THE NATIONAL ACADEMY OF SCIENCES (NAS) RECENTLY RELEASED A NATIONAL RESEARCH COUNCIL (NRC) STUDY ENTITLED THE FUTURE OF COMPUTING PERFORMANCE. IT CONFIRMED WHAT WE’RE ALL OBSERVING—THAT IS, THAT COMPUTING IS UNDERGOING A RADICAL change.¹ The study summarizes why the processing speed of individual computer chips will no longer increase dramatically each year, discusses the implications of this for computing, and outlines what is needed to continue to improve computing performance in the future.

Prior to 2004, the computer microprocessor's clock frequency (and the chip's peak processing power) was increasing by a factor of 100 every decade; in 2004, due to the electronic properties of CMOS technology silicon chips, it became impossible to increase the clock frequency without increasing the power density to unacceptable levels. Applications have generally run faster each year with little or no changes in the program. In 2004, the rate of increase slowed to about a factor of two per decade. The clock speed has now reached a plateau of approximately 3-4 GHz. The interpretation of Moore's law that implies that clock frequency will continue to increase exponentially is no longer valid (see http://en.wikipedia.org/wiki/Multi-core_processor). The impact of this has been rumbling through the world of high-performance computing since 2004 and is now starting to affect general computing. Up until 2004, the continual reduction in computer chips' feature size let chip manufacturers increase clock frequency while maintaining a constant power density for single processors.

This is not good news for scientific and general computing. As the NRC study documents, progress in computing—and in a significant portion of US international technology competitiveness—is due to the continued growth of computing performance. However, all is not lost. It will still be possible to continue reducing microprocessors' feature size for at least another five to 10 years, so Moore's law will continue to be valid as stated in his original paper.² Although microprocessors' raw performance speed will no longer improve, chip manufacturers can continue to increase the number of processors per chip without increasing the power density so that chips can perform more operations per second, even if single processor performance doesn’t increase. This has led to “many-core” chips, which hold an increasing number of processors. Companies have announced chip designs with several hundreds of cores (http://en.wikipedia.org/wiki/Multi-core_processor), so we can reasonably expect chips with thousands of cores and even more in the future.

Unfortunately, achieving better performance with this new, massively parallel computer architecture requires major changes in most software.³ The days of obtaining performance increases due to higher processor speed are mostly over. Most computer programs are sequential—that is, they perform operations sequentially, not in parallel, which is problematic as exploiting the new, massively parallel computers will require programs that can perform many calculations in parallel, something only a few highly advanced programs are now able to do. The NRC study describes the implications of this for computing:

• Massively parallel codes are more complex than sequential codes, and programming tools are rudimentary.
• Modifying existing applications to exploit these new computer architectures will be a substantial job and might not even be feasible as not all sequential algorithms can be made parallel.
• The architectures for massively parallel computers are still evolving rapidly, so programmers will be trying to hit a moving target.

To increase the challenge, it’s likely that single chips will have many different types of processor architectures. For instance, many chip vendors have achieved increased...
processor speed of factors of 10 to 100 or more using data-streaming concepts employed by general-purpose GPUs (GPGPUs; see http://en.wikipedia.org/wiki/GPGPU). Exploiting these “heterogeneous” architectures will require building applications of even greater complexity.

The NRC study recommended six high-priority research and development programs that are needed to make the transition from sequential computing to massively parallel computing with heterogeneous processors:

• algorithms that can exploit parallel processing;
• new computing “stacks” (applications, programming languages, compilers, runtime/virtual machines, operating systems, and architectures) that execute parallel rather than sequential programs and effectively manage software parallelism, hardware parallelism, power, memory, and other resources;
• portable programming models that allow expert and typical programmers to express parallelism easily and allow software to be efficiently reused on multiple generations of evolving hardware;
• parallel-computing architectures driven by applications, including enhancements of chip multiprocessors, conventional data parallel architectures, application-specific architectures, and radically different architectures;
• open interface standards for parallel programming systems that promote cooperation and innovation to accelerate the transition to practical parallel computing systems; and
• engineering and computer science educational programs that incorporate an increased emphasis on parallelism and use a variety of methods and approaches to better prepare students for the types of computing resources that they’ll encounter in their careers.

The report summarizes each topic’s need and scope and argues convincingly that we must address these areas if we are to have applications that can exploit the new generation of computer architectures.

At the 22 March 2011 meeting of the National Academies in Washington, DC in which review panel members outlined the study, a member of the audience said, “There have been many studies in the past that pointed out the need for more R&D on these topics, and nothing happened. Why will it be different this time?” Samuel Fuller, the study chair, said the difference is that both chip and computer vendors and the federal agencies now recognize the problem and consequences if application performance doesn’t improve. Specifically, vendors recognize that no one will buy their new products if those products don’t offer greater functionality and performance, while the government recognizes that US economic and military competitiveness are strongly tied to increased computer performance. The NSF and the US Department of Energy are beginning to address these issues.

But will these efforts be enough? Even if all of the research topics receive considerable support, it will be many years before the impact is felt. It won’t be easy and will take time for the computing community to go from approximately 1,000 parallel programmers to 100,000–200,000 parallel programmers. New computer languages and interface standards take time to mature and achieve broad adoption, and so on. Although it’s clear that computing has “suddenly” become a lot more challenging, the study also makes it clear that there are ways to continue improving computing performance if we take the right steps. As a start, I recommend that everyone in computing read the NRC study.

References

Douglass Post is associate editor in chief of CISE. Contact him at post@ieee.org.