Navigo: An Early-Stage Model to Study Power-Constrained Architectures and Specialization

Mark Hempstead, Gu-Yeon Wei, David Brooks
School of Engineering and Applied Sciences, Harvard University
{mhempste, guyeon, dbrooks}@eecs.harvard.edu

Abstract

As the number of transistors double, it becomes difficult to power all of them within a strict power budget and still achieve the performance gains of that the industry has achieved historically. This work presents, Navigo, a modeling framework for architecture exploration across future process technology generations. The model includes support for voltage and frequency scaling based on ITRS and PTM models. This work is designed to aid architects in the planning stages of next generation microprocessors, by addressing the space between early-stage back-of-the-envelope calculations and later stage cycle accurate simulators. Using parameters from existing commercial processor cores, we show how power consumption limits the theoretical throughput of future processors. Navigo shows that specialization is the answer to circumvent the power density limit that curbs performance gains and resume traditional 1.58x performance growth trends. We present analysis, using next generation of process technologies, that shows the fraction of area that must be allocated for specialization to maintain performance growth must increase with each new generation of process technology.

1. Introduction

Advances in computational capabilities have driven the information technology revolution, which in turn has driven advances in nearly all fields of science, medicine, and business. Although incredibly powerful computing devices are available today, this single-minded pursuit of performance has led to power consumption emerging as one of the main bottlenecks for nearly all types of computing systems, from high-end servers to wireless sensor devices. Due to limitations in device cooling at the high-end and battery technology at the low-end, processor designs are increasingly stratified into power-constrained market segments in which the challenge is to increase processor performance for a fixed power budget. While advanced fabrication technology will continue to provide computer designers a doubling of transistors per generation, slowing of constant-field scaling and worsening wire parasitics will see the energy per switching event scale at a rate in which chip power will essentially remain constant with fixed clock frequency and core activity. Current trends towards large multi-core systems utilize the additional transistor bounty for additional power-efficient cores, but with single-thread performance saturating, most benefits will come through thread-level parallelism. Assuming an optimistic scenario for continued extraction of thread-level parallelism from workloads, chip performance gains will track growth in transistor counts. ITRS projects a doubling in the number of transistors every three years (e.g. 1.25x per year) leading to an increasing gap between projected performance growth and historical performance growth rates. Bridging this performance gap will require an architectural paradigm shift to augment the multi-core trend, in which an increasing fraction of a chip real estate must be devoted to specialized logic that provides significant benefits in performance per switching event for a growing portion of workloads.

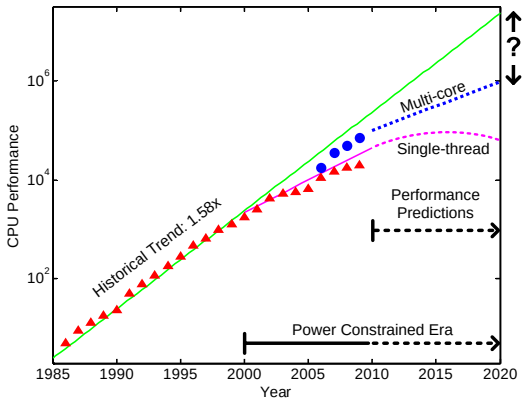


Figure 1. Growth in Microprocessor Performance. *Historically the industry has observed a total 1.58x performance gain. Power consumption constraints inhibits performance growth causing a gap between expected and delivered performance. Data from Hennessy and Patterson* [3] *and* spec.org [12].

To further explore these trends, Figure 1 plots both historical performance growth and projected multi-core and single-threaded performance growth until 2020. All data in the plot is relative to the VAX 11/780 as measured by SPECint

benchmarks – data in the plot previous to 2005 was obtained from [3], and data for recent years was obtained using the highest single-die performance SPECint2006 (single-thread) and SPECint2006rate (multi-core) from the SPEC website [12]. Performance growth began to deviate from the historical 1.58x per year trend in 2001, primarily due to the difficulty in obtaining clock frequency and instruction-level parallelism improvements in the face of power constraints. The computing industry has reacted to this trend by concentrating on multi-core designs that capture thread-level parallelism. Unfortunately, as detailed in this paper, power issues will limit multi-core performance growth from meeting the historical trend, and closing this gap will require more efficient use of transistors.

Given these trends, it is important for chip architects to understand the limitations of homogeneous parallelism and to consider more radical architectural approaches. This paper presents *Navigo*, a model that incorporates technology scaling effects to predict future power-constrained performance trends. Navigo can be used to predict, for a variety of processor cores, circuit parameters, and market segments, performance trends and shortfalls from the historical growth rate. Future designs that seek to bridge this gap must more effectively utilize switching events through specialized hardware. Specialization hardware can take many forms [1, 7, 6, 8] including programmable SIMD units, hardcoded ASIC cores, or reconfigurable logic, and Navigo includes a general analytical model that can capture the impact of parallel specialization on power-constrained performance gains. This model projects the amount of specialization, quantified in terms of several parameters, that will be required in future technology generations to meet the historical performance scaling trends. This modeling infrastructure can be used by designers to evaluate next generation architectures before the construction of more detailed cycle accurate simulators.

2. Navigo: A Model for Performance Trends in Future Technologies

Navigo aims to provide designers with a powerful and yet flexible tool to navigate the intricate tradeoffs between process technology, circuits, and architecture, in order to predict their implications on performance in future processor designs. Figure 2 presents a high-level graphical representation of the modeling infrastructure. The model takes in a variety of input libraries, which quantify detailed parameters corresponding to process technology, circuit performance, architecture, and market segment constraints. While each of these libraries can be modified by the user, Navigo includes built-in libraries based on ITRS technology scaling predictions out to 11nm (available in 2020), predictive technology models (PTM) [10, 16], IPCs of currently available processor cores (based on SPECint2006 scores), and high-level power and area constraints for different market segments. With the libraries in place, the designer can sweep a variety of input parameters such as technology node, voltage, frequency, target market, etc. Navigo then outputs the total system throughput and power. The user can then refine her design by iterating through different input parameters to meet a specific throughput and/or power target.

