Introduction to Parallel Computing (CMSC498X / CMSC818X)

Lecture 22: Performance Variability

Abhinav Bhatele, Department of Computer Science

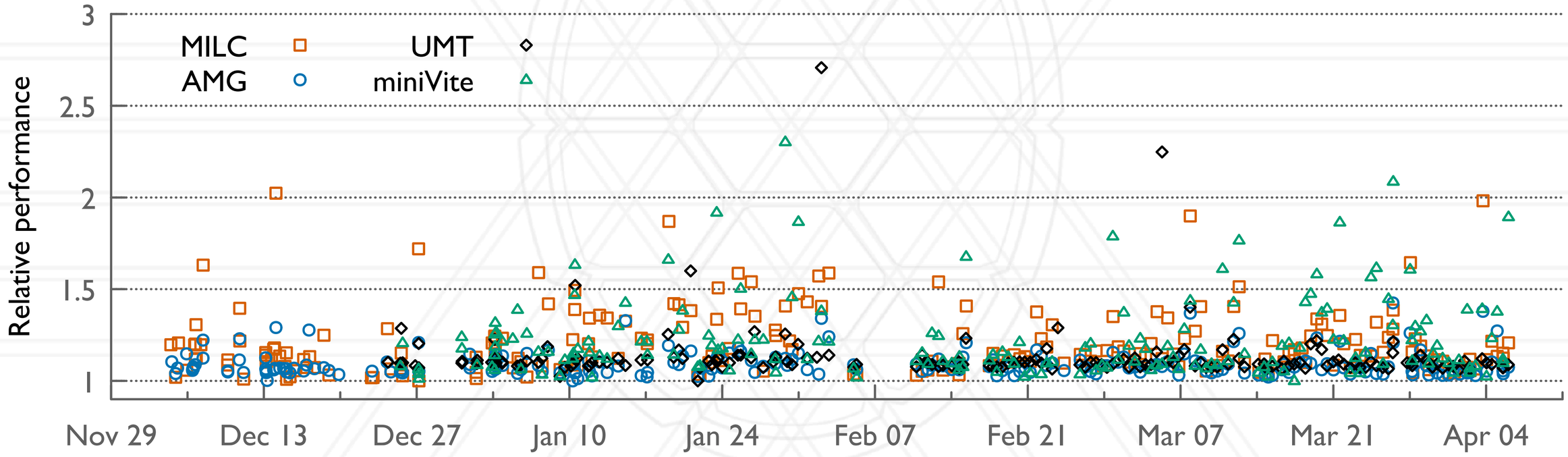
UNIVERSITY OF MARYLAND

Announcements

- E-mail Abhinav and Shoken with your preferences for the project presentation slot:
  - Provide three options in decreasing order of preference: Dec 3, 8, 10

Performance variability is a real concern

Performance of control jobs running the same executable and input varies as they are run from day-to-day on 128 nodes of Cori in 2018-2019



Bhatele et al. The case of performance variability on dragonfly-based systems, IPDPS 2020

Leads to several problems …

- Individual jobs run slower:
  - More time to complete science simulations
  - Increased wait time in job queues
  - Inefficient use of machine time allocation
- Overall lower throughput
- Increased energy usage/costs

Affects software development cycle

- Debugging performance issues
- Quantifying the effect of various software changes on performance
  - Code changes
  - System software changes
- Estimating time for a batch job or simulation

Sources of performance variability

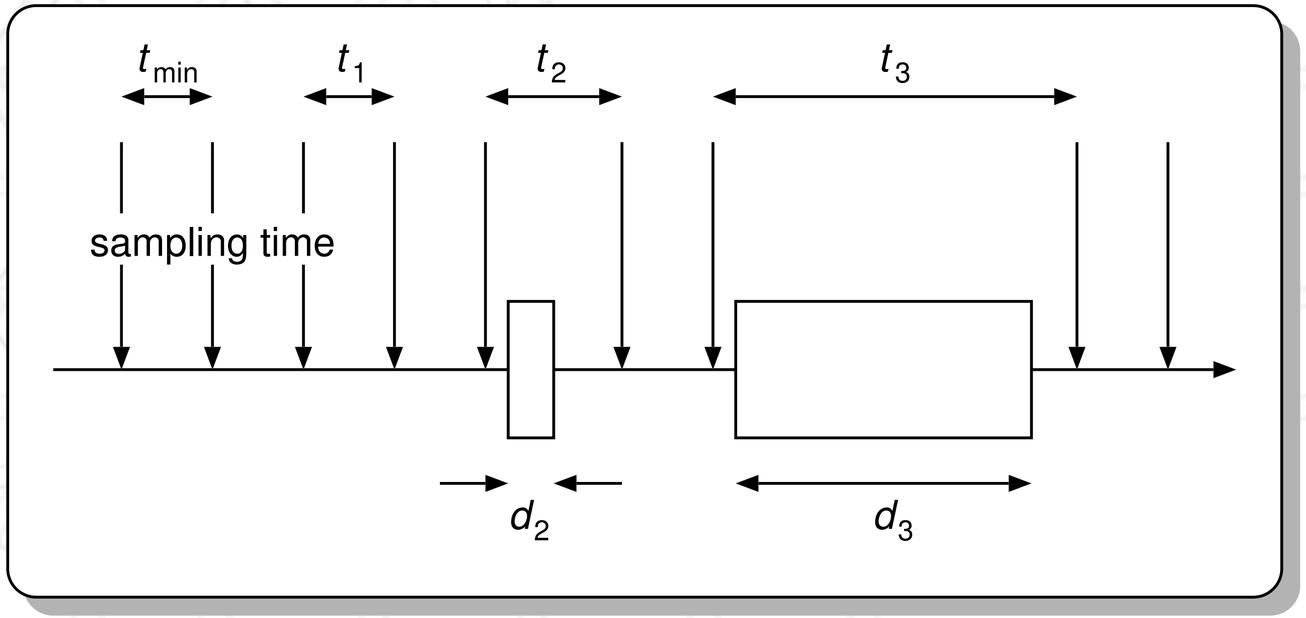
- Operating system (OS) noise/jitter
- Contention for shared resources
  - Network
  - Filesystem

