Task mapping, job placements and routing strategies

Abhinav Bhatele
Center for Applied Scientific Computing

Charm++ Workshop ◆ April 30, 2014

LLNL: Peer-Timo Bremer, Todd Gamblin, Katherine E. Isaacs, Steven H. Langer, Kathryn Mohror, Martin Schulz

Illinois: Ronak Buch, Nikhil Jain, Harshitha Menon, Laxmikant V. Kale, Michael Robson

Utah: Amey Desai, Aaditya G. Landge, Valerio Pascucci

Purdue: Ahmed Abdel-Gawad, Mithuna Thottethodi

LBL: Brian Austin, Nicholas J. Wright

This work was performed under the auspices of the U.S. Department of Energy by Lawrence Livermore National Laboratory under Contract DE-AC52-07NA27344.

Lawrence Livermore National Laboratory, P. O. Box 808, Livermore, CA 94551

Communication: the bottleneck at extreme scale

	Time (ns)	Energy spent (pJ)
Floating point operation	< 0.25	30-45
Time to access DRAM	50	128
Get data from another node	> 1000	128-576

P. Kogge et al., Exascale computing study: Technology challenges in achieving exascale systems, *Technical Report*, 2008.

Communication: the bottleneck at extreme scale

- High costs for data movement in terms of time and energy

	Time (ns)	Energy spent (pJ)
Floating point operation	< 0.25	30-45
Time to access DRAM	50	128
Get data from another node	> 1000	128-576

P. Kogge et al., Exascale computing study: Technology challenges in achieving exascale systems, *Technical Report*, 2008.

Communication: the bottleneck at extreme scale

- High costs for data movement in terms of time and energy
- Newer platforms stressing communication further (more cores, bigger networks)

	Time (ns)	Energy spent (pJ)
Floating point operation	< 0.25	30-45
Time to access DRAM	50	128
Get data from another node	> 1000	128-576

P. Kogge et al., Exascale computing study: Technology challenges in achieving exascale systems, *Technical Report*, 2008.

IBM		Cray	
Blue Gene/L	0.375	XT3	8.77
Blue Gene/P	0.375	XT4	1.36
Blue Gene/Q	0.117	XT5	0.23

Network bytes to flop ratios

A. Bhatele et al., Automated mapping of regular communication graphs on mesh interconnects, *Intl. Conf. on High Performance Computing (HiPC)*, 2010.

Communication: the bottleneck at extreme scale

- High costs for data movement in terms of time and energy
- Newer platforms stressing communication further (more cores, bigger networks)
- Imperative to minimize data movement and maximize locality

	Time (ns)	Energy spent (pJ)
Floating point operation	< 0.25	30-45
Time to access DRAM	50	128
Get data from another node	> 1000	128-576

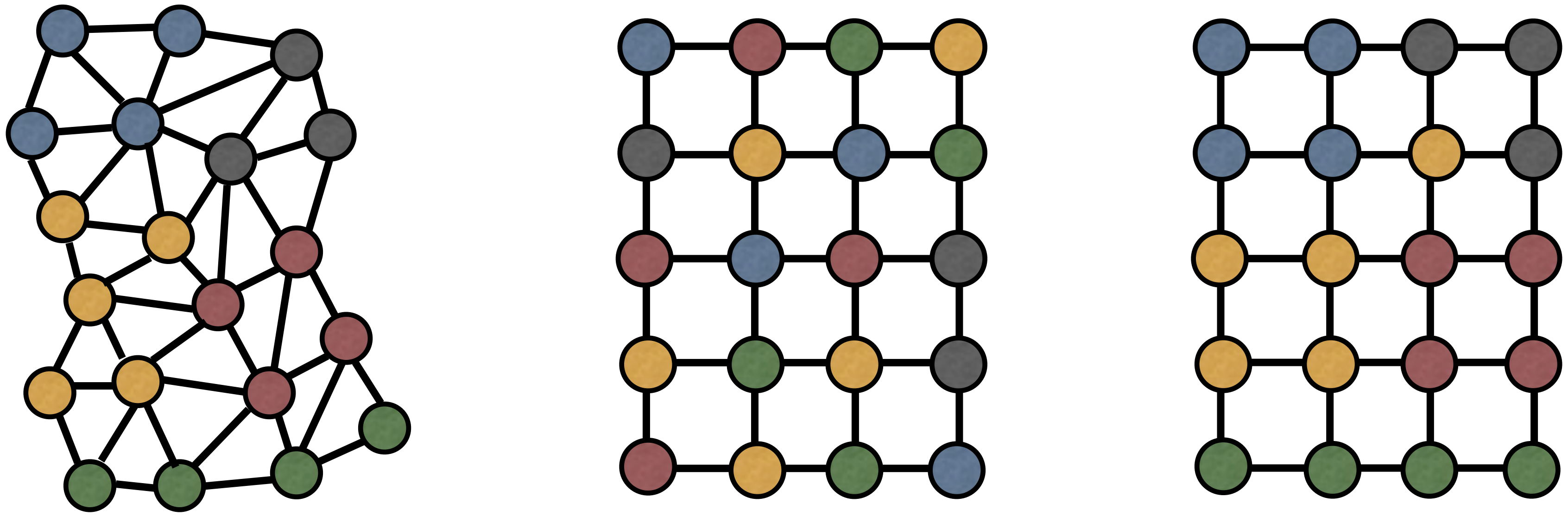
P. Kogge et al., Exascale computing study: Technology challenges in achieving exascale systems, *Technical Report*, 2008.