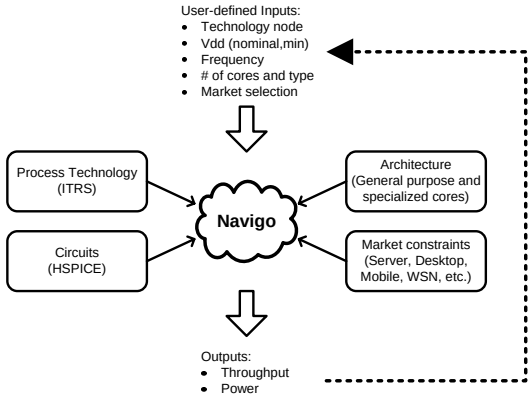


Figure 2. Graphical depiction of Navigo. *The model accepts library files for process technology, circuits, architecture, and market segments, and computes total and constrained power for a set of user-defined inputs such as supply voltage, frequency, etc.*

At the core of the model is an engine that takes the various libraries and input sweep parameters to calculate throughput and power consumption. This engine must consider a variety of factors such as the number and characteristics of computational blocks (i.e. cores), voltage and frequency scaling, wire loading, leakage power, and process technology, all constrained by power budget limitations. All of these factors are quantified by the different library parameters.

The process technology library quantifies several parameters and characteristics utilized by Navigo, which are listed in Table 1. These parameters set the basic device and wire characteristics that Navigo uses to determine circuit speed, power, and the number of cores that will be available in future technology nodes. The built-in process technology library uses published data from ITRS 2007 [10, 16] out to the 11nm technology node anticipated in year 2022. ITRS predicts double gate technology will supplant planar bulk devices at the 32nm node in year 2013. Because ITRS is a predictive roadmap based on current projections of technology, it is well-known that the semiconductor industry has a history of either under- or out-performing ITRS. For example, Intel's technology roadmap is more aggressive with processors at the 45nm node already shipping and plans to introduce processors on the 32nm node in late 2009. Hence, this library can be readily modified by the user to better reflect updated ITRS projections, or propriety information if available.

The circuits library utilizes predictive technology models (PTM) [10, 16], available from the 45nm node down to 16nm, to model how power and frequency scale with supply voltage and different amounts of wire parasitics. In the absence of detailed circuit blocks that can be simulated, we rely on

Year of Production	2007	2010	2013	2016	2019	2022
	Planar Bulk		Double Gate			
Approximate node (nm)	65	45	32	22	16	11
Supply Voltage (V)	1.1	1.0	0.9	0.8	0.7	0.65
Physical Gate Length (nm)	25	18	13	9	6.3	4.5
Id sat (uA/um)	1211	1807	2204	2627	2768	2786
Intrinsic delay (ps)	0.64	0.46	0.26	0.15	0.1	0.08
Intrinsic switching energy (fJ)	0.0639	0.0449	0.0201	0.00851	0.00367	0.00196
RC delay of 1mm wire (ps)	890	2100	4555	10652	23515	58525
Die Size-Server (mm^2)	310	310	310	310	310	310
Number of Transistors (M)	1106	2212	4424	8848	17696	35391

Table 1. Predicted Process Technology Characteristics. *High-Performance Microprocessor Technology ITRS 2007 Edition* [11].

HSPICE simulations of fanout-of-4 ring oscillators across the technologies to determine basic frequency, power, and voltage trends. We combine ITRS predictions with PTM-based simulations to extrapolate trends at the 11nm node. These trends allow Navigo to scale voltage and frequency to meet different power budgets. It is also important to consider the effects of imposing minimum voltage (VddMIN) constraints since allowing arbitrary reductions in supply voltage can lead to a variety of issues related to 6T SRAM cell instability issues [15] and exacerbation of on-chip voltage noise.

The architecture library contains a collection of processor cores that the user can choose to tile together in future multi-core systems. The built-in architecture library consists of three cores currently in production, listed in Table 2. These cores, Intel Xeon (Netburst), Intel Core2Duo (Core), and Intel Atom, represent high-end server, desktop, and mobile CPUs. We plan to include analysis for processors such as Intel's Core i7, as detailed information becomes available. Parameters for the processors were obtained from publications and SPEC scores in spec.org for Xeon and Core2Duo. Since official SPEC results are not available for Atom, we extrapolate based on benchmark comparisons between Atom and an Athlon with known SPEC scores [14]. While different processors have been implemented with different technologies, the power, performance, and area of each core is appropriately scaled by Navigo utilizing the process technology and circuits trends prescribed by their respective libraries. The user is not constrained by these cores, but can also include other user-defined cores into the architecture library. For example, Section 5 explores the impact of specialized cores.

The market segment library identifies different market segment targets that constrain total area and maximum power. Table 3 lists examples of different market segments. Throughout the rest of the paper, we focus on two particular market segments—server and mobile. The server market allows for a maximum area of $310mm^2$ and maximum power of 198W as defined by ITRS. In contrast, the mobile market allows for a maximum area of $100mm^2$ and maximum power of 35W. Again, different markets segments and/or constraints can be easily defined by the user via changes to the library.

