Toward a Science of Power Management

Krishna Kant, *Intel Corp.*

A formal understanding of energy and computation tradeoffs will lead to significantly more energy-efficient hardware and software designs.

Computers originally evolved with the goal of automating repetitive calculations, and ever since have focused on doing the job faster and cheaper. Consequently, the dominant resources in most computational settings have been processing time and memory occupancy.

Beginning in the 1950s, researchers have exploited computational complexity theory to achieve remarkable successes in devising faster algorithms, proving bounds on computational speed, and defining classes of problems of equivalent difficulty. Today this huge, vibrant field offers benefits that extend far beyond theoretical computer science into the design of systems that directly solve socially relevant problems.

With the explosive growth of Internet-enabled gadgets of all types, IT's energy consumption and sustainability impact are expected to continue climbing well into the future. Although the problem is well recognized and aggressive efforts to address it are under way in both industry and academia, much of the effort tends to be empirically driven.

Just as the formal notions of computational complexity have immensely benefited hardware and software design, a formal treatment of energy, power, and thermal issues should also lead to significant advances in how such systems are designed in the future. The vision is of a well-developed body of knowledge that can guide various power-performance tradeoffs in the design, tuning, and operation of IT devices ranging from nanowatt-level embedded devices to large server farms consuming tens of megawatts of power.

Although power and energy are often used interchangeably, there are important distinctions. Transactional applications are better described in terms of throughput and average power, whereas completion time and energy consumed are more meaningful for nontransactional applications. Power consumption can often be increased over short periods to accelerate computation or accumulate more data and thereby minimize overall energy consumption.

ENERGY VERSUS COMPUTATION

Our current understanding of energy issues is rudimentary. For example, is there a notion of energy (or power) complexity of an algorithm distinct from computational complexity? Certainly a program's energy consumption strongly depends on the number of instructions executed and the number of accesses to the memory hierarchy—these are the same factors that determine a program's completion time. Also, an instruction that takes fewer clock cycles to execute generally also consumes less energy.

Yet the correspondence between completion time and energy consumption is not one-to-one. For example, the order of certain operations and bit representation of data can alter the energy consumed without necessarily changing the completion time.

At a coarser level of detail, however, computation and power/energy complexity may differ because of various power-management actions available in modern computing hardware. For example, many devices provide the capability to transition to one or more lower-power modes when idle. If the transition latencies into and out of lower power modes are negligible, energy consumption can be lowered simply by exploiting these states.

Unfortunately, the transition latencies are rarely negligible and thus the use of low-power modes impedes performance. To minimize this impact, it is necessary to alter the workload so that many small idle periods can

0018-9162/09/$26.00 © 2009 IEEE

be aggregated into fewer large ones. This form of workload batching is a key technique for minimizing energy consumption and it extends from individual hardware components all the way up to entire computing clusters. Further, effective batching often requires coordination among multiple entities

Another form of power management provided by many devices is operation at different speeds, with lower speeds allowing for lower operating voltages as well. This can affect both the idle and active (computation-related) power consumption of a device, and it too involves transition overheads between states.

Power-management actions may affect performance in complex ways because the overall computation rate is a net result of the speed and coordination of multiple elements within the system. For example, doubling the CPU speed may do little to increase the computation rate if the memory transactions do not move any faster. This also indicates that models for the study of energy-computation tradeoffs would need to address more than just the CPU.

Clearly, the average power consumption and computation rates are intricately tied together, making it difficult to speak of power complexity in isolation. Especially in the context of reducing energy consumption via power-management techniques, meaningful models must consider the impact on computational performance. Thus, the models we seek to design and parameterize should relate to power consumption and computation rate (or energy consumption and completion time) simultaneously.

This raises the all-important issue of metrics that are appropriate to characterize power-performance interplay.

Single-objective metrics such as performance per watt or power-delay product can be useful but often do not explicitly relate to the desired performance level. For example, maximum performance per watt may be achieved by setting the power-management knobs such that performance is 20 percent lower than without any power management, but such a large degradation may be unacceptable.

Other, more direct metrics to consider are minimum energy consumption subject to tolerable impact on performance, or its dual counterpart: maximum performance subject to given energy consumption limits.

The average power consumption and computation rates are intricately tied together, making it difficult to speak of power complexity in isolation.

The first metric relates to the performance-centric view that has been the mainstay of computing until recently: maximizing energy efficiency of computations subject to a rather small impact on performance. The second connotes the more general energy-limited view: the maximum computation possible subject to a given cap on power or energy.

This cap may result from power circuits' limited capacity, restrictions in energy harvesting or production, or limitations in wasted energy-removal (or heat-dissipation) capabilities. In general, the energy or power cap could be dynamic, thereby leading to a dynamic optimization problem.

SCIENCE OF POWER MANAGEMENT

Tackling these problems requires an abstract machine model of both computation and power management that can be used to study numerous types of power-performance tradeoffs, derive feasibility results and bounds, and parameterize energy-management algorithms.

