23 of 46) would be idle. As a result the most recent server-class CMPs have incorporated extensive power gating methods to turn off idle cores to free thermal budget for active cores [1].

Because modern CMP performance is now limited by power and not die-area, it is necessary for a paradigm shift in CMP design: cores are plentiful but powering them is not. The overall CMP performance can be best measured as *task throughput*: the number of completed tasks per second. In a power-constrained CMP, the task throughput is limited by the number of tasks that can be simultaneously active on the CMP within the thermal constraint. Thus, if we are able to lower task energy, the number of simultaneous tasks on the CMP (and hence activated cores) can be increased, improving task throughput.

The most effective knob for reducing energy consumption of a task running on a microprocessor is lowering the operating voltage. In tandem, processor frequency is reduced and task completion latency is increased. This type of voltage reduction has been widely used in DVFS, but since it impedes task completion latency, it is not generally applicable for high-performance applications. To address this, parallelization can be used to counteract lower clock frequencies and maintain latency. In this approach, the execution code of the task is parallelized so that the task executes on multiple cores in the CMPs, each operating at a lower frequency and voltage. In this way, the completion time remains the same as when the same task was executed serially on a single core at full voltage, while significant savings in total energy expended for completing the task is obtained. This reduction in energy consumption in turn allows more tasks to be executed on the CMP, thereby increasing overall CMP task throughput.

This combined voltage / parallelization approach is similar to the simpler *circuit* based parallelization approach proposed earlier in [6] which trades-off energy for latency. The method envisioned here instead parallelized the *algorithm* and thereby maintains task latency while still obtaining energy improvement. In fact, with the emergence of CMPs, many key applications are currently being parallelized by software developers. However, parallelization entails a number of overheads, which tend to increase as the task is parallelized into smaller subtasks. These overheads limit the obtainable energy improvement from the proposed approach, as the overhead eventually dominates over the quadratic energy gains from voltage reduction. Hence, there exists a minimum energy point, at which a task is optimally parallelized and voltage scaling reaches its efficiency limit.

To our knowledge, no systematic analysis has been performed to determine where this energy minimum lies. Hence, in this paper, we study

DAC 2012, June 3-7, 2012, San Francisco, California, USA.
Copyright 2012 ACM 978-1-4503-1199-1/12/06 ...$10.00.

this energy minimum, its associated energy gains, core operating voltage and task parallelization. We model three key factors limit energy-efficient parallelization in modern CMOS technologies: The leakage of a transistor — *Leakage Overheads*; the inability to achieve ideal code parallelization — *Amdahl Overheads*; the impact of coherence, interconnect, and memory system design — *Architectural Overheads*. All these overheads are interrelated and limit the obtainable energy efficiency gains from voltage scaling, the optimal energy voltage (V_{opt}) and the number of parallel subtasks required for frequency drop compensation (N_{opt}). In addition, we study the behavior of V_{opt} and N_{opt} across process nodes from 180nm to 32nm technology, using commercial process models.

Our key finding is that when realistic application-dependent overhead is included the optimal operating voltage is near threshold, roughly 200-400mV above the threshold voltage, and that this voltage range is valid across the six generations of industrial technologies as well as across transistor Vt selection. When accounting for all three overheads, operation at V_{opt} yields an energy efficiency gain of ~4× compared to operation at nominal voltage in 32nm and therefore allows a 4× increase in CMP task throughput under thermal constraints. Additionally, we find the maximum amount of energy-efficient parallelism, N_{opt}, across SPLASH2 benchmarks has a median value of approximately 12. Because running at lower supply voltage increases sensitivity to variation, we also explore the impact of variation on V_{opt} and include this in our analysis.

2. SCALING LIMITERS

There are three key limiters to energy-efficient scaling when a task is parallelized to maintain constant latency: leakage, Amdahl, and architectural. Each of these contributes to increased minimum energy and raises the energy-efficient operating point V_{opt}. The three key limiters are analyzed in the following subsections.

2.1 Leakage

First, we will assume a task can be perfectly parallelized across cores to compensate for frequency loss at a lower voltage, and the only non-ideality from running at a slower clock frequency is transistor leakage. It is well known that reducing the supply voltage initially increases energy efficiency of a computation quadratically, yielding dramatic energy efficiency gains [7]. In the last 7 years, it has also been shown that leakage energy poses a *fundamental limiting factor* to energy efficiency gains through voltage reduction [8,9]. The required energy to complete a task can be divided into two categories, dynamic and static, and the classic relationship between energy and operating voltage is:

$$E_{total} = E_{dynamic} + E_{static} = CV_{dd}^2 + I_{leak}V_{dd}T_{task}$$

Dynamic or active energy is the energy consumed in charging and discharging the transistor and interconnect capacitances associated with the task being executed. Static or leakage energy is due to the always present subthreshold and gate oxide currents integrated over the time T_{task} to complete a task. While dynamic energy represents the energy needed to complete a task, static energy is parasitic and only poses an overhead on the computation. Although leakage can be mitigated in standby mode using techniques such as power gating and body biasing, it is more difficult to do so in active mode. Hence, leakage forms an unavoidable and fundamental limit on energy-efficiency.