Finally, Navigo's engine computes total throughput as follows:

Market	Max Power (W)	Die Area (mm^2)
MPU-CP Cost and Performance	151	140
MPU-HP High Performance	198	310
MPU-PCC Power Cost and Connectivity	3	70
Desktop-95	95	100
Desktop-65	65	100
Mobile Standard Voltage	35	100
Mobile Ultra-low Voltage	10	100

Table 3. Market Segment Constraints. *Die size and Max Power Consumption for a set of market segments. Values for the first three markets came from ITRS* [11]. *The final four market segments are based on die size and thermal design point of commercially available Intel Processors.*

$$Throughput = N_{cores} * freq(Vdd, tech) * IPC_{core} \quad (1)$$

where the number of cores, N_{cores}, is defined by the total die size (for a target market segment) divided by the core chosen and scaled by technology node. The IPC of each core can be derived from published (or simulated for new cores) SPEC benchmark results and clock frequency of the core. Operating frequency depends both on process technology and voltage, and is calculated based on the original frequency published for the core. First, Navigo calculates the maximum frequency of the core for nominal voltage in the new technology. We incorporate both the intrinsic switching delay of the transistor and effects due to wire delay scaling.

$$freq_{VddNom} = freq_{core_{basetech}} * \left(frac_{logic} * \frac{freq_{switch_{tech}}}{freq_{switch_{basetech}}} + frac_{wire} * \frac{freq_{wiretech}}{freq_{wire_{basetech}}} \right)$$

where $basetech$ is the original technology in which the core was fabricated. The nominal frequency is then multiplied by PTM-based scaling factors to calculate voltage-specific frequencies.

Power depends on voltage, operating frequency, and the transistor switching rate of the architecture. Average power

Processor	Technology (nm)	Total Die Size (mm^2)	Number of Cores	Vdd (V)	Freq (GHz)	Power (W)	IPC (SPEC2006/GHz)
Intel Xeon (Tulsa) [13]	65	435	2	1.25	3.4	110	3.72
Intel Core2Duo (Wolfdale)	45	107	2	1.36	3	65	6.82
Intel Atom [2]	45	25	1	1.0	2.0	2.0	2.35

Table 2. Example Cores used in analysis. *Data collected from conference and journal publications and datasheets. SPEC2006 results used to determine IPC are from* spec.org.

can be modeled with the following expression:

$$P_{avg} = P_{active} + P_{leak} \Rightarrow$$
$$freq * (E_{switch} * N_{switching} + E_{wire}) + P_{leak}$$

We calculate switching rate ($N_{switching}$) from published frequency and power numbers. Since energy per switch (E_{switch}) is technology dependent, it scales based on voltage-dependent scaling factors derived from HSPICE simulations for each technology node. Wires scale differently from transistors and, hence, are separately accounted for. We assume leakage power remains a fixed percentage of the total power consumption at maximum frequency and nominal voltage, which then scales with respect to different operating voltage levels. In order to accommodate different power budgets prescribed by different market segments, the model iterates through voltage and frequency settings until a specific power target is met. When the model encounters a VddMIN constraint, it only scales frequency to reduce power at the expense of inefficient energy usage.

While Navigo seeks to combine a variety of factors to accurately predict future performance, it makes several optimistic assumptions. First, it may not feasible to fit an integer number of cores into a predefined area. Hence, we allow for half-size cores with IPC and power that scale linearly by one half. Although this scenario is unfeasible, for near-term technologies (e.g. 45nm), large area cores introduce quantization effects which make it difficult to observe consistent trends. This effect becomes significantly less important as we scale to more advanced technologies. Second, future multi- and many-core systems will face a variety of challenges to enable core-to-core communications. Navigo optimistically assumes a perfect on-chip interconnection network. Lastly, and perhaps most important, we assume workloads can be fully parallelized to keep all cores running continuously. Hence, the model is orthogonal to Hill's investigation that compares single-threaded versus multi-threaded parallelism [4]. One of the main objectives of developing Navigo was to provide a detailed and yet flexible model to help designers predict performance trends and guide future designs before cycle accurate simulators are available. Moreover, we use this model to show that despite optimistic assumptions of perfect thread parallelism that are run on highly-parallel many-core designs, power constraints will hamper performance growth and motivate designers to seek out new solutions beyond simply increasing the number of cores on a die.

We have implemented Navigo as a set of Matlab scripts for the main engine and additional scripts to extract data for the libraries. The circuits library was developed from several thousand CPU hours of HSPICE simulations. We have developed additional scripts for complex studies that incorporate thousands of individual Navigo results, such as the analysis shown in Section 5. Our eventual goal is to package the system in a form usable by the architecture community.

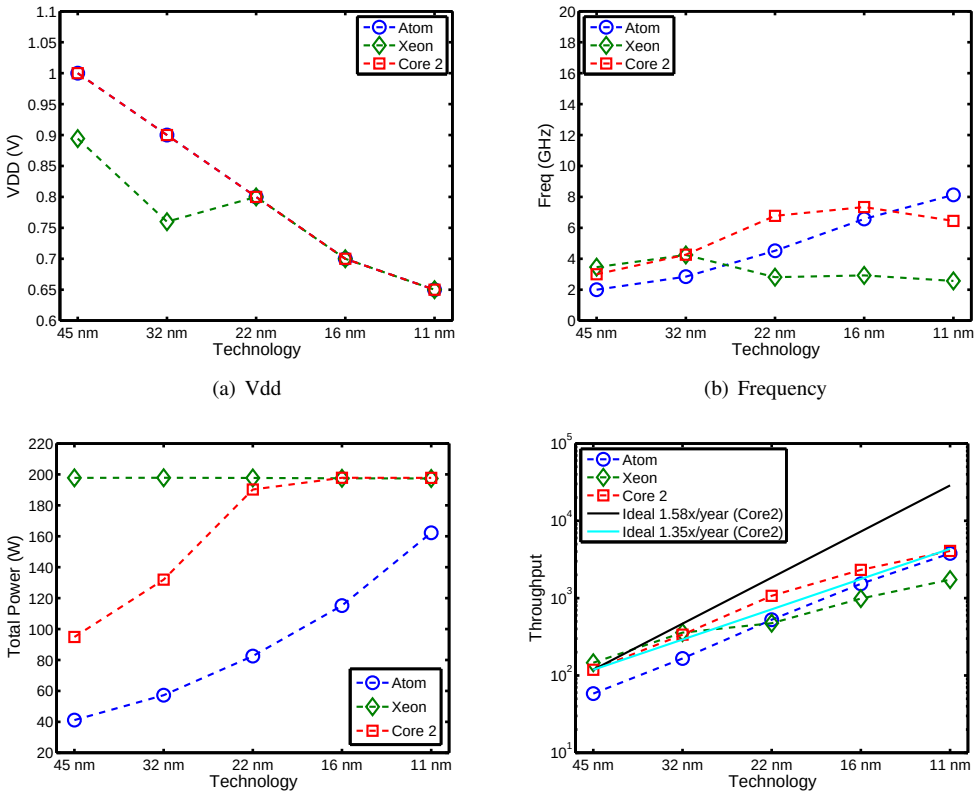
3. Power-constrained performance estimates

Navigo can be used to understand power-constrained performance scalability across technology generations. In this section, we demonstrate the utility of the model by exploring the scalability of three classes of CPU architectures when considering power-constrained market segments (Table 3) and the impact of the minimum supply voltage constraint.

