Use of Simple Analytic Performance Models for Streaming Data Applications on Diverse Architectures

Jonathan Beard, Roger Chamberlain

Overview

A model is introduced that uses a generalized gain/loss maximum flow algorithm along with a stochastic M/M/1 queuing network to calculate throughput and estimate buffering capacity at each edge in a streamed processing application.

The method presented is applicable to any application that can be formulated as a directed graph.

This model can be used where buffering capacities along “virtual-queue” edges may not be known.

Simple resource sharing models are used to explore their ability to function in place of computationally expensive models.

Validation of the flow model approach shows that it works very well (often within 10%), despite the simple sharing models.

Results

Simple sharing models based independent of flow and queuing models.

A tool called StreamAnalyst was developed to run models on streaming applications.

Multiple synthetic streaming applications with both deterministic and incremental service rates were used, as were generated kernels compiled into C and VPL by StreamModeler, along with native tools (GCC and ATLAS). A platform was then used to tune the specific configuration.


Hardware

Machine 1: 2 x Xeon 2.2GHz 4GB RAM AMD Opteron. 32GB RAM
Machine 2: 4 x Xeon interleave 2GB, 4GB RAM

Throughput Modeling

1. Single Streaming Application
2. Implementations of each Kernel
3. Assign Kernels to Compute Resources
4. Measure each Kernel in Isolation
5. Use Flow Model to Deduce Flow at each Edge

References and Acknowledgements


This work supported by NSF grants CCF-0645598 and CCF-0831852 as well as by Exage, Inc.