To understand how leakage energy scales with V_{dd}, clock frequency scaling must be considered since as clock frequency is reduced the time to complete a task increases. For illustration purposes, the relationship between operating voltage and clock frequency is approximately:

$$\frac{1}{T_{task}} \propto f \propto \frac{(V_{dd} - V_t)^{\alpha}}{V_{dd}}$$

where Vt is the threshold voltage and α is process dependent but close to 2. For our results we simulated industrial transistor models in Cadence Spectre to obtain energy and performance. The canonical circuit topology was a chain of 31 fanout-of-4 inverters along with dummy devices for realistic input and output slew rates. The logic activity factor was chosen as 15% to emulate a core where 15% of the logic gates switch on average per clock cycle [10]. In addition chains of other types of logic gates were simulated to confirm that the result obtained for an inverter chain were representative of other logic structures as well.

Initially, when V_{dd} is large relative to Vt, frequency scales proportionately to V_{dd} as shown in Figure 2. As V_{dd} is further reduced and nears Vt, frequency scales exponentially with V_{dd} because the transistor is no longer fully activated. Instead the transistor drive current comes from subthreshold leakage current which scales exponentially with the gate-to-source voltage, and thus exponentially with V_{dd}.

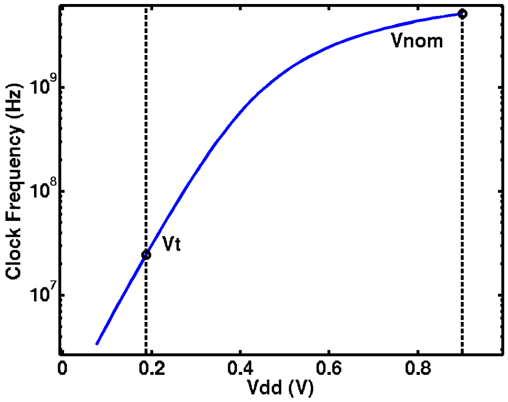

Figure 2: Clock frequency of a logic chain versus operating voltage.

As operating voltage is lowered, the static energy increases since the time to complete a task scales inversely with clock frequency. Eventually, at very low voltages, static energy dominates over dynamic energy, Figure 3. The operating voltage where total energy is minimized is called V_{opt} and occurs when the derivatives with respect to V_{dd} of the two energies are equal, $\frac{dE_{static}}{dVdd} = \frac{dE_{dynamic}}{dVdd}$ [9]. Beyond this point static energy increases more rapidly than dynamic energy decreases, and the total energy increases away from the energy minimum. For a 32nm node, V_{opt} when considering leakage overheads is ~300 mV.

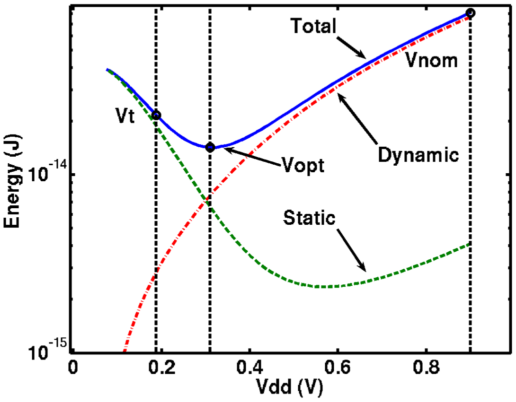

Figure 3: Total, static, and dynamic energy across V_{dd} for a 32nm process.

In recent years, several sensor processors that operate at this V_{opt}, which typically lies below the device threshold voltage, have been designed and demonstrated as much as 10× energy efficiency gains over operation at nominal supply voltage [11,12]. However, these sensor processors also incur phenomenal frequency loss, often operating at clock frequencies of 100s of kHz.

To fully compensate for a frequency loss of *X* because reduced voltage operation, a task with *k* instructions must be parallelized across X cores. If frequency is not compensated the total execution time would increase proportionally to X*k. But, since the task is parallelized, each of the X cores runs k/X instructions so the total execution time is X*(k/X) = k. Thus, no performance is lost from parallelizing. While most scientific and high-performance applications have been parallelized to operate on CMPs, it is not practical to recover a factor of 100's or 1000's in frequency loss without enormous parallelization overheads. Therefore, V_{opt}, considering optimization overheads will be at a higher voltage level, which will be discussed in the next subsection.