For each of these explorations, we make several assumptions. First, we assume that area and power will be fixed by the market segment. More advanced technology nodes provide an increase in the number of available transistors leading to a doubling of available cores per technology generation; however, frequency benefits will be constrained by power limits. If the power budget is exceeded for a given number of cores and clock frequency, we scale voltage and frequency down to meet the power budgets, subject to circuit constraints on the supply voltage, after which linear frequency scaling is utilized.

3.1. Results without Power Constraints

To understand the impact of power constraints on scaling, we first consider the scenario where power is *not* a design constraint. We evaluated our model and reported core types, the number of cores, clock frequency, total power, and total chip throughput for a fixed area budget of 310 mm^2. The figures are not included due to space constraints. Without power limitations, frequency scaling continues unabated surpassing 19.12 GHz for the Xeon core in 11nm, but this comes at the price of increased power dissipation, exceeding a kilowatt in the worst case. The throughput improvement increases at a slightly lower rate than the historical growth rate of 1.58x. This shows that if power is not a constraint, performance growth could be achieved through a combination of traditional frequency scaling and multi-core design.



(a) Vdd (b) Frequency

(c) Power (d) Throughput

Figure 3. Results with power constraints across process technologies - Server. *Results assume nominal voltage for specified technology and MPU-HP market segment with a die size of 310 mm^2 and max power of 198 W.*

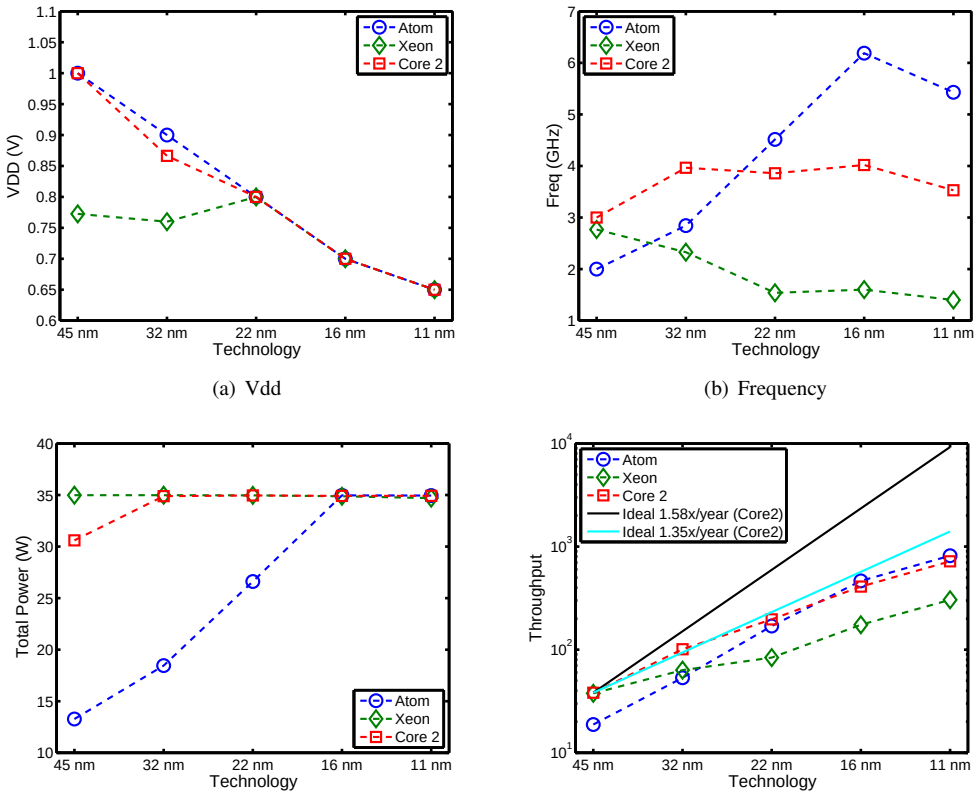
3.2. Results with Power Constraints

Incorporating power constraints into our analysis gives a true picture of expected trends in future technologies. We show that for market segments that tolerate higher power density systems, scaling trends are better compared to more constrained market segments. In this section, we compare the server market segment, which uses the same 310 mm^2 die with a power limit of 198W, and the mobile market segment, which uses a 100 mm^2 die with a power limit of 35W. Figure 3 and Figure 4 plot the server and mobile market segment scalability analysis across the three core types. Each plot shows the required supply voltage, clock frequency, total power, and total chip throughput.

Focusing on the results for the server market segment, we observe several important trends. For the Intel Xeon design, power is constrained beginning at the 45nm technology node, and the design must reduce supply voltage from nominal in order to meet the power goal. When moving to the 32nm node, the Xeon is able to achieve a small frequency increase by operating at the minimum supply voltage. Beyond 32nm, the Xeon frequency reduces slightly and then flattens out as the power budget is soaked up by additional cores. In contrast, the Intel Core2Duo design allows full frequency scaling until the 22nm technology node, after which scaling is curtailed; in 11nm, frequency must be throttled when adding more cores. The Intel Atom core is much more power-efficient and can continue to scale frequency until 11nm, with additional power headroom. However, Atom starts with a significant performance disadvantage compared to Core2Duo, and hence by 11nm, the Core2Duo and Atom roughly converge on total throughput. In 11nm, the best designs (Atom and Core2Duo) are increasing at a rate of 1.35x per year, which by 11nm is nearly 6.6x below the 1.58x per year curve.