Operating System

- Node on an HPC cluster may have:
  - A "full" linux kernel, or
  - A light-weight kernel
- Decides what services/daemons run
- Impacts performance predictability

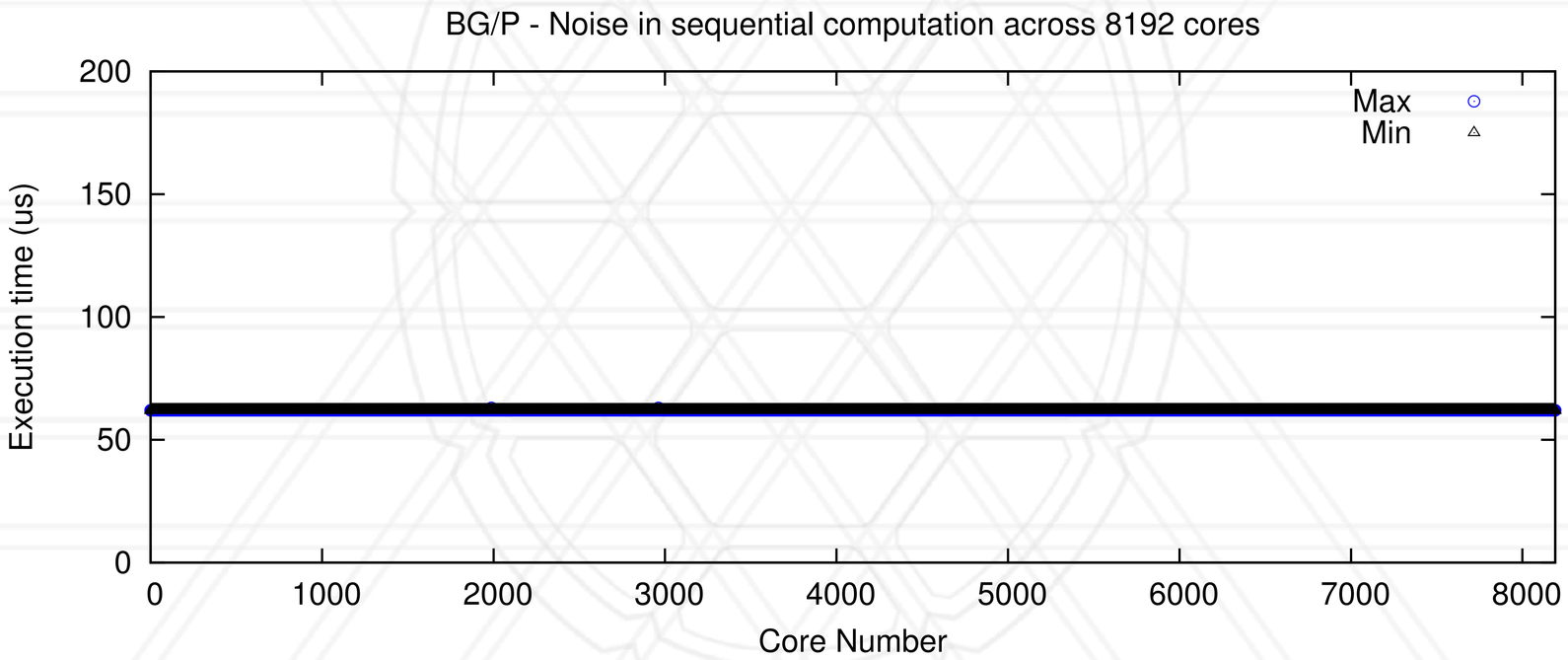
Operating System (OS) Noise

- Also called "jitter"
- Impacts computation due to interrupts by OS



