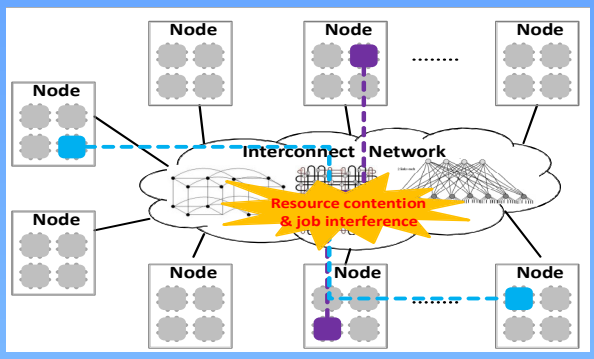
A Cooperative Framework for Topology Awareness on Large-Scale Systems

Zhiling Lan (PI) and Jia Wang (co-PI), Illinois Institute of Technology

What's the problem?

In the field of high-performance computing (HPC), users submit their jobs (e.g., scientific simulations or data processing tasks) to computer clusters through a *resource management and job scheduling* software. As computer clusters continue growing to satisfy ever-increasing computing demands, advanced ***resource management and job scheduling*** is of paramount importance. Systems today, and those anticipated in the future, are increasingly bound by their communication infrastructure and the power dissipation associated with data movement across the rapidly growing number of nodes.

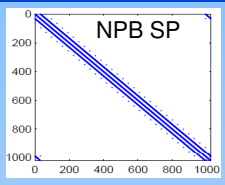


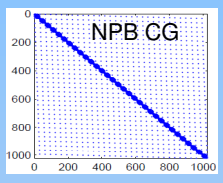
Why is it important?

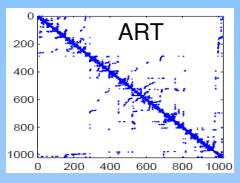
- Jobs cause resource contention and performance interference!
- Over time fragmented allocations appear!
- 5-hops latency could be 82% more than 1-hop latency!
- Power cost of inter-node communication could be 75 times more than a double-precision operation!

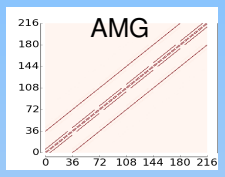
Goal:
Developing COTA (a COoperative framework for Topology Awareness) to improve communication- and power-efficiency on large-scale systems through a co-design approach.

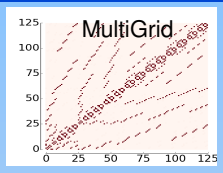
How do we solve it?

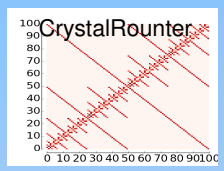




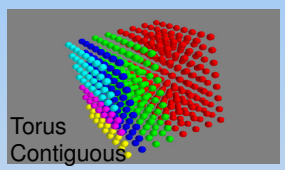


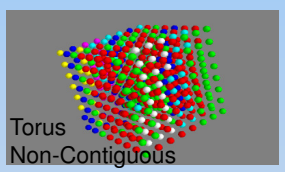


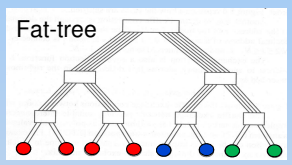


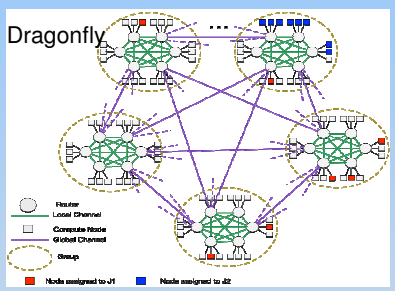


- Understand **application communication patterns**
  - ✓ communication intensity, e.g., total message amount
  - ✓ dominant communication behavior, e.g., nearest neighbor or many-to-many









- Deploy scheduling strategies for **systems with different topologies**
  - ✓ Torus connected systems using contiguous allocation, e.g., IBM Blue Gene series machines
  - ✓ Torus connected systems using non-contiguous allocation, e.g., Cray X series machines
  - ✓ Fat-tree based systems, e.g., TACC Stampede, Summit at OLCF
  - ✓ Dragonfly based systems, e.g., Cray Cascade, Aurora at ALCF

Key Findings

- On Blue Gene, we could relax network allocation (e.g., mesh allocation to replace torus allocation) for those applications dominated by local communications.
- On Cray XT6/XE6/XK7, we should choose an appropriate allocation unit for different applications. An optimal unit should be large enough to preserve neighboring communication of the application.
- On dragonfly or fat-tree systems, hybrid scheduling policies that integrate contiguous and non-contiguous allocation could reduce the worst-case performance degradation for less communication-intensive applications while retaining the performance of communication-intensive ones.

Project webpage:
http://www.cs.iit.edu/~lan/cota.html

Who are we?

- **Principal investigators:**
  Zhiling Lan and Jia Wang
- **Student participants:**
  (Ph.D.) Zhou Zhou, Xu Yang, Xingwu Zheng, Eduardo Berrocal
  (REU) Asad Patel, Tarun Gidwani, Aleksandra Kukielko
- **Collaborators:**
  Xuanxing Xiong (Synopsys), Jingjin Wu (UESTC), Wei Tang (Google), Narayan Desai (Google), Vitali Morozov (ANL), Paul Rich (ANL)

Acknowledgement

This work is supported in part **by US National Science Foundation grant CNS-1320125**. This research used data of the Argonne Leadership Computing Facility, which is supported under Contract DE-AC02-06CH11357.