

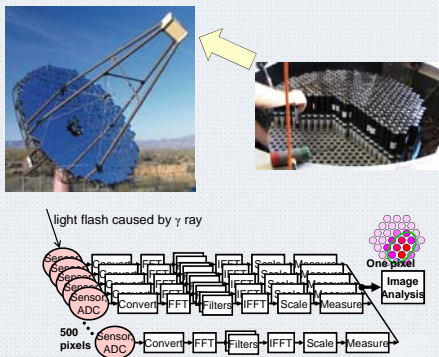
Low-Impact Monitoring of Streaming Systems

Roger D. Chamberlain and Ron K. Cytron

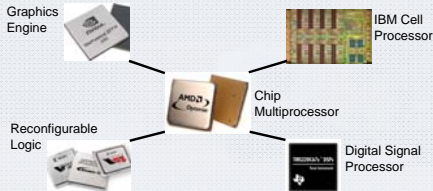
Department of Computer Science and Engineering, Washington University in St. Louis



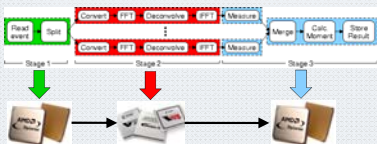
Streaming Data Computation



Architecturally-Diverse Computers

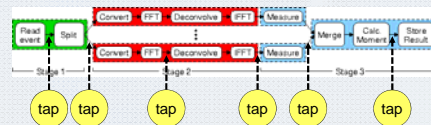


Mapping to Diverse Resources



Where is the Performance Bottleneck in an Application?

To understand performance, we would like to collect time stamped data traces at multiple taps



1. Simulation Models

- Excellent observability
- Concerns with fidelity of model
- Slow execution of simulation

2. Instrumented Execution

- Requires implementation be complete
- Limited observability
- Instrumentation can impact performance

How can we observe the real execution without perturbing its performance profile?

1. Dedicate compute resources to the measurement system

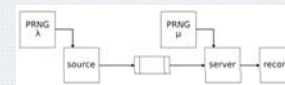
2. Use user-selected data compression to enable low-impact monitoring

TimeTrial: Measuring Performance

TimeTrial Agents

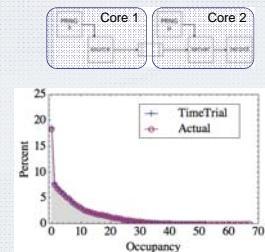
- Deployed on each compute resource
- Monitor traces from taps
- Aggregate trace data into summary statistics
- Periodically report results to server
- Server aggregates globally and displays results

Micro-benchmark Application



- Designed to mimic M/M/1 queuing system
- Application records queue occupancy
- Source and server can be allocated to distinct compute resources (e.g., proc. core or FPGA)

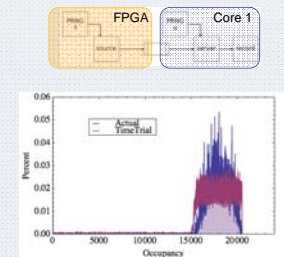
Measuring Queue Occupancy



- Excellent agreement with actual occupancy

Monitoring Virtual Queues

- Measure inter-insertion and inter-departure times
- Use to drive discrete-event simulation of queue



- Sanity check with sub-queue occupancies

Research Issues

- Construct efficient TimeTrial agents
 - FPGA [SoCC '09], CPU [ASAP '11], GPU
- Design performance query language [ERSA '10]
 - Specify measurements, assertions [HPCC'11]
- Compiler to generate low-impact instrumentation
 - Optimize data aggregation on each resource
 - Minimize communication between agents
- Unify time across resources [SAAHPC '10a]
- Understanding performance
 - Create performance models
 - Calibrate and validate models [SAAHPC '10b]