Low-Impact Monitoring of Streaming Systems

PI: Roger Chamberlain, Co-PI: Ron Cytron, Grant No. 0931693, 10/2009 – 9/2012
Dept. of Computer Science and Engineering, Washington University in St. Louis

The centerpiece of the activities to date on this project is the design, development, and implementation of the TimeTrial performance monitor. TimeTrial is a tool that enables observation of critical performance properties of streaming data applications without significantly perturbing the execution of the application under observation. It supports applications deployed on architecturally diverse computer systems, initially including the combination of multicore processors and/or field-programmable gate arrays (FPGAs).

Streaming data applications are typically characterized by a pipeline of computation kernels with communication between kernels via explicitly declared channels. When deployed on architecturally diverse computers, individual kernels might be assigned either to traditional processor cores or to non-traditional computing resources such as FPGAs or graphics engines. In this research, we are investigating approaches to understanding the performance of such applications in a minimally invasive manner. Central to our approach is a willingness to dedicate hardware resources (area on an FPGA, a processor core on a multicore chip) to the performance monitoring task in an attempt to have the performance monitoring actions not perturb the execution of the application itself.

Toward this end, the TimeTrial performance monitor has the following properties:

- TimeTrial deploys an agent on each physical chip that performs computation in the application. This agent is responsible for low-impact monitoring of the computation and aggressive compression of the accumulated performance data that result from the monitoring.
- The performance data compression is lossy, guided by the user, to minimize the utilization of shared inter-chip communication pathways by TimeTrial.
- The user guidance for both what is to be monitored and how it is to be aggregated (i.e., compressed) is specified via an input language that guides the deployment of the TimeTrial measurement infrastructure.

In addition to our work on the TimeTrial performance monitor, we are also pursuing investigations into the automatic design space exploration of streaming data applications deployed on architecturally diverse systems. This exploration is guided by topological information provided to the optimization solver via a queueing network model of the application's performance. This queueing network model is, in turn, calibrated, verified, and validated using TimeTrial.

Further, we are expanding the set of example applications we have currently deployed to include several that are common with benchmark sets for other streaming data computational environments, such as Brook and StreamIt. These additional applications include Burrows-Wheeler transforms, JPEG encoding, triple DES encryption/decryption, and sorting.