On Massively Parallel Simulation of Large-Scale Fat-Tree Networks for HPC Systems and Data Centers

Ning Liu, Xian-He Sun, Dong (Kevin) Jin

Department of Computer Science, Illinois Institute of Technology, Chicago, Illinois.

Introduction & Motivation

Fat-tree topologies have been widely adopted as the communication network in data centers in the past decade. Nowadays, high-performance computing (HPC) system designers are considering using fat-tree as the interconnection network for the next generation supercomputers [1]. For extreme-scale computing systems like the data centers and supercomputers, the performance is highly dependent on the interconnection networks. In this work, we present FatTreeSim, a PDES-based toolkit consisting of a highly scalable fat-tree network model, with the goal of better understanding the design constraints of fat-tree networking architectures in data centers and HPC systems, as well as evaluating the applications running on top of the network. FatTreeSim is designed to model and simulate large-scale fat-tree networks up to millions of nodes with protocol-level fidelity.

The design, evaluation and deployment of data center and HPC system is a systematic and time-consuming process. As the key component, the communication network has a significant impact on system performance. Large-scale data center and HPC system network architecture need to support a wide range of applications, each with different communication and I/O requirements. In distributed computing community, it is projected that a single data center can scale out to host millions of virtual machines or even physical servers and serve multi-millions of jobs/tasks. The requirements for building a data center network at such a scale also differ with that of the traditional data centers. The communication network must guarantee the high availability and reliability, desirable bisection bandwidth, and support for multi-tenancy. To quantify the design trade-offs of a network at a scale, it is desirable to build a large scale simulation toolkit that is capable of evaluating different design points in an efficient and cost-effective manner. A fat-tree or folded-Clos topology is the conventional and yet still the most prevalent design choice for data center communication networks.

Accuracy Experiment on Emulab & Functionality Experiment on YARNsim

To further evaluate the accuracy of FatTreeSim, we conducted experiments on Emulab. In these tests, we record the latency for each message from both the Emulab cluster and FatTreeSim and report the results in the CDF plots. We used two different configurations: a 4-port 2-tree and a 4-port 3-tree. The message size is 1,024 bytes and the number of messages is 1,000 per node. In all experiments, we observed that the curve for simulation is much smoother than the curve for Emulab. This is attributed to the fact that we model only one outgoing buffer in each outgoing port. If multiple messages are sent through this port, congestion will occur and this single point queuing effect lead to a unique waiting time for each packet.

The above figures present the FatTreeSim Scalability Experiment on Blue Gene/Q. The fat-tree model consists of 524,288 processing nodes and 20,480 switches. The total number of committed events is 567 billion. In each top subfigure, we vary the number of cores from 1,024 to 16,384 through running experiments on c1, c2, c4, c8 and c16 modes. From top-left subfigure to top-right subfigure, we vary the packet arrival interval from 200 ns to 1,000 ns. Experiments on the top subfigures are conducted using 1 Blue Gene/Q rack. In each bottom subfigure, we vary the number of cores from 2,048 to 32,768 through running experiments on c1, c2, c4, c8 and c16 modes. From top-left subfigure to top-right subfigure, we vary the packet arrival interval from 200 ns to 1,600 ns. Experiments on the top subfigures are conducted using 2 Blue Gene/Q racks. The traffic pattern is random destination.

References

[1] https://www.cscf.uri.edu/sumit/

Contact Information

Department of Computer Science
Illinois Institute of Technology
Chicago, Illinois

Acknowledgements

Many Thanks to:

Dr. Jonathan Jenkins, Dr. Mishab Mubarak & Dr. Robert Ross
Math & Computer Science Department (MCS)
Argonne National Laboratory (ANL)

http://www.anl.gov

Dr. Christopher Carothers
Department of Computer Science

Rensselaer Polytechnic Institute (RPI)

http://www.rpi.edu

Argonne Leadership Computing Facility (ALCF)

https://www.alcf.anl.gov

E-mail: jin@iit.edu