Controlling Core Temperatures and Saving Energy with an Adaptive Runtime System

Laxmikant (Sanjay) Kale and Osman Sarood

University of Illinois

Thursday, October 13th @ 4PM, 3405 Siebel Center

Professor Laxmikant Kale works on various aspects of parallel computing, with a focus on enhancing performance and productivity via adaptive runtime systems. His collaborations include the widely used Gordon-Bell award winning (SC'2002) biomolecular simulation program NAMD, and other collaborations on computational cosmology, quantum chemistry, rocket simulation, space-time meshes, and other unstructured mesh applications. He takes pride in his group's success in distributing and supporting software embodying his research ideas, including Charm++, Adaptive MPI and the ParFUM framework.

Dr. Kale received the B.Tech degree in Electronics Engineering from Benares Hindu University, Varanasi, India in 1977, and a M.E. degree in Computer Science from Indian Institute of Science in Bangalore, India, in 1979. He received a Ph.D. in computer science in from State University of New York, Stony Brook, in 1985.

He worked as a scientist at the Tata Institute of Fundamental Research from 1979 to 1981. He joined the faculty of the University of Illinois at Urbana-Champaign as an Assistant Professor in 1985, where he is currently employed as a Professor.

Abstract

Thermal issues and energy issues will present dominant challenges for at least the next decade. This is true for client-side computing, and even more important for high-end exascale computing. I will present my research group's approach to this problem, based on adaptive runtime systems.

Cooling energy is a significant component of energy spent by a data center. The room temperature is maintained at a very low level to avoid hot-spots --- because some processors are likely to heat up more than the others. On-chip temperature sensors permit real-time monitoring. DVFS capabilities allow one to change the frequency and voltage while the computation is running, at a very low cost. So, in principle one could simply monitor the temperature continuously and change frequencies up and down depending on whether the chip is getting too hot or not. However, for parallel applications running on clusters, this presents a major challenge: slowing down a core slows down the entire parallel application because of dependencies. We handle this challenge by migrating objects away from the slowed-down processors under a sophisticated load-balancing strategy.

The machine energy itself presents additional sets of challenges. Although, in principle, the power increases nonlinearly with frequency, the idle power is a significant component of the total power consumed. Saving energy by slowing down processors is elusive in this context. We show how techniques based on runtime instrumentation are helping us tackle this problem. These techniques are helpful in client-side setting, data centers as well as HPC clusters.

Illinois-Intel Parallelism Center (I2PC) Distinguished Speaker Series

I2PC

i2pc.cs.illinois.edu

http://media.cs.illinois.edu/live/I2PClive.asx

http://i2pc.cs.illinois.edu/chat