Measuring OS Noise

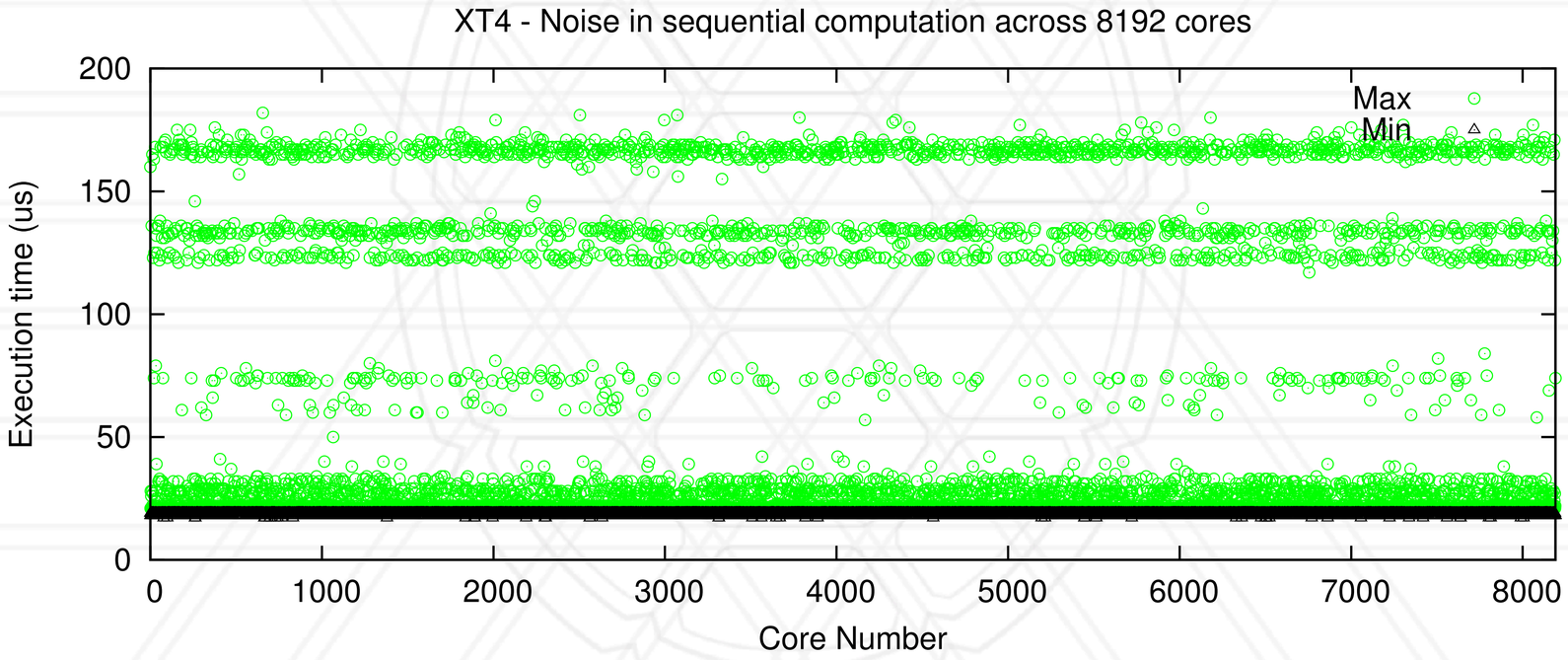
- Fixed Work Quanta (FTW) and Fixed Time Quanta (FTQ)



Benchmarks: https://asc.llnl.gov/sequoia/benchmarks/FTQ_summary_v1.1.pdf

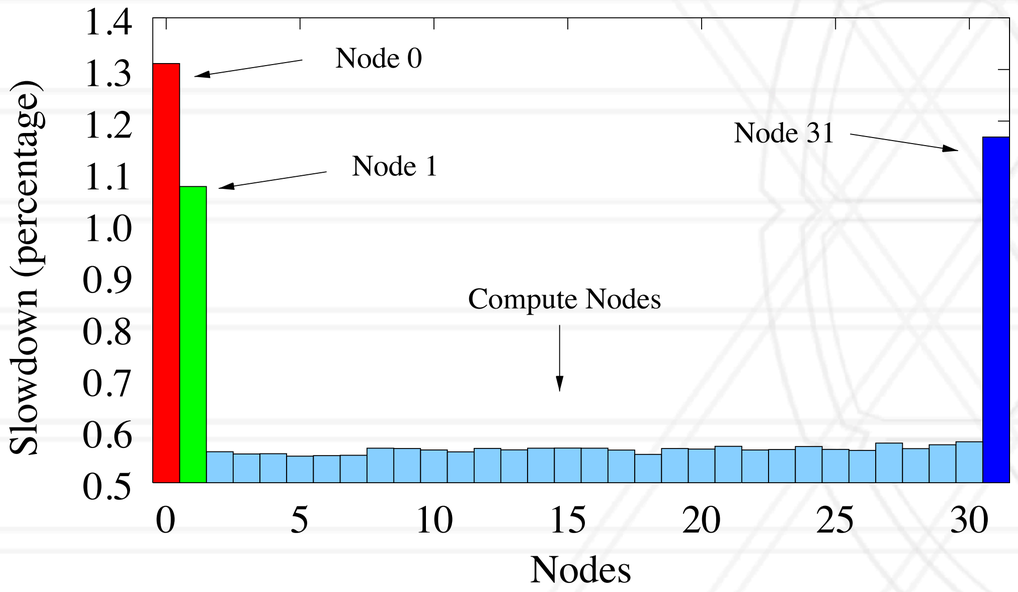
Measuring OS Noise

- Fixed Work Quanta (FTW) and Fixed Time Quanta (FTQ)