IBM		Cray	
Blue Gene/L	0.375	XT3	8.77
Blue Gene/P	0.375	XT4	1.36
Blue Gene/Q	0.117	XT5	0.23

Network bytes to flop ratios

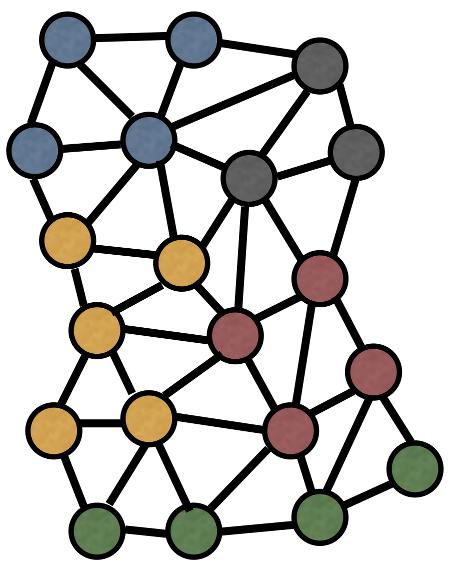
A. Bhatele et al., Automated mapping of regular communication graphs on mesh interconnects, *Intl. Conf. on High Performance Computing (HiPC)*, 2010.

TASK MAPPING



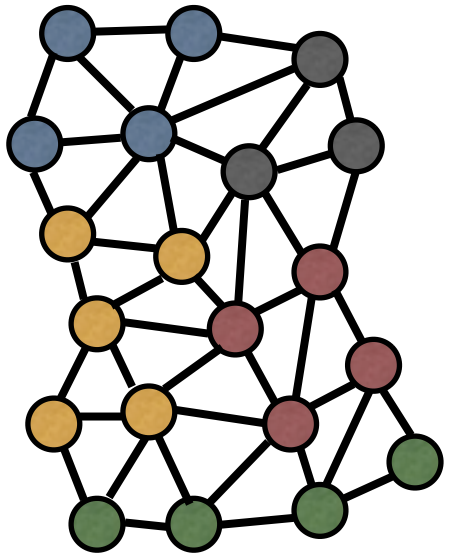
Topology aware task mapping

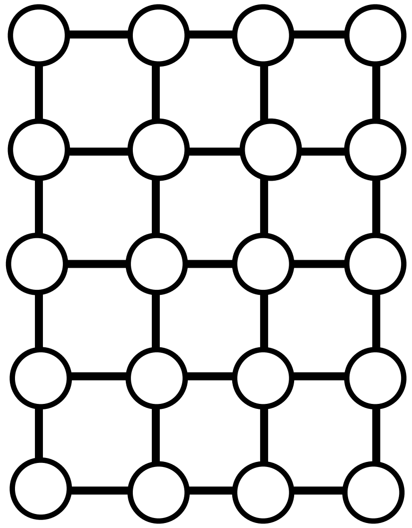
- What is mapping - layout/placement of tasks/processes in an application on the physical interconnect
- Does not require any changes to the application