The mobile market segment, seen in Figure 4 exhibits similar trends, but the tighter power constraints result in more severe reductions in clock frequency, and slowing in overall per-year throughput growth. For example, the Core2Duo hits a frequency cap around 32nm, and frequency flatlines until 16nm when it slightly dips. Even the Atom processor power caps at 16nm, after which frequency also dips to maintain the power budget.

An important issue that we see repeatedly throughout the above scenarios is the minimum Vdd constraint is met as we seek to fit designs with many cores into fixed power budgets by reducing voltage and clock frequency. When a design reaches this constraint, additional power reduction can only be achieved through inefficient frequency-scaling – essentially linear reduction in clock frequency offsets additional cores. Practically speaking, designers may prefer to simply



(a) Vdd (b) Frequency

(c) Power (d) Throughput

Figure 4. Results with power constraints across process technologies - Mobile. *Results assume nominal voltage for specified technology and Mobile market segment with a die size of 100 mm^2 and max power of 35 W. Vdd is limited to VddMIN.*

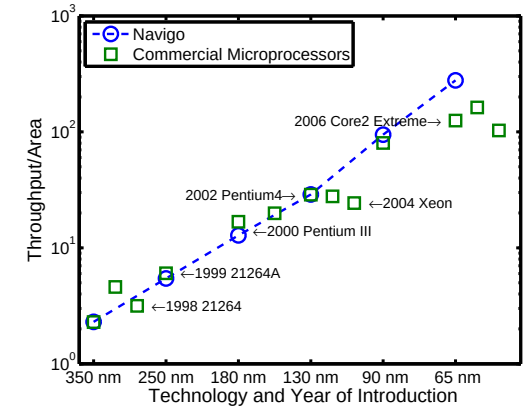
stop scaling the number of cores in a system at this point. In order to understand this effect, we have run additional simulations with the constraint removed. For the Atom processor, minimum Vdd is not a severe issue. For the mobile market segment in the 11nm node, throughput is reduced by 13.4%. However, the minimum voltage constraint reduces the throughput of the Xeon core by 57.6% for the same target. Even without this constraint the Xeon still performs poorly compared to the more power-efficient cores, because running at very low voltage does not provide ideal performance.

4. Validating the Model

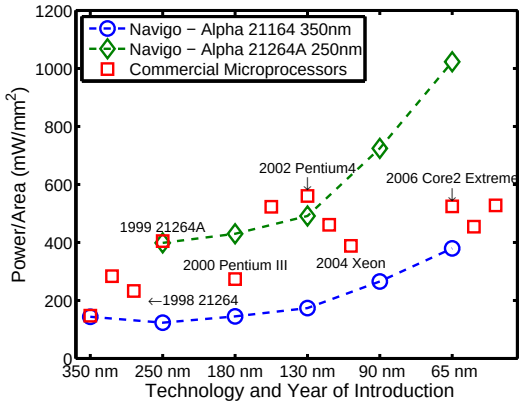
This section presents a back-validation of Navigo for microprocessors built from 1996 to 2007. Because of the predictive nature of the model, it is difficult to validate Navigo's predictions of the power and performance of microprocessors built using future process technologies. Therefore, we validate Navigo based on an initial data-point from 1996 against Microprocessors manufactured over the last 10 years. For validation, we seeded the microarchitecture library with the DEC Alpha 21164 microprocessor, introduced in 1996 and manufactured in 350nm technology. We developed the technology and circuits library based on ITRS data from 1997 to 2007 and circuit simulation results using SPICE models from industry and PTM. For each node, we chose the technology model from the ITRS year closest to the date of introduction. This technique isolates the error in ITRS predictions from the modeling framework. We compare predictions from Navigo with microprocessors manufactured between 1996 and 2007, as plotted in Figure 1. We gathered power consumption data from datasheets and online microprocessor reports. The die size of the microprocessors vary widely; therefore, we compare throughput per unit area and power per unit area.

Figure 5 (a) presents a comparison of throughput per unit area of Navigo predictions and commercially available microprocessors. The x-axis represents both technology node and year of introduction. Predictions from Navigo match the initial core, Alpha 21164 0.5 GHz, revealing an absence of static offset errors in the model. The throughput predicted by Navigo aligns with the results from the benchmarked microprocessors. Generally, Navigo estimates the upper bound of throughput per unit area. To combat increasing power consumption, designers of microprocessors in the 65nm node slowed the scaling of clock frequency and implemented multi-core processors with simpler cores. Navigo overestimates the throughput of multi-core designs because it assumes no cost for communication and thread synchronization.

While Navigo predicts a general trend of increased power density, its accuracy is dependent on the power density of



(a) Throughput



(b) Power

Figure 5. Validation of Navigo using Microprocessors from 1996 to 2007.

the initial microarchitecture in the library. Consequently, Figure 5 (b) plots predictions based on two different cores the lower density Alpha 21164, and the higher power density Alpha 21264A, introduced in 1999 in 250nm technology.

During the period between 1997 and 2005, microarchitects aggressively pursued single-thread performance resulting in several high-throughput and high-power consumption designs. The deeply pipelined Netburst microarchitecture, manufactured in 130nm, had notoriously high power consumption. Subsequently, the industry changed course and introduced more power efficient multi-core designs. The power consumption predicted by Navigo using the 21164 matches the initial core Alpha 21164 in 350nm. The Alpha 21264A represents a higher power density microarchitecture, therefore, predictions using this core match well with Pentium 4 (Netburst) based designs. Because we model unconstrained power consumption, the curve based on the 21264A climes steeply past $600mW/mm^2$, the typical maximum set by the market. To combat this increase in power density, around the 90nm node the industry changed to less power dense multi-core microarchitectures which better match the Alpha 21164 curve.