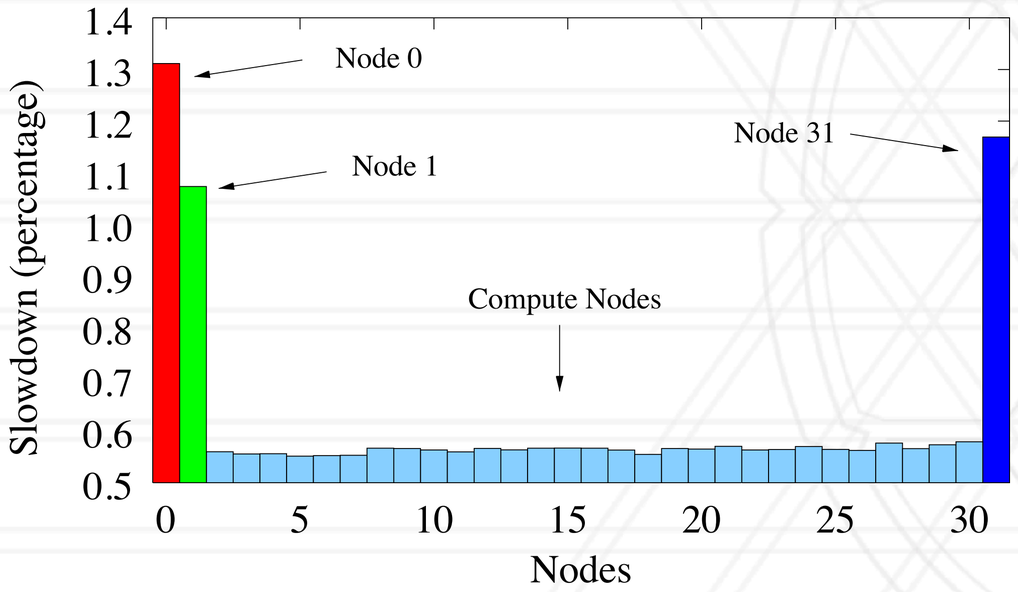
Benchmarks: https://asc.llnl.gov/sequoia/benchmarks/FTQ_summary_v1.1.pdf

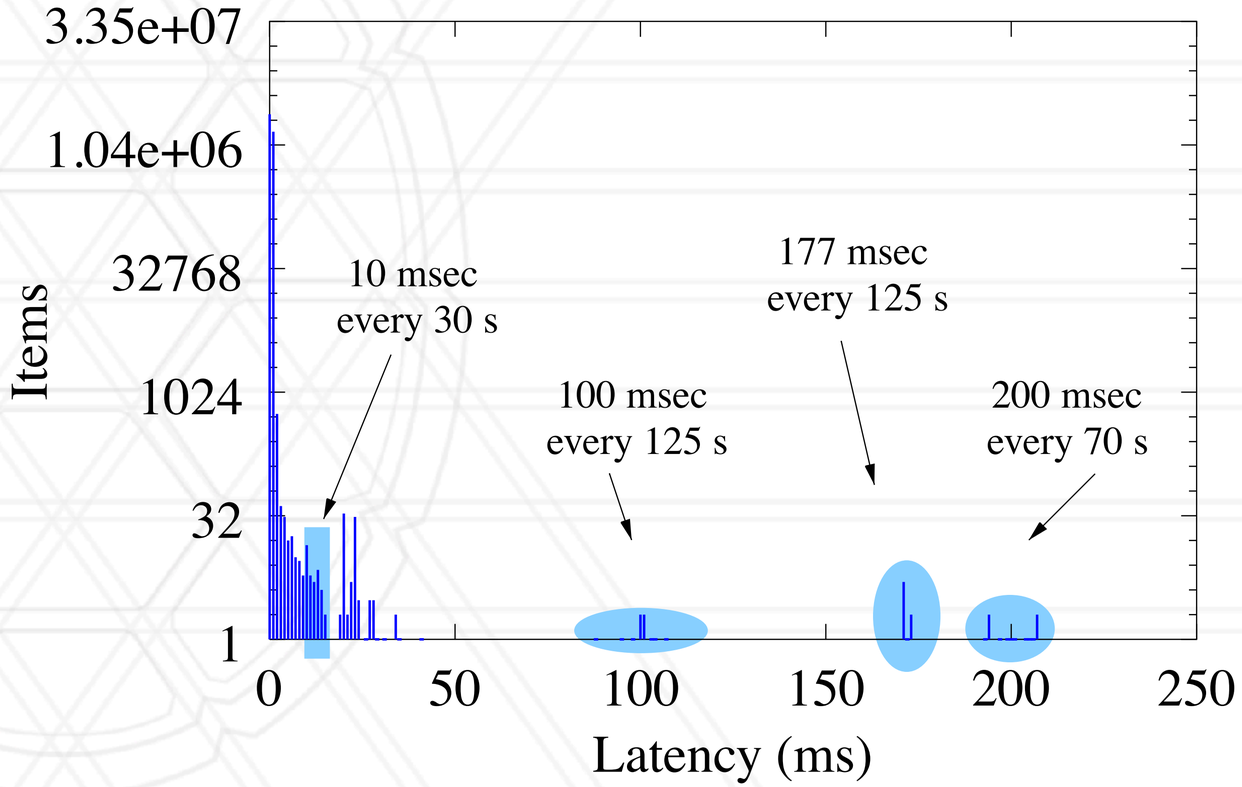
The Case of the Missing Supercomputer Performance