2.2 Amdahl

As discussed earlier, scaling voltage is essential in the CMPs to achieve maximum computational performance for a fixed thermal budget, since the number of simultaneous tasks that can fit in a TDP is directly proportionate to the energy efficiency of the task. When scaling supply voltage for a latency-sensitive task, slower clock frequency can be compensated by executing the task in parallel across more cores. For real applications the process of subdividing a task includes non-idealities, such as serial portions of code, and thus incurs parallelization overhead. To compensate a task of *k* instructions for a frequency loss of *X* requires X cores (each running k/X instructions) plus *m* additional instructions of parallelization overhead. These extra m instructions consume additional energy, penalizing lower voltage operation, and therefore increase V_{opt}. Hence, parallelization overheads compound the impact of leakage overheads which limits the voltage scalability of a latency-sensitive task. Compensating below V_{opt} by further subdividing the task results in a net energy increase due to leakage and parallelization overheads.

The well-known Amdahl's law [13] shows that speedup of algorithms as they are parallelized over an increasing number of cores is limited by the parallelizable portion of the code and by new code introduced to initialize and decompose the program. Speedups are therefore bounded asymptotically as parallelization increases because the serial portion eventually dominates. These overheads will be referred to as A*mdahl overheads* and include only the impacts of algorithmic parallelization.

The gem5 [14] system simulator is used to evaluate the impact of Amdahl overheads on a Network-on-Chip (NoC) system. We evaluated the SPLASH-2 benchmark suite which is a set of highly parallelized scientific algorithms applicable to CMPs. Each core is an Alpha architecture with one instruction-per-cycle running at 1GHz. To separate Amdahl overheads from additional architectural non-idealities, we simulated the system with infinite interconnect bandwidth and an ideal memory with 1 cycle latency. Architectural non-idealities are addressed in the next subsection.

The effective speedup of parallelizing by running on 1 to 64 cores is shown in Figure 4. For illustration only three representative benchmarks are labeled, but the entire suite is plotted in the figure. Some benchmarks, such as Barnes, have nearly ideal speedup indicating very little Amdahl overheads and perfect parallelization. Other benchmarks, such as LUNC, reach a speedup of only 10 with 64 cores indicating a high percentage of serial code. These benchmarks represent a range of parallelized scientific applicable to CMPs and, as the number of CMP cores continues to increase, more high-performance applications will be similarly parallelized.

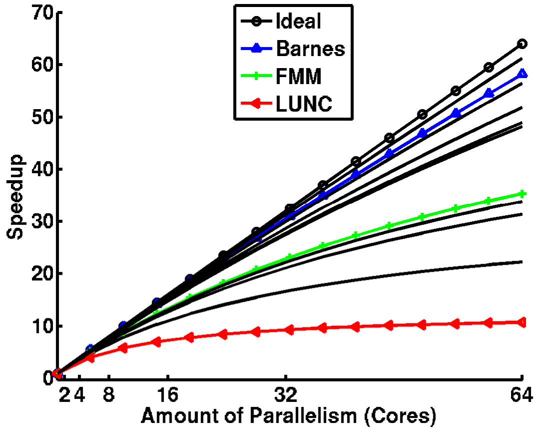

Figure 4: Speedup versus amount of parallelism demonstrating application-dependent Amdahl Overheads.

Amdahl's Law [15] gives $Speedup = \frac{n}{1-P_s+P_s n}$, where n is the number of cores parallelized over and P_s is the Amdahl serial coefficient. We fitted the SPLASH-2 benchmark speedups to Amdahl's law and applied it to the voltage scaling calculations to obtain V_{opt} when considering non-ideal parallelization. The benchmarks were parallelized to fully compensate for frequency loss from lower voltage operation. Figure 5 shows V_{opt} increasing in 32nm due to Amdahl overheads. When Amdahl overheads are added, the V_{opt} operating range for most overheads is ~25-150 mV above the leakage overheads only case. Although the serial coefficient is highly application-dependent, the range of V_{opt} for the benchmarks is small, varying by only ~150mV. If the serial coefficient is 100% (e.g., none of the code is parallelizable) then nominal voltage would be optimal.