Our back-validation shows that Navigo predicts throughput well and points out general trends in power consumption. Navigo incorporates a static model of microarchitecture, and thus for a more accurate prediction of power consumption, users should include cores in their libraries which best represent their target core design.

5. Modeling Specialization

Consistent progress towards smaller, faster, and more numerous transistors with each generation of process technology no longer yields the steady growth in computing performance enjoyed throughout the 20th century. The power ceiling forced a "right-hand turn" in single-thread performance and CPU designers have been racing to implement multi-core systems ever since. Unfortunately, Navigo predicts that even for the server market segment, multi-core scaling will only yield a 1.35x/year performance growth trend. In order to get back onto the 1.58x growth trend, designers must maximize the efficiency of transistor (and wire) switching. In other words, designers must minimize the overheads associated with a general-purpose (GP) CPU. One obvious direction is to replace general-purpose computing with dedicated, specialized hardware that offers higher computation per unit area and power, for an increasing fraction of the machine's workload. IBM's CELL processor is one such example. It includes 8 SPEs, which are specialized cores used to speed up SIMD workloads [1]. Another example may be to introduce dedicated hardware specialized to H.264 decoding. In order to understand the potential benefits of specialization, this section introduces a parallel-variant of Amdahl's Law for specialization. Then, by augmenting Navigo with specialization, we project the amount of specialization that will be required in future computing systems to increase system throughput by 1.58x per year.

5.1. Variant of Amdahl's Law for Specialization

Amdahl's Law is commonly used to describe the theoretical limitations of application speedup given constraints on the fraction of the workload that can be sped up.

$$Speedup_{enhanced}(f, S) = \frac{1}{(1-f) + \frac{f}{S}} \quad (2)$$

where f is the fraction of the workload that can be enhanced and S is amount of speedup possible through enhancements. Amdahl's Law has been adapted to model symmetric and asymmetric multicore systems [4], where parallel cores can execute all workloads. With specialized cores, we must make a few assumptions in order to model speedup using Amdahl's Law. First, we assume special-purpose (SP) cores can only run specific parts of an application (f) while general-purpose cores can run the entire workload, albeit with lower efficiency. Second, we optimistically assume that workloads are arbitrarily parallelizable (also previously assumed in Navigo). The

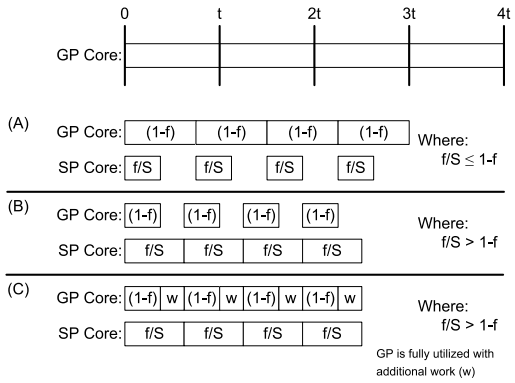


Figure 6. Speeding up an application with specialized cores. *A workload is split to an additional set of resources—the specialized core. The fraction of the application that can be executed on the specialized core is f, with a speedup of S.*

basic framework for calculating speedup possible with specialization is presented in Figure 6. In the absence of specialization, assume a GP core computes 4 units of workload in 4t units of time. By adding a specialized core, a fraction of the workload (f) can be offloaded and completed in f/S units of time. The GP cores only computes $1-f$ of the work, requiring $(1-f)*t$ units of time. If $f/S < 1-f$ (scenario A), then the GP core is the bottleneck and the specialized core idles. However, if $f/S > 1-f$, the SP core becomes the bottleneck as shown in scenario B. However, work ($w = \frac{f}{s} - (1-f)$) can be allocated to the GP core to prevent it from idling (scenario C). Total throughput is calculated to be the original throughput multiplied by the total application speedup. Total speedup is calculated for scenarios A, B, and C as follows:

$$Throughput_{new} = Throughput_{original} * Speedup_{total}$$

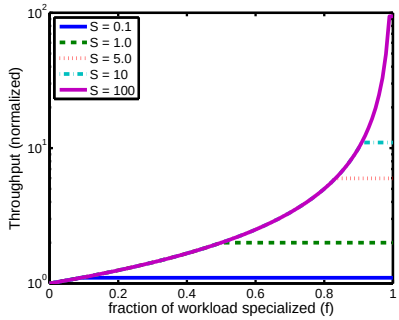
$$Speedup_A = \frac{4t}{4t*(1-f)} \Rightarrow \frac{1}{1-f} \quad \text{if } f/S \leq 1-f$$

$$Speedup_B = \frac{4t}{4t*(f/S)} \Rightarrow \frac{1}{(f/S)} \quad \text{if } f/S > 1-f$$

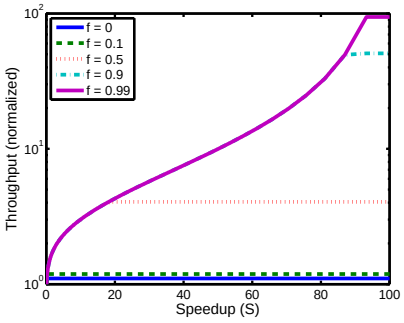
$$Speedup_C = \frac{4t}{4t*(f/S)} + (\frac{f/S-(1-f)}{f/S}) \Rightarrow \frac{1}{(f/S)} + (\frac{f/S-(1-f)}{f/S}) \quad \text{if } f/S > 1-f$$

$$Throughput_{new} = Throughput_{original} * min(\frac{1}{1-f}, \frac{1}{(f/S)} + \frac{f/S-(1-f)}{f/S})$$

Throughput is highest when both f and S are maximized. Figure 7(a) plots throughput enhancements versus f for different S. When $S = 1$, throughput increases with f until



(a) Throughput vs f



(b) Throughput vs S

Figure 7. Understanding the impact of specialization on throughput. *Calculations of throughput with specialization for different speedups (S) and fractions of workload (f). Assumes the general purpose core is fully utilized and resources for an additional specialized core has been provisioned.*

$f = 0.5$ and flattens out with a throughput of 2X because the machine is limited by the SP core (scenario C). As S increases, the throughput flattens out at higher values of f. Similarly, Figure 7(b) shows that throughput flattens out despite increases in S when the machine is limited by the GP core (scenario A). To explore the effects of area and power on this throughput enhancement model, we consider two examples of SP cores—CELL SPE and H.264 decoder.