What is needed is a hierarchy of models with varying abstraction levels. In addition, the models should reflect the physical and logical structure of computing systems in terms of multiple subsystems operating at multiple granularities of control. Collectively, such models will form the beginnings of the science of power management (SciPM).

A comprehensive SciPM should let researchers address many other facets of energy consumption as well. As the need for reducing energy consumption grows, so does the number of control knobs. Hardware designers are developing new mechanisms for reducing energy consumption, most of which have performance implications. For example, deeper sleep states and more speed states may be added to various platform components.

The crucial question is whether formal techniques can indicate how many states are needed and what their essential characteristics should be. Further, since a cross-product of power states of all components quickly becomes unwieldy, a formal model is needed to establish relationships between various states not only in terms of their number and characteristics but also in terms of their simultaneous usage.

Power management relates to more than just coordinated power state management. As semiconductor technology begins to reach limits of reliable operation, computation models will need to change to account for this unreliability. SciPM thus must address three attributes together: performance, energy, and reliability.

It may be necessary to invent new hardware architectures that slow down error propagation, which would raise a new set of issues in modeling this three-way tradeoff. Can we develop a science to start examining such tradeoffs before the products need to deal with it?

Another benefit of a formal approach is the exploration of more energy-efficient, semantically equivalent software transformations—for example, automated reordering of operations to allow optimum use of available power-management functions.

Finally, SciPM should ideally help quantify the capabilities and limitations of energy-efficiency mechanisms at lower levels of detail, including hardware architecture, circuit design, semiconductor processes, and materials physics.

SCIPM WORKSHOP

With these goals in mind, the National Science Foundation sponsored a SciPM workshop in Arlington, Virginia, in April 2009 (http://scipm.cs.vt.edu). The workshop generated tremendous interest among researchers across the US facing various data-center-related issues. A significant number of theoretical computer scientists also participated.

The 85 attendees were divided into five groups to focus the deliberations.

The *physicals* group examined issues related to physical infrastructure, which often represents a substantial part of data-center cost and could easily consume more than 50 percent of the energy. These include cooling from room level down to server fans, electrical conversion and distribution from substation to board-level voltage regulators, displays, and so on.

The *hardware and architecture* group was tasked with addressing issues ranging from new materials and processes to new architectures and power-management knobs.

The *software and middleware* group was concerned not only with sophisticated multilevel and multidomain energy management but also with energy-efficient software design.

The *storage* group considered the energy impact of the increasingly data-intensive nature of computing and various nonvolatile storage technologies currently under development.

Finally, the *networking* group was asked to explore network-related energy-management issues ranging from platform-level interconnects to data-center fabrics to the Internet routing infrastructure.

Each group was "seeded" with a few theoretical computer scientists to blend domain knowledge with abstractions.

The SciPM 2009 deliberations supported a more scientific investigation of power/thermal issues and development of good metrics and models. At least two groups expressed the need for a big O notation for power. Several groups emphasized the importance of multidisciplinary education so that future computer scientists are better prepared to deal with "physicals"-type energy issues in particular and, more broadly, the numerous sustainability challenges inherent in IT equipment and infrastructure design.

Finally, workshop participants recognized the need to tap into the enormous potential of IT to tackle the "remaining 98 percent of the problem"—that is, use IT solutions to significantly reduce the 98 percent of CO_2 emissions not attributed to IT itself (www.cra.org/ccc/docs/init/itandenergy.pdf)

A formal understanding of energy and computation tradeoffs will open up new areas of investigation and lead to significantly more energy-efficient hardware and software designs while preserving the continued improvement in performance that the emerging applications demand. A formal approach will hopefully lead to energy-management solutions that are not only effective but also easy to implement and validate.

Traditionally, energy consumption and supply sides have been fairly well isolated. However, with a greater role played by variable energy sources such as renewable and harvested energy, it is imperative to design systems that are not just energy efficient but also *energy adaptive*—capable of altering their behavior depending upon the energy availability. A formal study of performance-energy tradeoffs will allow us to better design future energy-adaptive systems.

It has been observed that the energy required to create an IT gadget can often exceed its energy consumption during its useful life. Thus from a sustainability perspective, the grand challenge before us is to develop a science that can address energy management in the entire life-cycle context including manufacturing, distribution, operation, and disposal.

***Krishna Kant** is with Future Technologies Research at Intel Corp. and is currently on leave of absence serving as a program director in the Computing Systems Cluster of the National Science Foundation's CISE/CNS division. Contact him at krishna.kant@intel.com or kkant@nsf.gov.*

The ideas discussed in this article stem from discussions with numerous NSF, academic, and industry colleagues.

Editor: Kirk W. Cameron, Dept. of Computer Science, Virginia Tech; cameron@cs.vt.edu