

FROM ADAPTIVE TO SELF-TUNED SYSTEMS

Sudhakar Yalamanchili, Subramanian Ramaswamy and Gregory Diamos

Computer Architecture and Systems Laboratory (CASL)
School of Electrical and Computer Engineering
Georgia Institute of Technology
Atlanta, GA 30332

Email: sudha@ece.gatech.edu, ramaswamy@gatech.edu, gtg250v@mail.gatech.edu

1. INTRODUCTION

The relentless progress of Moore's Law has periodically inspired major hardware and software innovations at specific points in time to keep performance growth on pace with transistor density. The industry has reached another such point as it encounters major intellectual and engineering challenges in the form of i) increasing energy demands, ii) increasing processor-memory performance gap, iii) limits to instruction level parallelism (ILP), iv) an effective end to frequency scaling, and v) rising non-recurring (NRE) engineering costs for chip design. As a consequence, when we consider how to employ the transistors that will be supplied at future technology nodes to sustain performance growth, one can see some inevitable trends including i) performance scaling via replication of cores and consequent rise in the importance of the on-chip and chip-to-chip interconnection networks, ii) the use of custom accelerators due to the fact that these devices have small footprints and dramatically increased efficiency for certain operations, and iii) innovations in memory hierarchies due to the large percentage of transistors devoted to on-chip memories. The preceding collectively inspire the development of *heterogeneous many-core platforms* - large scale, heterogeneous systems comprised of homogeneous general purpose cores intermingled with customized heterogeneous cores and interconnected to diverse memory hierarchies. Such systems will be prevalent in both on-chip as well as in rack scale and multi-rack scale systems.

The principal goal of future designers is to architect these systems in a manner that sustains the performance impact of Moore's Law while inhibiting the growth of energy demands. Two major classes of challenges that have been well documented in the literature are the i) programming models and accompanying compilation and software development platforms, and ii) technology challenges, principally process, voltage, and temperature (PVT) variations and escalating energy requirements. The impact of the latter is the subject of this paper. Sustaining performance growth will require abandoning principles of the past which were effective but inefficient, and moving to approaches that emphasize robustness and efficiency. As we move into deep sub-micron region of design, traditional design flows based on worst case assumptions are beset with large design margins that are economically impractical. As a result, we are witnessing the end of an era wherein design is based on worst-case analysis and the movement to an era whose main design abstractions rely on statistical analysis, probabilistic design, and (as advocated here) continuous software-controlled calibration and tuning. PVT variations are not the exclusive problem of device, circuit or even micro-architecture technologists. Rather variation tolerance is the ability of HW/SW systems to deliver performance to end applications - the ultimate consumer. As such, some aspects of variation tolerance can be addressed within the micro-architecture and others within software-based methods that are co-designed with the micro-architecture techniques. From this systems perspective, die yield is defined by the distribution of the performance that can be achieved with this co-designed, device, architecture and software system, where yield of the micro-architecture is based on the ability to meet performance goals and not solely on the presence of defect-free substructures. This observation implicitly recognizes that many micro-architecture structures designed for the worst-case design era have inherent inefficiencies that can be eliminated with more flexible, adaptive structures. This paper presents some early examples of pursuing this approach in the context of the cache hierarchy and on-chip networks.

2. TECHNOLOGY IMPACT

The driving technology impact issue for future computing systems is one of sustaining the historical performance and cost benefits of Moore's Law as industry moves into the deep sub-micron region. The continued benefits are potentially jeopardized by two phenomena: physical variability and energy demands. Several studies in the past few years have documented the challenges of the manufacturing process to fabricate devices within anticipated design tolerances leading to significant variations in transistor device characteristics within a die (WID), across dies (D2D), and between wafers (W2W) (e.g., [1]). The cause of variations can be traced to random dopant fluctuations at small device geometries as well as parametric variations due to the manufacturing process caused by sub-wavelength lithography, imprecision in chemical polishing, uneven exposures, etc. Furthermore, during operation, supply voltage fluctuations due uneven power distribution, thermal gradients and device wear-out can dramatically affect operation while the feedback loop between temperature and leakage currents can lead to thermal runaway if not corrected. The consequences of such phenomena are an increase in die level variance of key parameters and a severe drop in defect free yield. As a result, modern design regimes that are based on worst-case assumptions can incur prohibitive costs in performance and yield in the presence of Process, Voltage and Temperature (PVT) variations motivating the need for architectures that are variation tolerant (VT).