Fabrizio Petrini, Darren J. Kerbyson, and Scott Pakin. 2003. The Case of the Missing Supercomputer Performance: Achieving Optimal Performance on the 8,192 Processors of ASCI Q. In Proceedings of the 2003 ACM/IEEE conference on Supercomputing (SC '03). Association for Computing Machinery, New York, NY, USA, 55. DOI:https://doi.org/10.1145/1048935.1050204

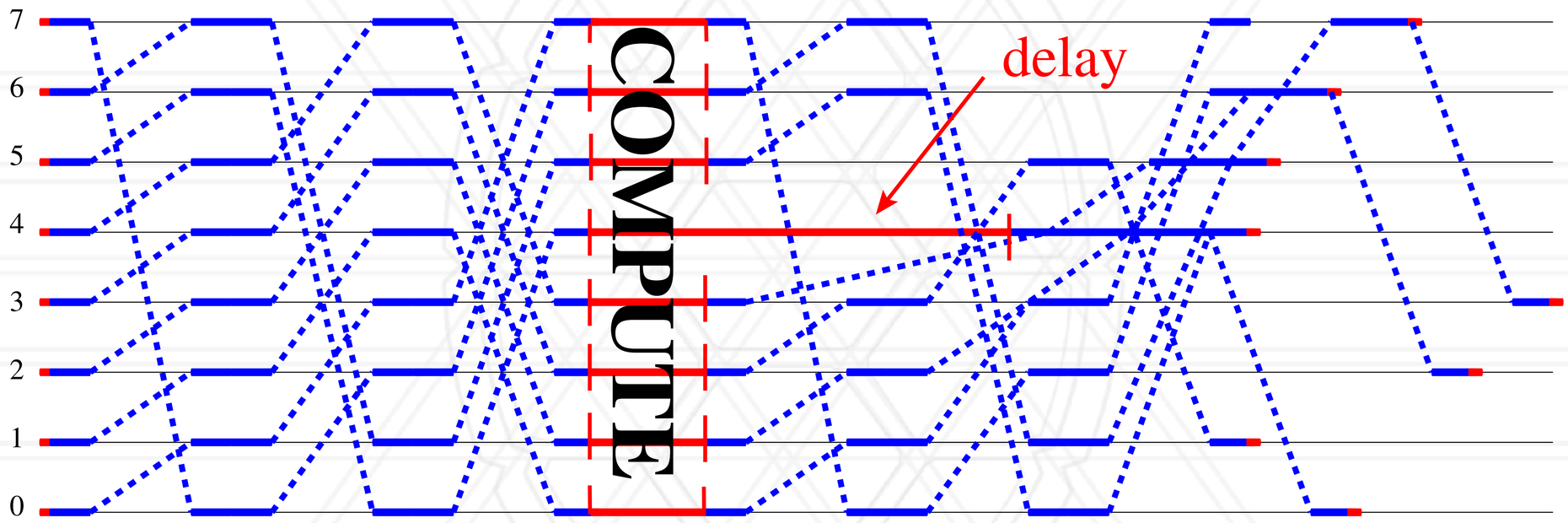
The Case of the Missing Supercomputer Performance





Fabrizio Petrini, Darren J. Kerbyson, and Scott Pakin. 2003. The Case of the Missing Supercomputer Performance: Achieving Optimal Performance on the 8,192 Processors of ASCI Q. In Proceedings of the 2003 ACM/IEEE conference on Supercomputing (SC '03). Association for Computing Machinery, New York, NY, USA, 55. DOI:https://doi.org/10.1145/1048935.1050204

Impact on communication



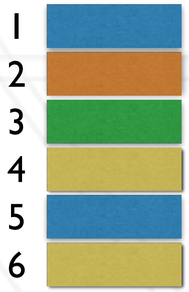
Hoefler et al.: https://htor.inf.ethz.ch/publications/img/hoefler-noise-sim.pdf

Job scheduling

Job scheduling

- HPC systems use job or batch scheduling
- Each user submits their parallel programs for execution to a "job" scheduler

Job Queue

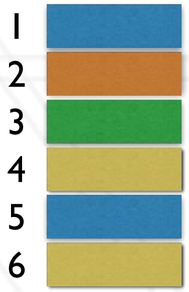


#Nodes Requested	Time Requested
128	30 mins
64	24 hours
56	6 hours
192	12 hours
…	…
…	…

Job scheduling

- HPC systems use job or batch scheduling
- Each user submits their parallel programs for execution to a "job" scheduler
- The scheduler decides:
  - what job to schedule next (based on an algorithm: FCFS, priority-based, ….)
  - what resources (compute nodes) to allocate to the ready job