Topology aware task mapping

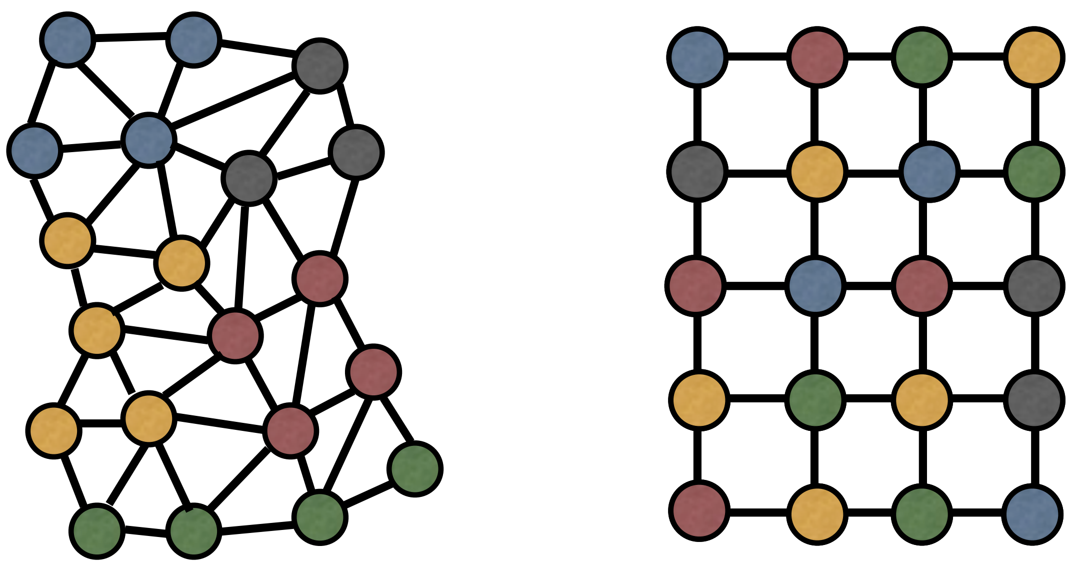
- What is mapping - layout/placement of tasks/processes in an application on the physical interconnect
- Does not require any changes to the application





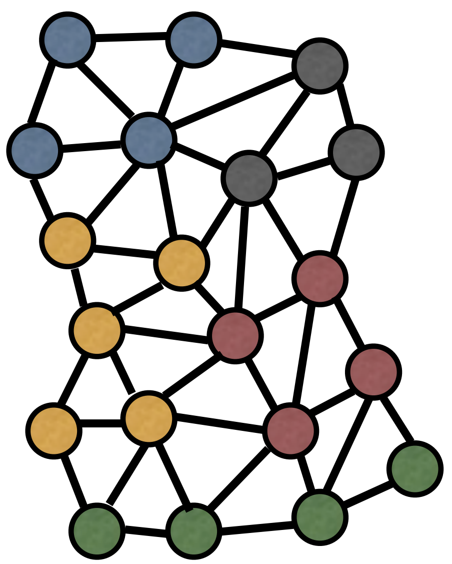
Topology aware task mapping

- What is mapping - layout/placement of tasks/processes in an application on the physical interconnect
- Does not require any changes to the application