Traditional system designs have generally accepted the convention that device parametric variations should be dealt with by adjusting and refining the process used to manufacture them, rather than by making changes to the micro-architecture. As long as parametric variations are relatively small when compared to their corresponding device attributes, conventional micro-architecture designs have proven that they can be safely ignored without compromising yield. However, as manufacturing processes advance to future technology nodes, parametric variations are expected to increase to such an extent that they will begin to significantly alter device properties and potentially compromise yield at high performance points [2–6]. The increase in variations can be attributed to several sources, one of which is the relatively constant wavelength of light used to etch transistors. For a 65 nm process, the wavelength of light used to etch the smallest feature, the gate, is 193 nm. This problem will be further exacerbated at future technology nodes as this wavelength decreases at a lower rate than the reduction in minimum feature size. A second source of variation is the decreasing concentration of dopant atoms in transistor channels. Borkar [3] has shown that the number of channel dopant atoms in a 65 nm transistor is around one hundred. When dealing with such a small number of atoms, seemingly insignificant fluctuations in the number of dopant atoms can significantly affect the electrical characteristics of the transistor. To compound the problem even further, linear increases in parametric variations have been shown to produce nonlinear increases in timing variability and design margins in general. Using a specific case study of the Intel Core Duo processor, Annavaram *et al.* [2] have shown that as parametric variations increase from 0% to 50%, timing margins begin increasingly slowly but quickly become faster. Though this study did not include variations due to interconnect delay which have been predicted to be comparable to transistor variations at technology nodes below 90 nm.

Parametric manufacturing variations are not the only source of variations that modern designs have to consider. Environmental conditions such as fluctuations in power supply voltage, power supply noise, changing temperature gradients, and inductive and capacitive crosstalk introduce variations that occur during normal operation. Worse, long term effects can cause a device's performance to degrade over time. More recent observations point to the phenomenon of hard failures due to device "wear-out". Wear-out in silicon can be attributed to three phenomenon - oxide breakdown [7], electromigration [8,9] and hot-carrier interaction [10,11]. Oxide breakdown is the creation of a conducting path in the oxide layer underlying the gate, and is generally caused by high electric field gradients on the oxide and is becoming more prevalent due the reduced thickness of the oxide layers. Electromigration is the movement of metal atoms (interconnect) caused by the force exerted on the lattice atoms of the metal by the conducting electrons leading to open circuits. Finally, hot-carriers are high speed electrons that penetrate the gate oxide layer, which over time form conducting paths through the oxide. All these effects are worsened as design margins shrink. Thus one can foresee the need for systems to be tolerant of hard faults.

These increases in variations have functional consequences (reliability), performance consequences (speed, energy efficiency), and economic consequences (e.g., yield, verification and testing costs). For example, to keep yields high, timing margins must be wide enough to account for variations in delay and supply voltage margins must be wide enough to ensure that all transistors are correctly biased. It is no longer efficient to devote active transistors to speculative operations be they for storage, computation or communication. As long as frequency was a proxy for performance, devoting transistors to speculation sustained performance growth via faster cores across technology nodes. Now that frequency has been replaced by area as a proxy for performance, effective utilization from an