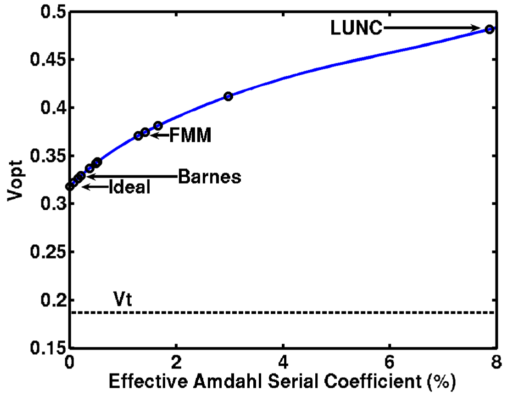

Figure 5: V_{opt} vs. Amdahl coefficient for all SPLASH-2 benchmarks (three labeled) in 32nm.

2.3 Architectural

Architectural features, such as coherency, inter-core communications, and cache pollution, further add overhead to a CMP system as voltage is reduced and a task is parallelized. Furthermore, application memory access patterns can affect overhead. For example, a subtask competes for L2 cache resources and may evict another subtask's data. Coherence overhead is added when a multiple subtasks share a single block of data. Communication overhead is increased when there is heavy communication between distant cores on an NoC because data must transverse multiple hops.

To quantify architectural overheads, the SPLASH-2 benchmarks were simulated with gem5 as in Section 2.2 but the configuration was changed to add non-ideal memories, caches, and interconnect. The NoC simulations were run using a tiled Mesh topology where each tile contains a core, private L1 caches, and a slice of a shared L2 cache. A MOESI directory protocol is used to maintain coherence. Table 1 lists the detailed simulation parameters.

Feature	**Description**
Cores	1 to 64 one-IPC Alpha cores @ 1GHz
L1 Caches	32 kB, 1 cycle latency, 4-way associative, 64-byte line size
L2 Caches	Shared 1MB divided evenly between cores, 10 cycle latency, 8-way associative, 64-byte line size
Interconnect	2-GHz Routers, 128-bit, 2-stage routers, 50 cycle-access to main memory

Table 1: A list of simulation parameters to measure architectural overheads.

Architectural overheads from memory and interconnect non-idealities reduce the obtainable speedup when parallelizing. These non-idealities were added in the V_{opt} calculation when parallelizing and the benchmarks were again parallelized to fully compensate for frequency loss. Like leakage and Amdahl overheads, architectural overheads further increase the minimum energy consumption and V_{opt} as shown in Figure 6. Certain benchmarks are highly parallelizable before caches and coherency is introduced, while others have negligible architectural overheads. For example, ocn has almost no Amdahl overheads but significant architectural overheads. To contrast, lun has little architectural but significant Amdahl overheads. Across the benchmarks shown V_{opt} increases by no more than 200mV. Thus, architectural overheads are another key limiter to voltage scaling, increasing V_{opt} and the minimum obtainable energy consumption when a task is parallelized to compensate for frequency loss. Though not included in this paper for brevity, we also simulated SPLASH-2 running on a bus-based architecture and the corresponding increase in V_{opt} is similar.

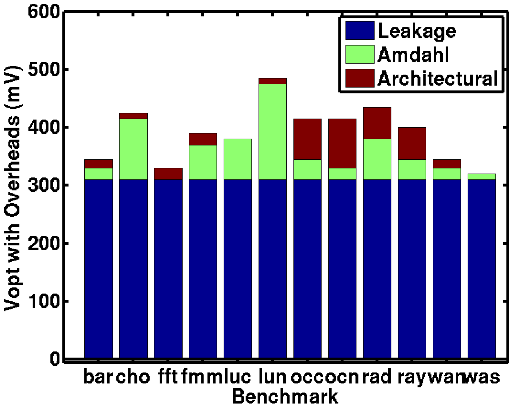

Figure 6: Architectural and Amdahl overheads increase V_{opt} by no more than 200 mV.

3. IMPACT OF TECHNOLOGY AND CIRCUIT FEATURES ON NTC

The previous section discussed the three key limiters of energy-efficient scaling. However, V_{opt} is also impacted by additional technology and circuit factors, including technology node, transistor Vt, and process variation, which are discussed below.

3.1 Technology

In the previous section V_{opt} was analyzed at single 32nm technology node. To identify if there is a voltage scaling and parallelization guideline consistent across many technologies, we calculated V_{opt} for SPLASH-2 across 6 industrial technologies when accounting for all three voltage scaling overheads, as shown in Figure 7. Circuit simulations of energy and performance were done in Cadence Spectre using industrial foundry technology kits from 32nm to 180nm.

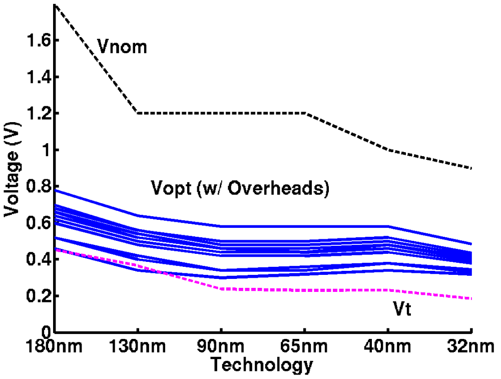

Figure 7: V_{opt} across technologies when including all three overheads. V_{opt} has been trending downward with each generation and is ~200-400mV above Vt for most benchmarks.