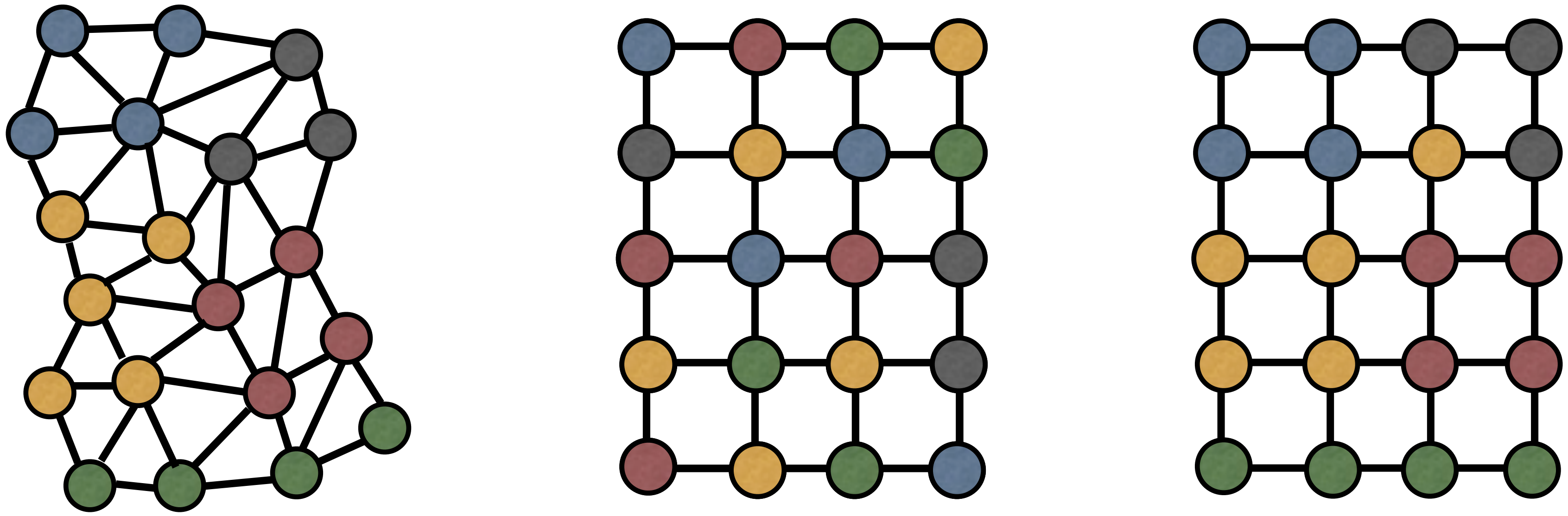
Topology aware task mapping

- What is mapping - layout/placement of tasks/processes in an application on the physical interconnect
- Does not require any changes to the application



Topology aware task mapping

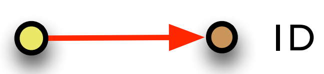
- What is mapping - layout/placement of tasks/processes in an application on the physical interconnect
- Does not require any changes to the application



- Goals:
  - Balance computational load
  - Minimize contention (optimize latency or bandwidth)

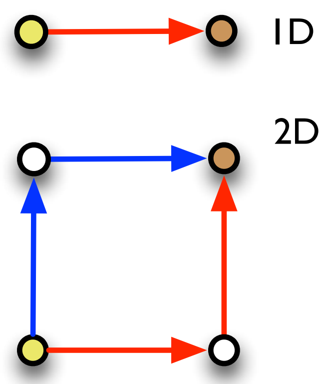
Maximize bandwidth?

- Traditionally, research has focused on bringing tasks closer to reduce the number of hops
  - Minimizes latency, but more importantly link contention
- For applications that send large messages this might not be optimal



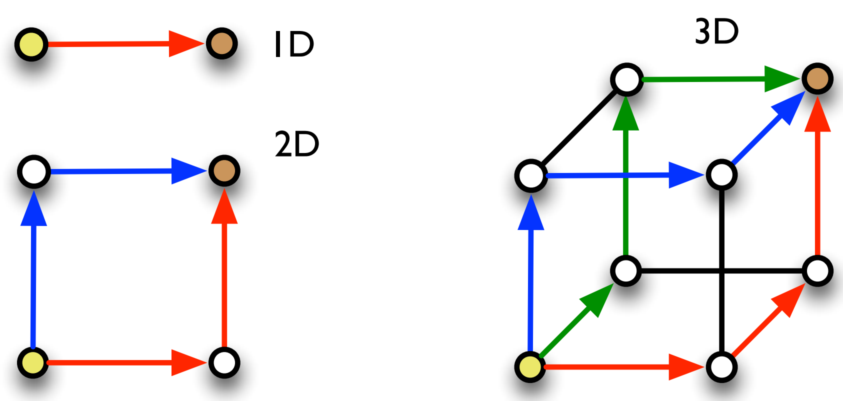
Maximize bandwidth?

- Traditionally, research has focused on bringing tasks closer to reduce the number of hops
  - Minimizes latency, but more importantly link contention
- For applications that send large messages this might not be optimal