Job Queue



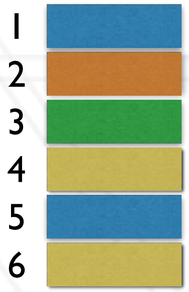
#Nodes Requested	Time Requested
128	30 mins
64	24 hours
56	6 hours
192	12 hours
…	…
…	…

Job scheduling

- HPC systems use job or batch scheduling
- Each user submits their parallel programs for execution to a "job" scheduler
- The scheduler decides:
  - what job to schedule next (based on an algorithm: FCFS, priority-based, ….)
  - what resources (compute nodes) to allocate to the ready job

- Compute nodes: dedicated to each job
- Network, filesystem: shared by all jobs

Job Queue



#Nodes Requested	Time Requested
128	30 mins
64	24 hours
56	6 hours
192	12 hours
…	…
…	…

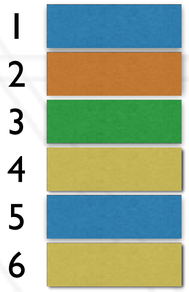
Job scheduling

- HPC systems use job or batch scheduling
- Each user submits their parallel programs for execution to a "job" scheduler
- The scheduler decides:
  - what job to schedule next (based on an algorithm: FCFS, priority-based, ….)
  - what resources (compute nodes) to allocate to the ready job

Concurrently running jobs can contend for shared resources: network, filesystem

- Compute nodes: dedicated to each job
- Network, filesystem: shared by all jobs

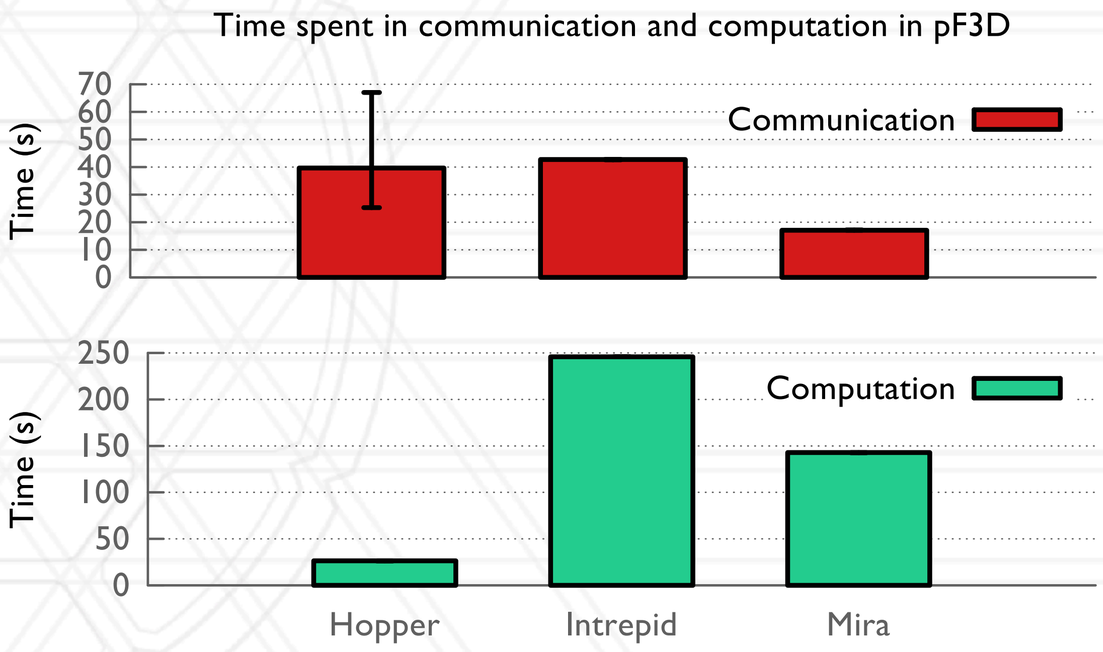
Job Queue



#Nodes Requested	Time Requested
128	30 mins
64	24 hours
56	6 hours
192	12 hours
…	…
…	…