The process node affects V_{opt} primarily because technologies have become more leaky generation-to-generation due to reduced threshold voltage. Higher leakage increases V_{opt}, however, the lower threshold voltage will also improve the frequency degradation with voltage scaling which will reduce V_{opt}. A key finding of this work is that V_{opt} consistently tracks ~200-400mV above the threshold voltage for most benchmarks across the six technology nodes. We define this region above the threshold voltage as the near-threshold (NTC) region. Three benchmarks, Barnes, FFT, and Water Spatial, were close to ideally parallelizable and are not contained in the NTC region. However, most general-purpose, high-performance CMP applications will have some degree of parallelization overhead and thus lie in the NTC region.

The median energy gains at V_{opt} operation and optimal number of cores to parallelize across to compensate for clock frequency loss, N_{opt}, for SPLASH-2 across technology nodes is shown in Figure 8. Table 2 includes a breakdown of energy gains and optimal number of cores for each benchmark in the SPLASH-2 suite. Energy gains have diminished by ~1.8× from 180nm to 32nm as leakage has increased and the dynamic range available for voltage scaling has narrowed from 180nm to 32nm. This difference is less dramatic with less scalable benchmarks, since the parallelism overheads are higher and thus the amount of voltage scaling in older technologies is limited. The energy gains in newer technology from operating at V_{opt} instead of at nominal voltage are ~4× and N_{opt} has a median of ~12, and no more than 25, cores for SPLASH-2 in 32nm. This increased energy efficiency directly increases CMP performance when limited by a thermal budget. Thus, to maximize thermally-limited CMP performance, tasks should operate in the near-threshold region and parallelize on no more than 25 cores in 32nm. The energy gains and optimal amount of parallelism has decreased with each generation.

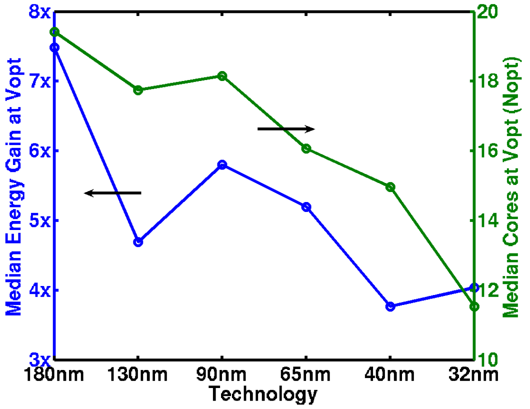

Figure 8: Median energy gains and optimal number of cores, N_{opt}, when operating at V_{opt} as compared to nominal voltage for SPLASH-2 benchmarks.

	180nm	130nm	90nm	65nm	40nm	32nm
bar	12.7× (71)	7.9× (55)	10.0× (69)	7.8× (43)	5.7× (44)	5.2× (19)
cho	6.1× (13)	3.9× (10)	4.6× (14)	4.3× (12)	3.2× (11)	3.5× (9)
fft	13.2× (68)	8.3× (82)	10.4× (67)	8.0× (42)	5.8× (43)	5.2× (23)
fmm	7.7× (21)	4.8× (20)	6.0× (19)	5.3× (16)	3.9× (16)	4.1× (12)
luc	8.6× (26)	5.3× (25)	6.7× (24)	5.9× (18)	4.2× (21)	4.4× (13)
lun	4.1× (8)	2.7× (6)	3.1× (7)	3.0× (6)	2.3× (6)	2.6× (6)
occ	6.9× (14)	4.3× (16)	5.3× (17)	4.8× (14)	3.5× (13)	3.8× (9)
ocn	6.8× (15)	4.2× (12)	5.2× (17)	4.7× (14)	3.5× (14)	3.8× (9)
rad	5.7× (11)	3.6× (12)	4.3× (12)	4.0× (10)	3.0× (9)	3.4× (8)
ray	7.3× (17)	4.6× (15)	5.6× (16)	5.0× (17)	3.7× (13)	4.0× (11)
wan	12.6× (71)	7.8× (56)	9.9× (70)	7.8× (32)	5.7× (45)	5.2× (19)
was	18× (186)	11.3× (250)	13.0× (121)	9.0× (51)	6.8× (79)	5.5× (25)

Table 2: Energy gain and optimal number of cores N_{opt} (in parenthesis) across SPLASH-2 benchmarks and technologies when including the three voltage scaling overheads.