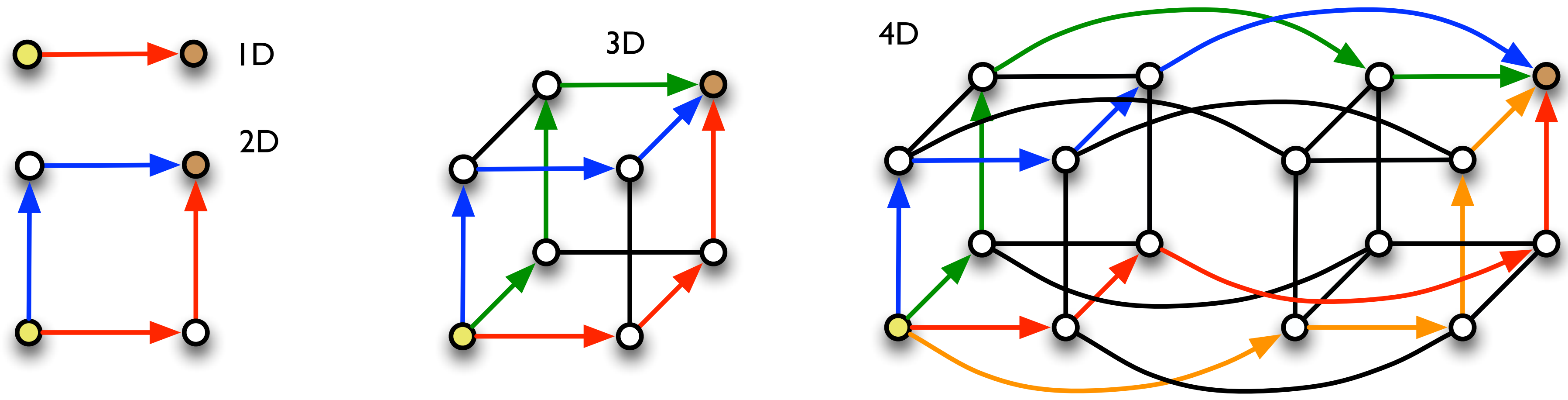
Maximize bandwidth?

- Traditionally, research has focused on bringing tasks closer to reduce the number of hops
  - Minimizes latency, but more importantly link contention
- For applications that send large messages this might not be optimal



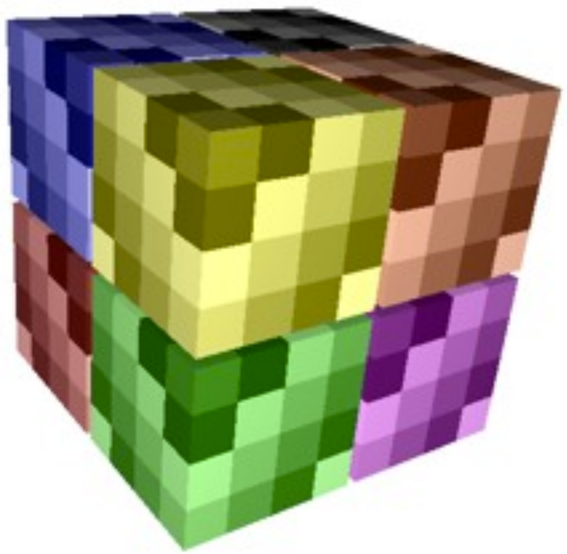
Maximize bandwidth?

- Traditionally, research has focused on bringing tasks closer to reduce the number of hops
  - Minimizes latency, but more importantly link contention
- For applications that send large messages this might not be optimal



