

## Asymmetry Everywhere (with Automatic Resource Management)

Onur Mutlu, Carnegie Mellon University, [onur@cmu.edu](mailto:onur@cmu.edu), <http://www.ece.cmu.edu/~omutlu/>  
Area of focus: Data center and peta-scale

**The Setting:** Management of shared hardware resources (caches, interconnects, memory, disks, power, and lifetime reliability) among threads/applications is one of the most difficult tasks in writing high-performance parallel programs and distributed systems with shared resources (e.g., a data center on a chip). In addition, the main limiter to efficiency and performance in future systems will be power/energy consumption, which is possibly the most valuable shared resource in a many-core system.

**The Position:** History tells us that writing even sequential software is hard enough – getting programmers to optimize their code for a complex shared-resource parallel system will be a nightmare. To enable ubiquitous parallel programming, *programmers should not worry about hardware resource management at all when optimizing parallel programs*. Therefore, **future computer architectures (the hardware, the hardware/software interface, and the runtime system) should be designed to minimize programmer effort and maximize the runtime system's efficiency in its task of automatic resource management**.

Since power is a major shared resource, future many-core systems should be designed to support the management of power (along with performance and reliability) automatically across competing or cooperating applications. ***Asymmetric or partially reconfigurable components (not only processing elements, but also shared resources with different power/performance/reliability tradeoffs) are essential to enable power-efficiency using automatic resource management.*** The users and the system can specify service-level agreements (SLAs) for tasks and the runtime system should automatically exploit the underlying asymmetry to satisfy SLAs with minimal power/energy consumption. It is our vision that future runtime systems (with the support of hardware) will automatically decide where to run different tasks (or program portions) and how to exploit/morph asymmetric/configurable hardware to maximize power efficiency while satisfying application requirements.

To enable this, there needs to be much research done on questions such as 1) How should asymmetric components be designed to achieve maximum efficiency? 2) What monitoring should be performed in hardware/software for task characterization, matching to components, and mapping? 3) How should the feedback/control loop between the asymmetric components and runtime system software be designed? 4) How should the system software be designed to automatically manage resources based on dynamic demand? Given the complexity of these tasks and the need for automatic discovery of task demands, we believe statistical/machine learning techniques should play an important role in future systems. Automatic resource management with hardware/software cooperation will also enable a system that continuously optimizes itself by adapting to dynamic changes in operational environment and workloads.

**Role of Academic and State-Sponsored Research:** Hardware or software alone cannot solve the difficult problems we face in parallel programming in future data center systems, which will essentially be a **distributed aggregation of distributed-systems-on-chip**. Cooperative academic/industrial research will be the driving factor for future. This research should be bold and collaborative – researchers should break all the barriers between hardware and software and optimize the entire stack, yet, of course, tasks should be placed carefully at different levels. We will need to invent new ideas and approaches using theory, machine learning, expert systems, computer architecture, networking, distributed systems, and systems software. Such collaborative efforts are essential for enabling ubiquitous parallel programming and need to be encouraged early on in research careers.

**Parallel System Building Efforts:** Academic research should be all-inclusive. Both systems building and theoretical/algorithmic research are essential for fruitful research and provide different types of insights. Both approaches should be supported with the utmost enthusiasm.