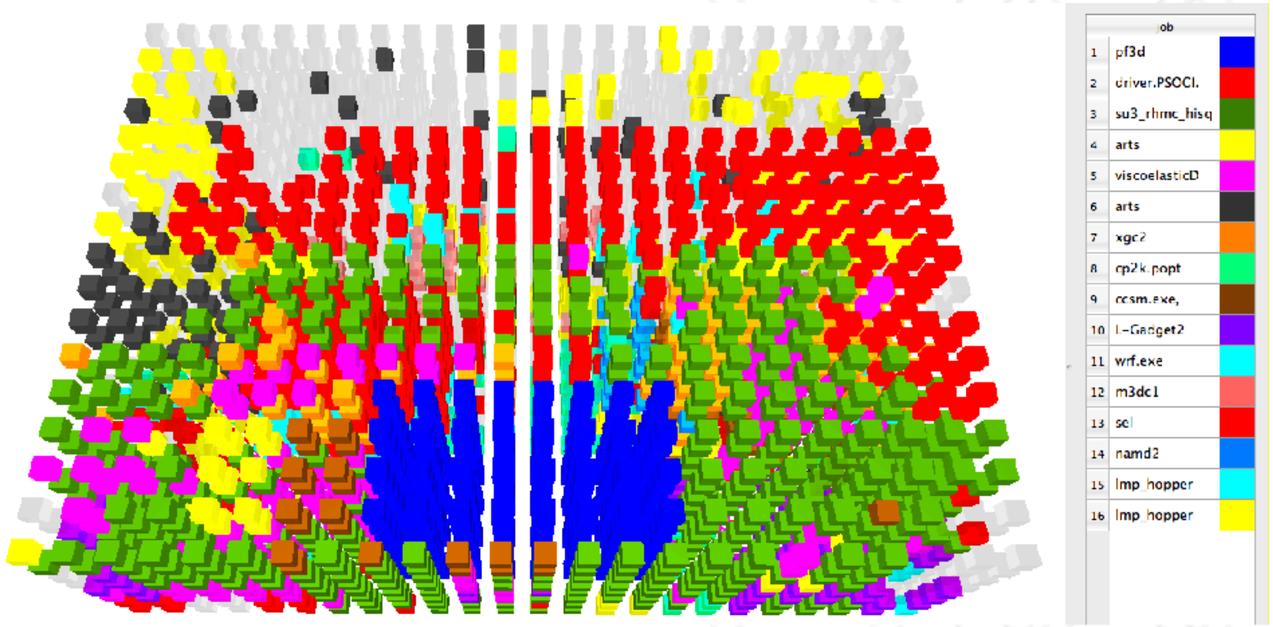
Performance variability due to congestion

- No variability in computation time
- All of the variability can be attributed to communication performance
- Factors:
  - Placement of jobs
  - Contention for network resources