If Amdahl and architectural overheads can be neglected because an application is latency insensitive (for instance, sensor applications) then only the fundamental leakage overhead needs to be considered. To provide a comparison with the trend of V_{opt} for latency-sensitive applications, we show in Figure 9 the fundamental lower bound on V_{opt} across technologies, where leakage is the only voltage scaling overhead. In 180nm and 130nm V_{opt} for a perfectly parallelizable task is below threshold. Because technologies are becoming leakier with process scaling, V_{opt} has been trending upward with each generation and becomes super-threshold in 90nm. For a perfectly parallelizable task the energy gain has decreased from 52× in 180nm to 6× in 32nm. Likewise, the optimal number of cores N_{opt} has decreased from ~21,000 (clearly unachievable) in 180nm to 29 in 32nm. Though parallelizing across thousands of cores in older technologies is not achievable, the gains and N_{opt} in recent technology nodes have dramatically decreased even when neglecting Amdahl and architectural overheads.

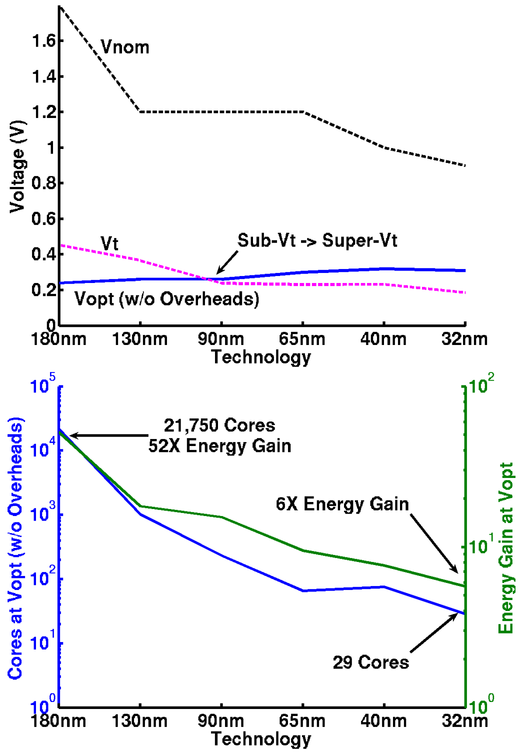

Figure 9: Theoretical maximum energy-efficient parallelism, showing V_{opt}, N_{opt}, and gains across six technology nodes with leakage overheads only.

3.2 Process Variation

A challenge of operating at a reduced voltage is increased sensitivity to process, temperature, and supply voltage variations that causes variability in circuit delay and energy consumption. A slower critical path and leakier devices decrease energy-efficiency thus increasing V_{opt}. Figure 10 (top) shows the 3-sigma delay variation relative to mean for a single gate and a chain of logic in 40nm technology using industrial variation models. Process variation can be global, affecting all transistors uniformly across a die, or local which causes delay mismatch between different devices and paths on a chip.

In the NTC region, local variation accounts for 30% of 3-sigma delay of a single gate. Since a CMP's maximum clock frequency is limited by the worst-case critical path, mismatch between different critical paths raises V_{opt} as the leakiest path runs at clock frequency set by the slowest path. However, local variation is reduced for deeper logic depths since local variation is usually uncorrelated and hence averages out along a path. Thus, for a chain 31 gates, local variation is only 10% of total variation, so its impact is minor.

Global variation raises V_{opt}, Figure 10 (bottom, for three technology nodes) but all paths will either: (1) slow and have less leakage or (2) have more leakage but run fast, so the total leakage overhead is relatively constant. This is unlike local variation where the leakiest path is run at the slowest clock frequency, thus global variation's contribution to increasing V_{opt} is less than local variation. Total delay variation at V_{opt} is significant, but high-performance CMPs are usually binned for speed so that each die can run at its optimal frequency. Thus, the range of bins will increase but, since each die is tuned to its optimum speed, global variation does not significantly increase V_{opt}.

The increase in V_{opt} when considering 3-sigma delay variation and the parallelization overheads described above is 30mV-60mV for an average SPLASH-2 benchmark. The delay variation also depends on the number of critical paths in a design, since local variation reduces by taking a maximum across multiple paths. As the number of paths increases the mean shifts up, but the variation is reduced, shown in Figure 10 (bottom). Thus, variation does impact delay but its impact on V_{opt} and minimum energy are small.

