CS554 Project Ideas

MATRIX:MonSim - Evaluating Communication Overheads for Distributed Monitoring with Aggregation Trees through Simulation

Overview

A distributed monitoring framework serves as an important building block for large-scale data aggregation and continuous event monitoring applications, such as cluster and grid resource monitoring. The underlying aim is to provide a global view of information in the system at a reasonable cost and within a specified precision bound.

The main activities involved in maintaining a global view include one to all multicast communication and all to one reduction communication. The communication is usually based on a hierarchical design where an aggregation tree of communicating components is built, and the edges represent the data flow. The multicast communication is from the root spreading the information all the way down to all the leaf nodes, while the reduction communication is aggregating the information in all the leaf nodes to the root. The basic problem is to find the optimal fan out and height of the tree to achieve the best (smallest) latency for large-scale systems; the more extensive problem is about resiliency, which means how to rebuild the tree efficiently when node failures happen. We will explore these problems through simulation on top of PeerSir or SimMatrix.

Relevant Systems and Reading Material

MRNet: ftp://ftp.cs.wisc.edu/paradyn/papers/Roth03MRNet.pdf

Ganglia:

http://www.ittc.ku.edu/~niehaus/classes/750-s07/documents/ganglia-parallel-computing.pdf

PeerSim: http://peersim.sourceforge.net/

SimMatrix paper: http://datasys.cs.iit.edu/publications/2013_HPC13-SimMatrix.pdf

SimMatrix source code: http://datasys.cs.iit.edu/~kewang/software.html

Methodology

Simulation the aggregation tree up to Exascale on top of PeerSim or SimMatrix

Preferred/Required Skills

Required: Java; Preferred: Discrete Event Simulation

Parameters

fan out, tree height, and failure rate

Performance Metrics

latency, speedup (vs flat or centralized pattern)

Project Mentor

Ke Wang, http://datasys.cs.iit.edu/~kewang/