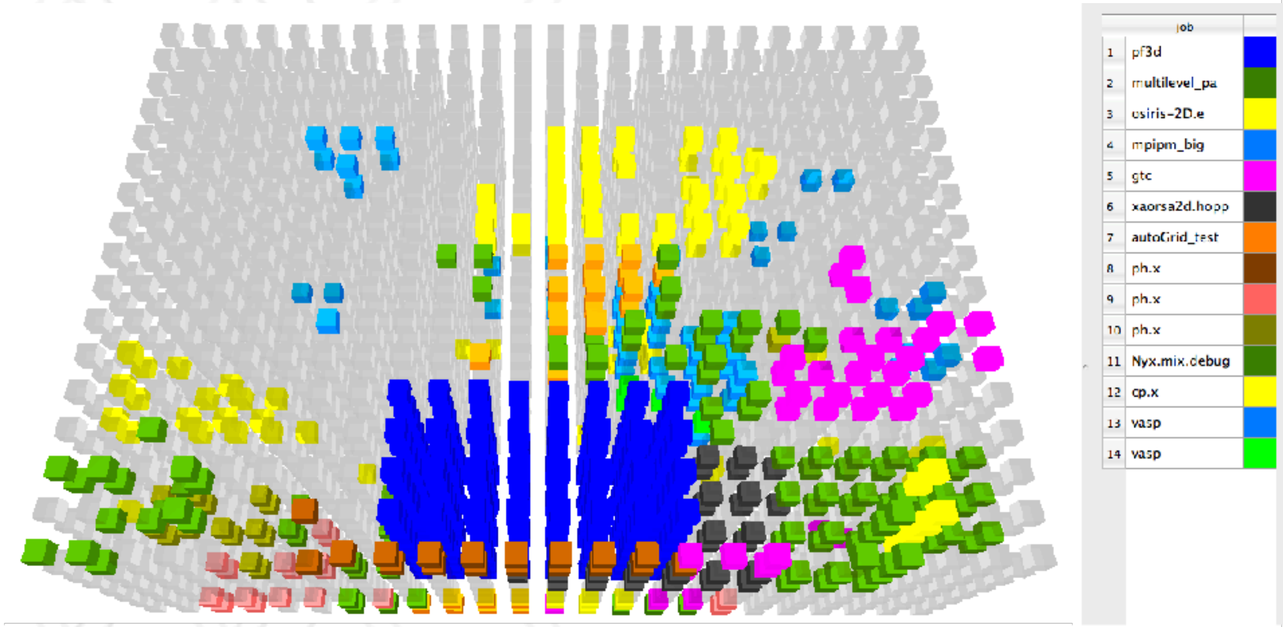


Bhatele et al. http://www.cs.umd.edu/~bhatele/pubs/pdf/2013/sc2013a.pdf

Impact of other jobs

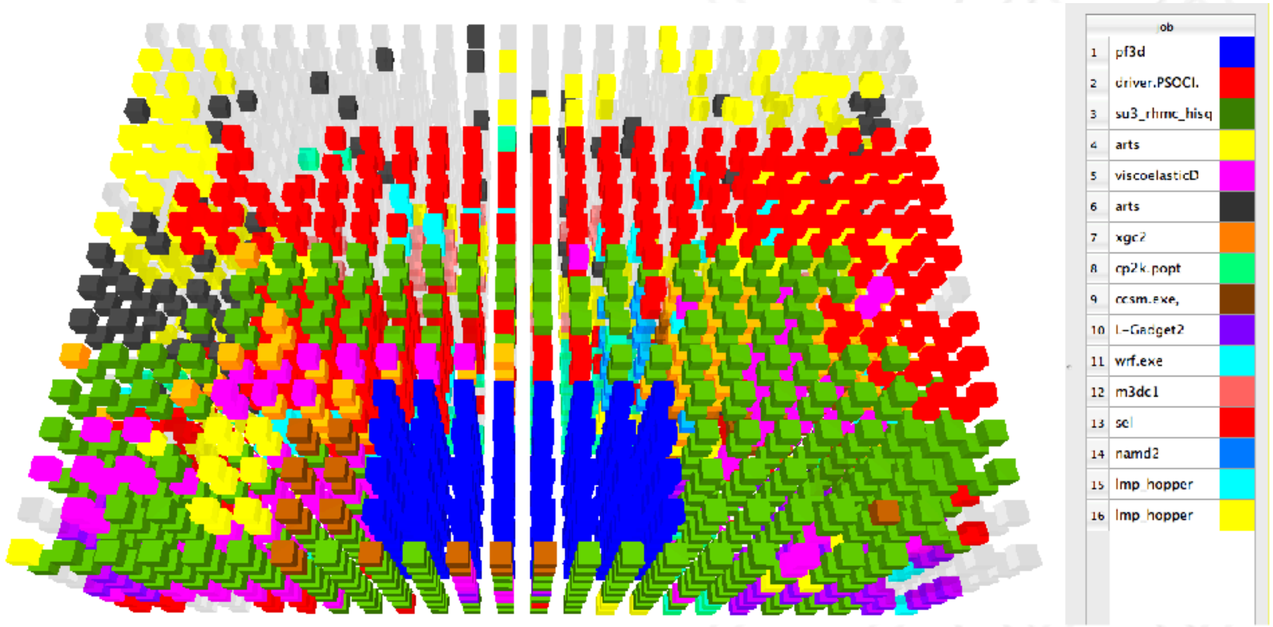


April 11



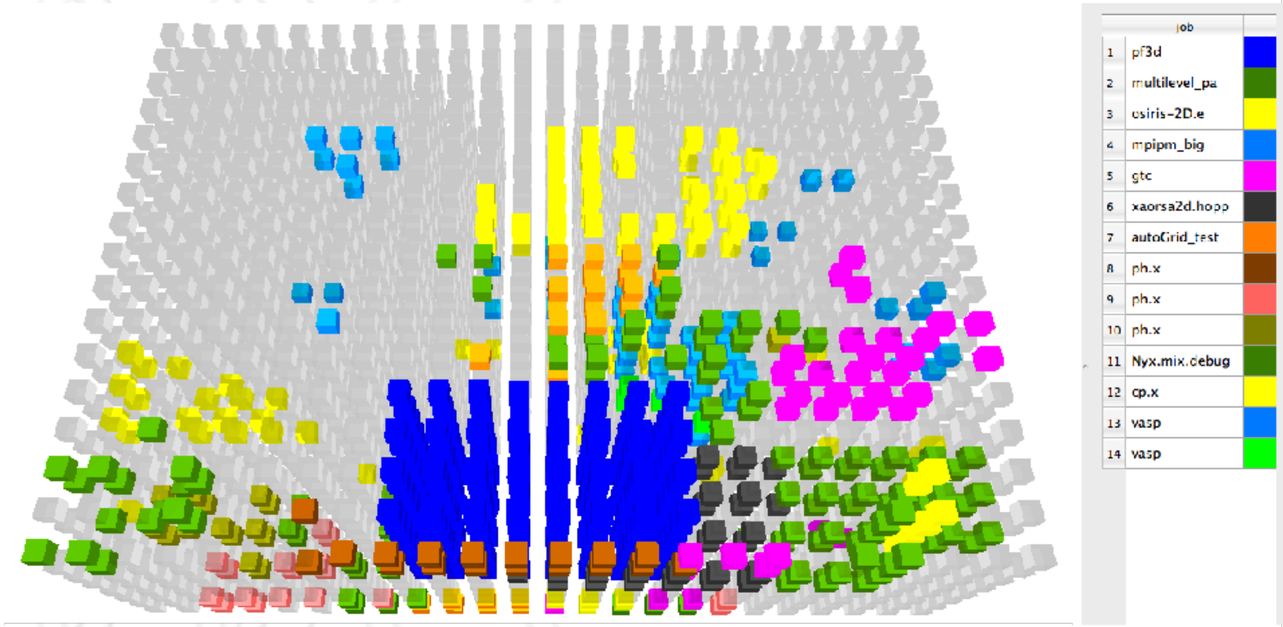
April 16

Impact of other jobs



April 11

MILC job in green



April 16

25% higher messaging rate

UNIVERSITY OF
MARYLAND

Abhinav Bhatele

5218 Brendan Iribe Center (IRB) / College Park, MD 20742

phone: 301.405.4507 / e-mail: bhatele@cs.umd.edu