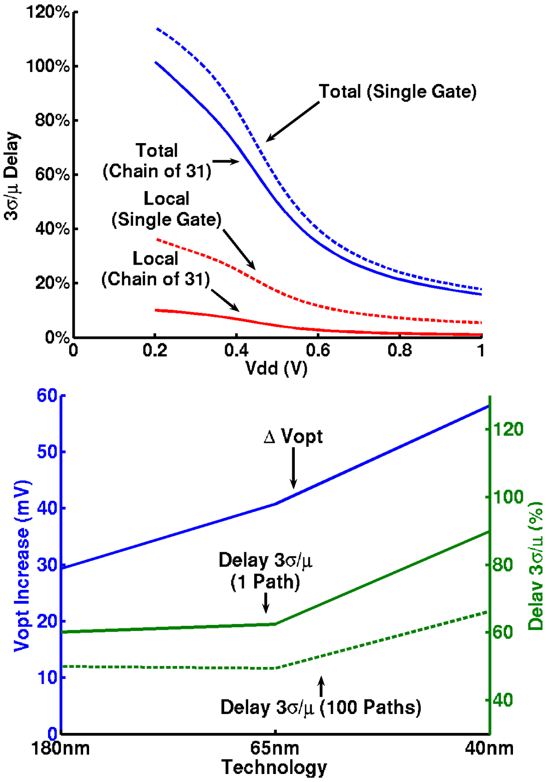

Figure 10: Change of delay for total and local process variation of a single gate and a logic chain of 31 gates (top). Increase in V_{opt} because of 3-sigma variation across generations (bottom).

3.3 Transistor Threshold Voltage

The energy-efficient operating voltage V_{opt} also depends on transistor threshold voltage selection. Conventionally regular threshold voltage transistors are used for high-performance applications, since they have the best drive strength, whereas the higher threshold voltage transistors are used where low static-power is a concern, such as in mobile applications.

When considering a parallelized task, Amdahl and architectural overheads limit the energy-efficiency and voltage scalability, thus setting V_{opt}, Figure 11 (top). As threshold voltage is reduced, leakage begins to dominate until the voltage scalability is limited by leakage overheads and not Amdahl or architectural overheads. The energy drops initially as threshold is reduced, since the task can run faster, and V_{opt} correspondingly tracks. Once leakage dominates the energy stays relatively constant. As a rule-of-thumb, the optimal threshold voltage is at the inflection point (~250mV in figure) between the parallelism-dominant and leakage-dominant region, since above this point energy increases and below this point the process becomes unnecessarily leaky.

For comparison, Figure 11 (bottom) shows V_{opt} when a task only includes leakage overheads. Since there is no Amdahl or architecture overheads, V_{opt} lowers as threshold voltage increases, since the integrated leakage current is reduced, until V_{opt} enters the subthreshold regime. Once V_{opt} is subthreshold, raising the threshold voltage does not change the energy-efficient operating voltage or energy consumption to first-order [8].

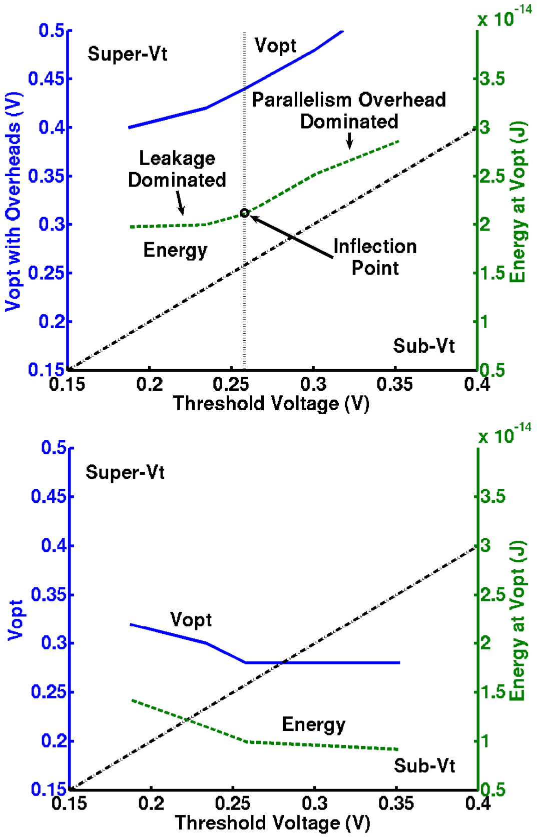

Figure 11: V_{opt} vs. Vt in 32nm for with Amdahl and architecture overheads (top) and leakage only (bottom).

Tasks that are not latency sensitive can operate in subthreshold with high Vt transistors, but this is not optimal for latency-sensitive applications. To achieve maximum performance in latency sensitive applications, even when limited by a thermal budget, the threshold voltage should be reduced until leakage starts to dominate voltage scalability.