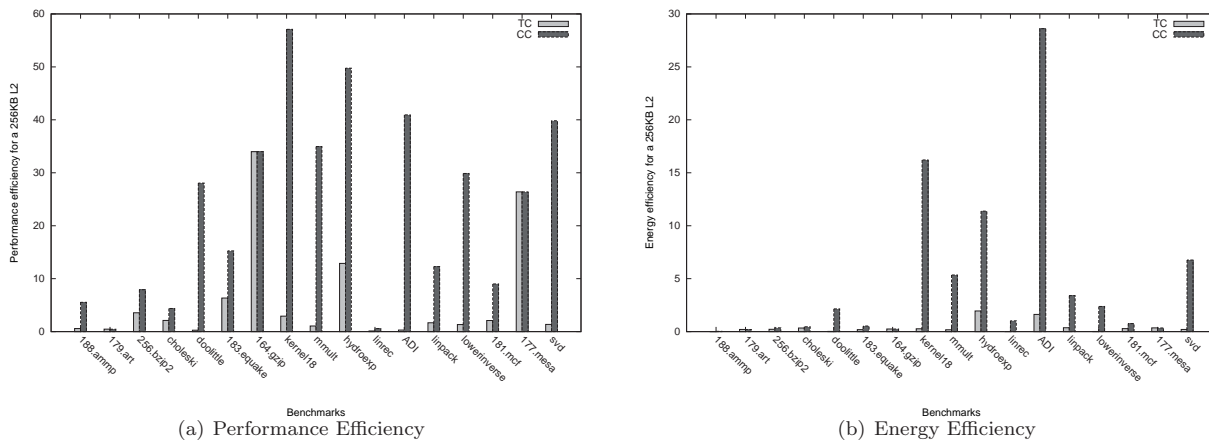


Fig. 1. Comparison Between a 256KB Traditional Cache (TC) and Customized Cache (CC)

errors such as read access failures [24]. Supply voltage scaling to reduce power consumption increases the failure rate of SRAM cells and produces faulty lines in the cache. Such failures have historically been addressed via redundancy management techniques and effectively reduce the die yield. The ability to efficiently harvest fault-free cells can be used to significantly improve effective die yield.

3.1. Dynamic Scaling of Caches: Sizing and Shaping

Cache sizing refers to scaling the cache size by turning off cache sets to (ideally) match program memory footprint, while cache shaping refers to definition of the manner in which memory shares the cache, i.e., the cache placement function. Software controlled placement differs from prior approaches that seek to pro-actively turn off parts of the cache in that the subset of active cache resources represent fully functional caches - parts that are turned “off” will not be referenced, nor is the placement uniform, i.e., variable sized memory regions may be mapped to each cache set. This approach is advocated for the large L2 and L3 caches which are the major sources of inefficiency due to their size and relatively low activity level. We view our approach as a natural extension to the prior techniques for intelligently turning off portions of the cache for short periods of time, where now new placement functions can be invoked under software control for program regions as a consequence of program analysis or run-time measurement. This philosophy extends naturally to fault tolerance, where larger numbers of defects can be tolerated by remapping memory to fault free cache sets, for example as shown in Figure 2. A major advantage of this approach is that it can be combined with existing hardware and software cache optimizations while realizing many instances of fixed indexing optimizations found in literature. For example, compiler optimizations such as loop blocking or tiling, which were limited to a fixed cache design can potentially become more powerful by employing a cache that is concurrently sized and shaped to maximize performance.

3.2. Sizing and Shaping - Sample Results

Our results indicate that such active management approaches improve cache efficiencies significantly with relatively modest overheads. For example, using customized placement for shaping conflict sets to improve the performance yield for caches for a given fault distribution, increased the total number of usable dies by as much as 600 for a set of 1000 dies [25] (Figure 2.) The shaping heuristics improved yield by using profile information to remap memory to a smaller cache, but with reduced conflict misses.

Figures 1(a) and 1(b) show the improvement in performance and energy efficiencies for a customized cache that employs sizing and shaping. In the figure shown, sizing and shaping strategies suited to scientific applications were applied statically. Scientific applications are well suited to static sizing and shaping because of the vast knowledge base that exists for these domains. Using sizing and shaping, efficiencies improved across the board with efficiencies increasing by almost an order of magnitude for many benchmarks exhibiting the efficacy of sizing and shaping for customized cache management. These improvements in efficiency are evident in drops in the energy-delay product