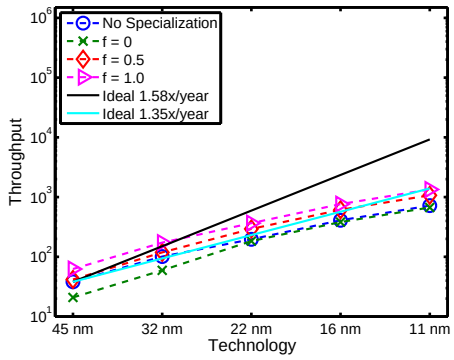
5.2. Examples of Specialized Cores

While specialization offers great potential for throughput enhancements, it is important to carefully account for limitations imposed by the power and area consumed by the specialized cores, as they invariably eat into the overall system power and area allotments normally allocated to GP cores. Adding SP cores reduces the number of GP cores in the system and their higher power densities also impact the voltage and frequency scaling of the GP cores. Furthermore, each SP core's contribution to leakage power is accounted for by Navigo based on transistor counts and technology models. Table 4 lists the two SP cores we investigate. The CELL SPE unit is an example of a programmable SP core designed to speedup media and other streaming computations, which ex-

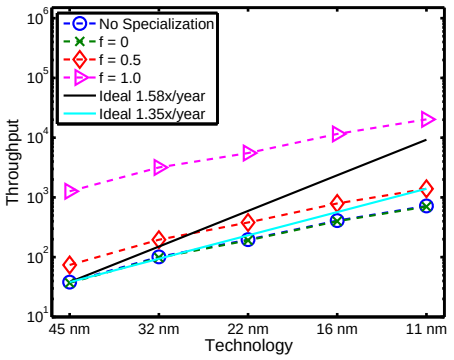
hibit SIMD characteristics. The H.264 decoder an SP core designed to speed up one specific task—in this case, decoding H.264 streams. While the H.264 decoder has much higher speedup per area and power compared to both the SPE and GP core, overall speedup highly depends on the workload fraction that can run on it. In comparison, the CELL SPE offers more modest speedup, but its programmability offers more opportunities to map a larger fraction of the workload onto it.

To understand how specialization can improve overall system performance, we incorporate the example SP cores into Navigo and analyze throughput trends versus technology nodes for the Mobile 35W market segment. Figure 8 presents throughput trends across technology nodes when eight SPEs and a single H.264 decoder are added per Core2 GP core, respectively, for several values of f. In order to account for the impact of maintaining constant overall area, the GP core's IPC scales down linearly with area reduction due to addition of SP cores. The trend plots are normalized to a chip in the 45nm technology using Core2Duo-based GP cores to be consistent with all other analysis in the paper thus far. The plots reveal several expected outcomes. First, higher f's consistently improve throughput since larger fractions of the workload can be sped up. Second, higher speedup (by utilizing more SPE cores or a H.264 core) further improves maximum achievable throughput. Third, as technology continues to scale, specialization will be critical to maintain a 1.58x/year growth in system performance. Lastly, given a fixed SP core, f must increase with each generation of technology to maintain performance growth.

Another way understand the above analysis is to determine how much f and fraction of area for specialization (A_{SP}) designers must target for each process generation to maintain the 1.58x/year performance growth. We again consider the Mobile 35W market segment and assume the total chip area remains constant across each technology generation. Figure 9 overlays the regions of f versus A_{SP} that can maintain 1.58x/year performance growth using SPEs and H.264 SP cores. To understand this plot, let us focus on the region outline for the 45nm technology node using SPEs in Figure 9(a). Since the analysis is normalized to the 45nm technology without specialization, as A_{SP} grows, the fraction of the workload, f, offloaded to the SP core must grow proportionally. Otherwise, the degraded GP core alone would not be able to achieve the original throughput. At the 32nm node, specialization is needed to maintain the 1.58x/year throughput increase, but a small amount of specialization is sufficient as long as there is work that can be offloaded to the SP core. Continued technology scaling requires larger amounts of A_{SP} and f to maintain throughput trends. At the 11nm and 16nm nodes, the speedup of SPEs is inadequate. In contrast, the much larger speedup possible with H.264 SP cores leads to much larger regions across the technology generations as shown in Figure 9(b). Throughput growth trends can be maintained even at the 11nm node, provided a large enough fraction of the workload can be offloaded to the SP core ($f > 0.9$). In summary, future system designers can leverage SP cores to maintain throughput growth trends, but the SP cores must be carefully chosen to provide sufficient high speed up and be able to execute a significant fraction of the workload. While this analysis only considers a single type of SP core, a combination of multiple heterogeneous SP cores ought to be explored.



(a) CELL SPE x8



(b) H.264 Decoder

Figure 8. Specialization across process technologies with real SP cores. *Total throughput for different values of f assuming the area and speedup of one example SP core per GP core. Mobile 35W market segment.*

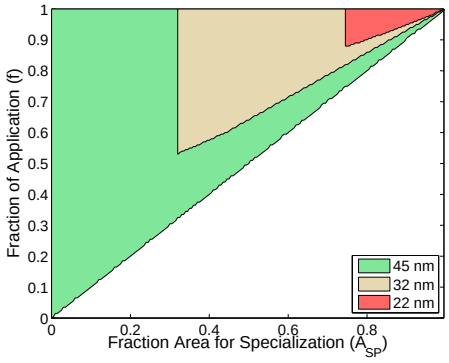
6. Conclusion and Future Work

Growth in the computational throughput of future devices will be limited by power density and strict market segment-oriented power constraints. In this work we introduce a model designed to fit in the space between the cycle accurate models used by industry design teams to validate their architectures, and the spreadsheets currently used by industry architects to plan the next generation of processors five to ten years from tapeout. Our results show that under power constraints total throughput growth is slowing. We show that by allocating an increasing amount of area to specialization for each process technology generation, designers could make up for the gap in throughput and maintain growth.

