Research Challenges in Computer Science and Engineering

Jonathan Turner
Washington University
jst@cs.wustl.edu

The modern era for computer science and engineering is now 50 years old. It's been an exciting and dynamic time and the contributions that our field has made to society have been profound. As the field has matured, there is a growing sense of diminished possibilities. This is understandable. Certainly in the more mature parts of the field, there is less unexplored territory. The low-hanging fruit has been picked. It is more difficult for new ideas to have a profound influence on established thinking, or a dramatic impact on commercial practice. On the other hand, the pervasiveness of information technology in modern society means that improvements in mature areas can have a large impact, through the multiplicative effect of the mass market. Information technology is no longer the private preserve of a technical elite. In the modern era, advances in mature areas may be harder to come by, but their absolute impact on society can be huge. This is a golden age for our field and in many important ways, the potential for research is unprecedented.

While the well-established parts of our field have matured, its boundaries continue to expand. Computer science and engineering has applications in all walks of modern life and in many areas we are just beginning to scratch the surface of what is possible. The ongoing revolution in the underlying electronics technology continues to create new opportunities for innovation. Computational barriers are coming down every day, bringing previously intractable problems within the realm of practical technological solutions. Real-time natural speech recognition and advanced computer vision systems are just two of the more obvious examples of technologies that are poised on the brink of widespread, practical applications, as the combination of technology and better understanding of the fundamental problems, removes the barriers that have prevented effective solutions in the past.

One common characteristic of this new era is a growing emphasis on systems projects designed to demonstrate new research ideas and allow them to be evaluated in a realistic experimental setting. Increasingly, these projects require coordinated efforts by large groups, not simply the smaller efforts of individual researchers. One of the challenges that we must meet in this era is to find new models for organizing academic research, to allow the development of complex systems by larger research teams. While there have been many efforts along these lines over the years, the record of success is mixed. We need to do a better job, if academic research is to play a serious role in the development of advanced systems, and continue its record of important contributions to society.

The remainder of this note describes two research challenges in systems that offer significant practical benefits to society, while requiring new ideas, innovative thinking and major R&D efforts, if the challenges are to be successfully met.

Challenges for Networking – Merging Computing and Communication

One crucial challenge for our field is to take a major new step in networking and communications. While the Internet has been a great success, the limitations of its core protocols are very constraining and are becoming an impediment to progress. The last ten years has seen a very effective attack on the core

problem of boosting the performance of Internet routers. The next ten years will need to focus on enriching the functionality of communication networks, using both embedded computing subsystems and reconfigurable hardware.

The greatest challenge here is to conceptualize an effective model for large-scale programmable networks, that will allow a world-wide infrastructure of network-embedded computational resources to be placed at the disposal of application developers with innovative ideas for new services designed to benefit end users. While the active networking initiatives of recent years have begun to stimulate thinking in this area, the most well-known model within the active networking field (packets being processed as programs) does not appear to be particularly promising. More structured methods are needed, that will enable the underlying network infrastructure to provide appropriate low-level services to application developers, to allocate resources appropriately (on a network-wide scale) and to provide the isolation and protection needed to ensure consistent, high quality service to end users.

There are significant challenges in the design of routers, with embedded computational capabilities. The performance tradeoffs for processors in this environment are quite different from those of conventional processors optimized largely for desktop computing environments. The computational workload in this environment is well-suited to systems designed for task-level parallelism, placing a premium on systems with multiple processors on a chip and multiple hardware contexts per processor. The potential role of reconfigurable hardware in this context is particularly intriguing. The capabilities of reconfigurable hardware have expanded dramatically in recent years and the performance gap with fixed hardware devices has narrowed significantly. There is a tremendous potential here, if we can find effective models for incorporating reconfigurable hardware into flexible and general purpose computational engines and find innovative ways to speed up the configuration process (the place & route computations for large FPGAs today can take hours – this must be cut to seconds).

Challenges in Hardware Design – Coordinating Parallelism

The last decade has seen important advances in computer-aided design tools and hardware design methodologies, generally. Hardware description languages allow digital hardware designers to operate at significantly higher levels of abstraction than previously, and the supporting CAD tools automate substantial parts of the design process. However, in important ways, hardware design has advanced hardly at all in the last thirty years. While CAD tools help automate the design of smaller hardware components, they provide little or no assistance in the assembly of these components into larger, more complex systems. The burden of managing the interfaces among components and coordinating the parallel operations of the dozens of components that complex chips contain, falls entirely on designers. There are no automated tools to support this, forcing designers to use manual methods, involving painstaking documentation of the data formats and timing requirements of all interfaces and the construction of elaborate timing plans for the coordination of the myriad activities taking place within the system.

There are deep intellectual challenges here. Even in the software domain, automated tools provide little intellectual assistance to the human designer when it comes to conceptualizing and managing parallelism. The hardware domain brings its own new challenges to this problem, while also offering the possibility of novel solution methods. Timing-driven coordination is common in hardware, but is rarely used in software systems. This raises the possibility of automated tools that use timing-driven coordination methods, but which free the designer from the tedious and error-prone task of constructing elaborate timing plans to achieve the desired effect.

The pervasiveness of parallelism in the hardware domain means that the potential impact of better solutions to managing parallelism on hardware design can be huge. Parallelism is not a small niche within the hardware design domain, it is the essence of the field. A successful attack on this problem can have immediate practical benefits and can potentially deepen our fundamental understanding of parallel computation allowing us to harness parallelism more effectively in other contexts as well.

BIO:

Jonathan S. Turner received the MS and PhD degrees in computer science from Northwestern University in 1979 and 1981. He holds the Henry Edwin Sever Chair of Engineering at Washington University, and is Director of the Applied Research Laboratory. The Applied Research Laboratory is currently engaged in a variety of projects ranging from Active Networking, to Network Management and Visualization, to WDM Burst Switching.

He served as Chief Scientist for Growth Networks, a startup company that developed scalable switching components for Internet routers and ATM switches, before being acquired by Cisco Systems in early 2000.

Professor Turner's primary research interest is the design and analysis of routers and switching systems, with special interest in systems supporting multicast communication. His research interests also include the study of algorithms and computational complexity, with particular interest in the probable performance of heuristic algorithms for NP-complete problems.

Turner is a fellow of ACM and a fellow of the IEEE. He received the Koji Kobayashi Computers and Communications Award from the IEEE in 1994 and the IEEE Millennium Medal in 2000. He has been awarded more than 20 patents for his work on switching systems and has many widely cited publications.