New Frontiers in Energy-Efficient Computing

Vladimir Getov, University of Westminster
Adolfy Hoisie, Pacific Northwest National Laboratory
Pradip Bose, IBM T.J. Watson Research Center

Energy-efficient computing remains a critical challenge across the wide range of future data-processing engines—from ultra-low-power embedded systems to servers, mainframes, and supercomputers.

The microprocessor and computer systems design community encountered the so-called “power wall” in the early years of the 21st century. Researchers anticipated this challenging barrier a decade earlier, but general awareness came only when the race to achieve gigahertz clock frequency hastened sight of that wall. Since then, we have seen a slowdown in clock-frequency growth and the emergence of multicore processors that emphasize throughput increase over single-thread performance boosts.

However, other “walls,” such as the “memory bandwidth wall,” have arisen even as the multicore paradigm has somewhat eased the stress on power density. At the same time, the need to keep pushing single-thread performance (albeit at less-than-historical rates) means that the power wall remains a daunting obstacle to server performance growth. In the embedded systems domain, the escalating phenomenon of sensor-rich big data and the need for real-time edge computing at affordable energy costs have fueled the need for ultra-high energy efficiency. The Internet of Things (IoT) revolution will further push the intelligent sense-compute edge devices toward ultra-low-power operation, making them dependent on energy harvesting from environmental sources.

In short, energy-efficient computing remains a critical challenge across the
wide range of future data-processing engines—from ultra-low-power embedded systems to servers, mainframes, and supercomputers. In addition, the advent of cloud and mobile computing as well as the explosion of IoT technologies have created new research challenges in the already complex, multidimensional space of modern and future computer systems.

These new research challenges led to the establishment of the IEEE Rebooting Computing Initiative, which specifically addresses novel low-power solutions and technologies as one of the main areas of concern.1 With this in mind, we thought it timely to survey the state of the art of energy-efficient computing. Prior magazine issues on this topic2–4 are over a decade old, when the power wall first loomed. Given the ever-changing dynamics of computer systems, we wanted to explore recent research findings and potential developments in light of the paradigm shifts noted above.

EXASCALE ENERGY CHALLENGES

Overcoming the power wall will require major advances in energy-efficient computing. In particular, in the race to reach exascale computing speed and process exabytes of big data, future extreme-scale computer systems must significantly increase their energy efficiency.5 When such systems become operational, which is projected to be after 2020, power and energy consumption will continue to be primary concerns. Achieving viable performance will require revolutionary methods as well as stronger integration among hardware, system software, and applications. Equally important will be the ability to carry out fine-grained spatial and temporal measurements and control, which will facilitate energy-efficient computing across all layers of current and future computer systems.

Unfortunately, existing energy-efficient approaches rely heavily on low-power hardware, which alone will not address emerging exascale energy challenges. Hardware must enable mechanisms to dynamically optimize energy consumption for various workloads and to reduce data motion, which uses considerable power. Also, high-fidelity measurement techniques, typically ignored in datacenter-level power management, are important for scalable and energy-efficient interplay among different application, system software, and hardware layers.

IN THIS ISSUE

In preparing this special issue, our goal was to highlight ongoing research efforts in four main areas:

- power-aware applications, system software, and architectures that significantly increase energy efficiency;
- analysis, modeling, simulation, and optimization techniques for balancing the interplay among performance, power, and reliability;
- in-band and out-of-band infrastructures for high-resolution energy management throughout the hardware/software stack; and
- low-power-consumption design methodologies for novel systems using disruptive technologies.

In response to the initial call for papers, we received 43 extended abstracts expressing author interest to submit articles. Following this first step, 17 manuscripts were formally submitted for review. With the help of a large number of expert reviewers, we selected the five papers that we felt best covered the topic areas and provided a representative sampling of the energy-efficiency challenges and corresponding solution strategies across different computer systems. As such, we had to turn down several excellent and interesting manuscripts.

In “Using Performance-Power Modeling to Improve Energy Efficiency of HPC Applications,” Xingfu Wu, Valerie Taylor, Jeanine Cook, and Philip Mucci describe a modeling framework that incorporates application-specific measurements with data gained from hardware counters to predict performance and power consumption. These models identify the most significant counters for collecting this type of data, leading to substantially improved execution time and energy savings in large-scale systems. Indeed, successful large-scale parallel computing depends increasingly on both performance and energy optimization. Because of the complexity inherent in application software and system architectures, effectively identifying the most promising software features as optimization targets is critical but can be very difficult.

To address this challenge, the authors devised a novel modeling methodology for capturing, predicting, and optimizing application performance and power consumption on high-performance computing (HPC) resources. The identification of key metrics that contribute to performance or power problems is based on formal principles using Spearman correlation and principal component analysis. The methodology gives developers a way...
to effectively adapt and modify their applications to maximize the utilization of particular HPC architectures. Unfortunately, after the relevant metrics have been identified, the process currently appears to be quite labor-intensive because the user must still determine how the designated counters affect the applications and then implement the required code changes, data structure, and layout transformations to remove the bottlenecks.