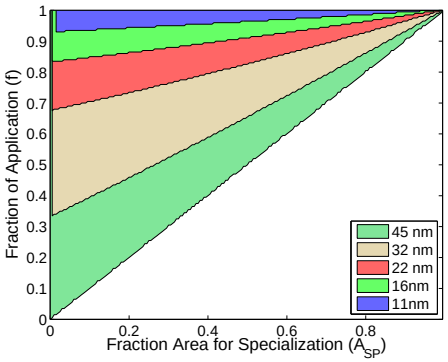
Initially, Navigo was constructed with a set of assumptions — that workloads are completely parellelizable and that

Core Type	Application Type	Area (mm^2)	Freq.	Power	Speedup (S)	S/Area	S/Area/W
CELL SPE [1, 9]	Programmable SIMD	11.08	4 GHz	2 W	0.446	0.040	0.020
H.264 [6, 5]	Specialized H.264	3.42	30 MHz (30 fr/sec)	91mW	33.6	9.82	107.91
Core2Duo	General Purpose	100	3 GHz	65 W	1	0.01	$154*10^{-6}$

Table 4. Specialized Cores. *Example SP cores used in the model. All measurements were scaled to 65nm technology and speedup was calculated by comparing published performance results to the performance on a general purpose CPU. The Core2 is included to show the relative area and performance cost of including another GP core instead of an SP core. Power and speedup for CELL SPE running Linpack.*



(a) CELL SPE



(b) H.264 Decoder

Figure 9. Configurations that can achieve 1.58x/year throughput. *Model two different accelerator structures the programmable CELL SPE and an H.264 accelerator. Core2Duo-based GP cores and the Mobile 35W market assumed.*

the on-chip network and thread synchronization cost nothing. These modeling decisions ensured that multi-core designs were not overly penalized and that the results represented an upper-bound to performance and power consumption. Some architects doing early analysis and exploration would prefer a less idealized notion of cost and performance. Consequently, Navigo could be enhanced to model memory access and network synchronization overhead and allow a distinction between serial and parallel workloads.

While we have populated Navigo's scaling libraries with an initial data set, we anticipate that the methodology will be applied by researchers with more detailed technology, circuit, and architectural information. We believe that a new architectural paradigm focused on specialized resources will be needed to reclaim performance growth, and this work allows researchers to explore the amount of specialization required to achieve target performance growth for future technology nodes.

References

[1] Brian Flachs et al. The microarchitecture of the synergistic processor for a cell processor. *IEEE Journal of Solid-State Circuits*, 41(1):63–70, January 2006.

[2] G. Gerosa et al. A sub-1w to 2w low-power ia processor for mobile internet devices and ultra-mobile pcs in 45nm hi-k metal gate cmos. In *IEEE International Solid-State Circuits Conference (ISSCC)*, February 2008.

[3] J. Hennessy and D. Patterson. *Computer Architecture: A Quantitative Approach*. Morgan Kaufmann Publishers, Boston, Massachusetts, 2003.

[4] M. Hill and M. Marty. Amdahl's Law in the Multicore Era. *IEEE Computer*, July 2008.

[5] V. Iverson, J. McVeigh, and B. Reese. Real-time h.264/avc codec on intel architectures. In *International Conference on Image Processing (ICIP)*, October 2004.

[6] H.-Y. Kang, K.-A. Jeong, J.-Y. Bae, Y.-S. Lee, and S.-H. Lee. Mpeg4 avc/h.264 decoder with scalable bus architecture. In *IEEE International Symposium on Circuits and Systems (ISCAS)*, February 2004.

[7] Y. Lin, H. Lee, M. Woh, Y. Harel, S. Mahlke, and T. Mudge. Soda: A low-power architecture for software radio. In *International Symposium on Computer Architecture (ISCA)*, June 2006.

[8] A. Mahesri, D. Johnson, N. Crago, and S. Patel. Tradeoffs in designing accelerator architectures for visual computing. In *International Symposium on Microarchitecture (MICRO)*, November 2008.

[9] O. Takahashi et al. Migration of cell broadband engine from 65nm soi to 45nm soi. In *IEEE International Solid-State Circuits Conference (ISSCC)*, February 2008.

[10] PTM. Predictive Technology Model. `http://www.eas.asu.edu/~ptm/`.

[11] Semiconductor Industry Association. International Technology Roadmap for Semiconductors (ITRS). `http://www.itrs.net`.

[12] Standard Performance Evaluation Corporation. SPEC Benchmark Suite. `http://www.spec.org`.

[13] Stefan Rusu et al. A 65-nm dual-core multithreaded xeon processor with 16-mb l3 cache. *IEEE Journal of Solid-State Circuits*, 42(1):17–25, January 2007.

[14] Tom's Hardware. Atom Benchmarked: 4W of Performance : Intel Atom 230 At 1.60 GHz with Hyper-Threading. `http:tomshardware.com/reviews/Intel-Atom-Efficient,1981.html`.

[15] C. Wilkerson, H. Gao, A. Alameldeen, and Z. Chishti. Trading off cache capacity for reliability to enable low voltage operation. In *International Symposium on Computer Architecture (ISCA)*, June 2008.

[16] W. Zhao and Y. Cao. New generation of predictive technology model for sub-45nm early design exploration. *IEEE Transactions on Electron Devices*, 53(11):2816–2823, November 2006.