4. CONCLUSIONS

We have detailed the limits of voltage scaling for latency-sensitive applications, when slower clock frequency is compensated by parallelization across multiple cores. As CMPs become limited by thermal cooling constraints, near-threshold operation is needed to maximize computations for a fixed thermal design power. The three voltage scaling limiters, leakage, Amdahl, and architectural, contribute to increasing the minimum energy and optimal supply voltage V_{opt} to maximize total CMP performance. As a guideline, the near-threshold region for maximum energy-efficiency is roughly 200mV-400mV above threshold voltage for most applications and this trend held for the six technology nodes we examined.

NTC operation increases energy-efficiency of a core by approximately 4× in 32nm for the SPLASH-2 benchmarks we investigated, roughly translating to a 4× improvement in performance for a thermally-limited CMP. Additionally, the maximum amount of energy-efficient parallelism is no more than 25 cores in 32nm. Delay variation increases in the NTC region, but has little impact on V_{opt}. For latency sensitive applications threshold voltage should be minimized until leakage dominates voltage scalability, whereas latency insensitive applications benefit from subthreshold operation.

5. ACKNOWLEDGEMENTS

The authors gratefully acknowledge the National Science Foundation and ARM Holdings for their support of this project.

6. REFERENCES

[1] G. Moore, “No exponential is forever: But ‘forever’ can be delayed!” *IEEE International Solid-State Circuits Conference* Keynote address, 2003.

[2] N. Weste, D. Harris. *CMOS VLSI design: a circuits and systems perspective. (3rd edition)*, Location: Boston, MA, 2005.

[3] R. Dennard, F. Gaensslen, V. Rideout, E. Bassous, and A. LeBlanc. “Design of ion-implanted mosfet’s with very small physical dimensions,” Solid-State Circuits, IEEE Journal of, 9(5):256 – 268, Oct. 1974.

[4] S. Sawant, et. al., "A 32nm Westmere-EX Xeon® enterprise processor," *Solid-State Circuits Conference Digest of Technical Papers (ISSCC), 2011 IEEE International* , vol., no., pp.74-75, 20-24 Feb. 2011

[5] H. Esmaeilzadeh, E. Blem, R. St. Amant, K. Sankaralingam, and D. Burger, “Dark Silicon and the End of Multicore Scaling,” *The 38th Annual International Symposium on Computer Architecture ISCA*, June 2011.

[6] A. Chandrakasan, S. Sheng, and R. Brodersen, "Low-power CMOS digital design," JSSC, vol. 27, no. 4, Apr. 1992, pp. 473-484.

[7] A. Wang and A. Chandrakasan, “A 180mV FFT processor using subthreshold circuit techniques,” *IEEE International Solid-State Circuits Conference*, pp. 292-529, 2004.

[8] B. Zhai, D. Blaauw, D. Sylvester, and K. Flautner, "Theoretical and practical limits of dynamic voltage scaling," *Design Automation Conference, 2004. Proceedings. 41st,* pp. 868 - 873, Jan 1 2004.

[9] L. Nazhandali, B. Zhai, R. Helfand, M. Minuth, J. Olson, S. Pant, A. Reeves, T. Austin, and D. Blaauw, "Energy Optimization of Subthreshold-Voltage Sensor Processors," *The 32nd Annual International Symposium on Computer Architecture ISCA*, Madison, Wisconsin USA, June 4-8, 2005.

[10] J.M. Rabaey, A.P. Chandrakasan, and B. Nikolic, *Digital Integrated Circuits: A Design Perspective*, Prentice Hall, 2003.

[11] S. Hanson *et al*., “Performance and variability optimization strategies in a sub-200mV, 3.5pJ/inst, 11nW subthreshold processor,” *Symposium on VLSI Circuits*, pp. 152-153, 2007.

[12] M. Seok, S. Hanson, Y. Lin, Z. Foo, D. Kim, Y. Lee, N. Liu, D. Sylvester, D. Blaauw, “The Phoenix Processor: A 30pW platform for sensor applications,” *IEEE Symposium on VLSI Circuits*, pp. 188-189, 2008.

[13] Amdahl, Gene (1967). "Validity of the Single Processor Approach to Achieving Large-Scale Computing Capabilities". *AFIPS Conference Proceedings* (30): 483–485.

[14] N. Binkert, B. Beckmann, G. Black, S. K. Reinhardt, A. Saidi, A. Basu, J. Hestness, D. R. Hower, T. Krishna, S. Sardashti, R. Sen, K. Sewell, M. Shoaib, N. Vaish, M. D. Hill, and D. A. Wood, “The gem5 simulator,” *Computer Architecture News (CAN)*, June 2011.

[15] G. M. Amdahl. "Validity of the single processor approach to achieving large scale computing capabilities," *AFIPS '67 Proceedings*, April 1967