In “Power, Reliability, and Performance: One System to Rule Them All,” Bilge Acun, Akhil Langer, Esteban Meneses, Harshitha Menon, Osman Sarood, Ehsan Totoni, and Laxmikant V. Kalé propose a comprehensive adaptive system and programming abstraction that enables the user to dynamically optimize applications for performance, power, and reliability. Based on the Charm++ parallel programming framework, the system design integrates several modules necessary for runtime management and optimization of a supercomputing framework. It also includes dynamic solutions for adaptive job scheduling, load balancing, power management, thermal management, and fault tolerance.

The article describes all the diverse tasks required of a power-aware resource manager for HPC systems and the programming framework that would make such a system feasible. Unlike other existing systems that focus on job-scheduling techniques primarily with the objective of load leveling and maximizing throughput, the authors’ approach also makes it possible to optimize performance under fixed power and thermal constraints. The tradeoffs between the mean time between failures and performance for different temperature constraints are clearly illustrated.

In “Standardizing Power Monitoring and Control at Exascale,” Ryan E. Grant, Michael Levenhagen, Stephen L. Olivier, David DeBonis, Kevin T. Pedretti, and James H. Laros III present the Power API monitoring framework, which provides a standard view on how to collect power and energy data at scale in HPC systems. The plethora of devices and techniques in this area highlight the growing need to standardize the power and energy monitoring process. Indeed, many vendors have developed various packages and infrastructures for the instrumentation and management of consumed energy.

However, the existing solutions are far from being portable, and porting application code from one system to another usually requires implementing from scratch the energy analysis and tuning procedures for the new system. The Power API specification aims to provide this missing portability and can be used by tools to collect, display, and gain insight into the varying power requirements of different computer workloads and system architectures. To showcase Power API’s capabilities, the authors provide an example that centers on the application code’s workload characteristics.

In “H-EARtH: Heterogeneous Multicore Platform Energy Management,” Efraim Rotem, Uri C. Weiser, Avi Mendelson, Ran Ginosar, Eliezer Weismann, and Yoni Aizik describe how they achieve higher energy efficiency for different workloads running on different multicore processor architectures through the use of advanced dynamic frequency and voltage scaling (DVFS). Their low-overhead energy-management platform for modern processors with either homogeneous or heterogeneous cores relies on the proposed H-EARtH algorithm, which is implemented in the processor firmware.

Many vendors have developed various packages and infrastructures for the instrumentation and management of consumed energy.

However, the existing solutions are far from being portable, and porting application code from one system to another usually requires implementing from scratch the energy analysis and tuning procedures for the new system. The Power API specification aims to provide this missing portability and can be used by tools to collect, display, and gain insight into the varying power requirements of different computer workloads and system architectures. To showcase Power API’s capabilities, the authors provide an example that centers on the application code’s workload characteristics.

In “H-EARtH: Heterogeneous Multicore Platform Energy Management,” Efraim Rotem, Uri C. Weiser, Avi

The algorithm is based on an analytical model and extracts factors at runtime to determine the number of cores and frequency that lead to the minimum energy consumption.

The authors provide a detailed and thorough evaluation of this algorithm on three systems—a symmetric-core system with a homogeneous microarchitecture, and two asymmetric-core systems (big and small) with a heterogeneous microarchitecture—using experimental data from 37 benchmark programs from several suites. The results show reduced energy consumption by up to 21 percent in general cases, and up to 33 percent in some special cases, demonstrating the estimated energy savings that can be expected from heterogeneous processors with DVFS.

Finally, in “Neuromemristive Systems: Boosting Efficiency through Brain-Inspired Computing,” Cory
Merkel, Raqibul Hasan, Nicholas Soures, Dhireesha Kudithipudi, Tarek Taha, Sapan Agarwal, and Matthew Marinella offer their design of a neuromemristive system (NMS), a neuromorphic computing system that is emerging as an alternative to mainstream CMOS-based systems. Unlike several other ongoing neuromorphic computing efforts, the authors’ approach uses a disruptive technology—memristors—to try to mimic the human brain’s operation.

After providing an overview of memristor devices, the article presents a memristor-based neural multicore architecture and compares it to classical static RAM and reduced-instruction-set computing multicore architectures. Two different NMS designs are proposed—one based on an isolation transistor with a memristor at each cross-point within the crossbar, and one without such isolation transistors. A preliminary evaluation of the authors’ NMS accelerator shows that their approach can reduce power dissipation and improve energy efficiency in certain applications from the machine-learning domain by several orders of magnitude.

We hope you enjoy this special issue on energy-efficient computing. We are grateful to all of the authors who took the time and effort to submit manuscripts, as well as the many anonymous reviewers who helped us select for publication this outstanding collection of articles.

REFERENCES