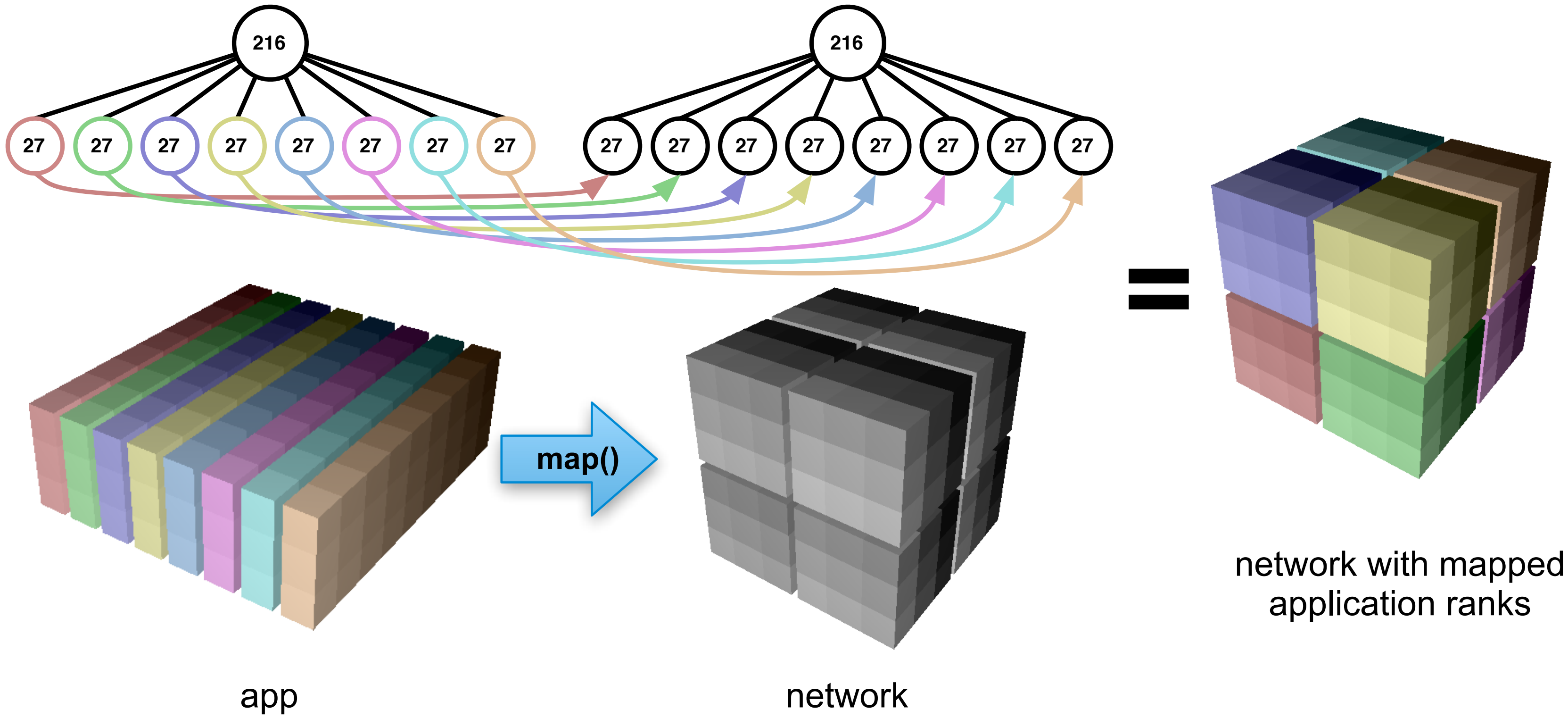
Rubik

- We have developed a mapping tool focusing on:
  - structured applications that are bandwidth-bound, use collectives over sub-communicators
  - built-in operations that can increase effective bandwidth on torus networks based on heuristics
- Input:
  - Application topology with subsets identified
  - Processor topology
  - Set of operations to perform
- Output: map file for job launcher



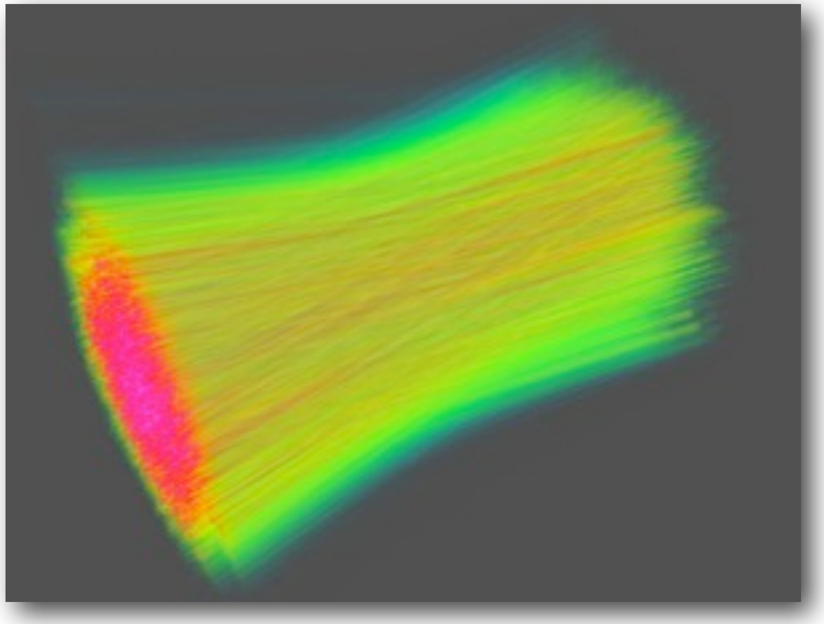
Application example

```
app = box([9,3,8]) # Create app partition tree of 27-task planes
app.tile([9,3,1])

network = box([6,6,6]) # Create network partition tree of 27-processor cubes
network.tile([3,3,3])

network.map(app)  # Map task planes into cubes
```



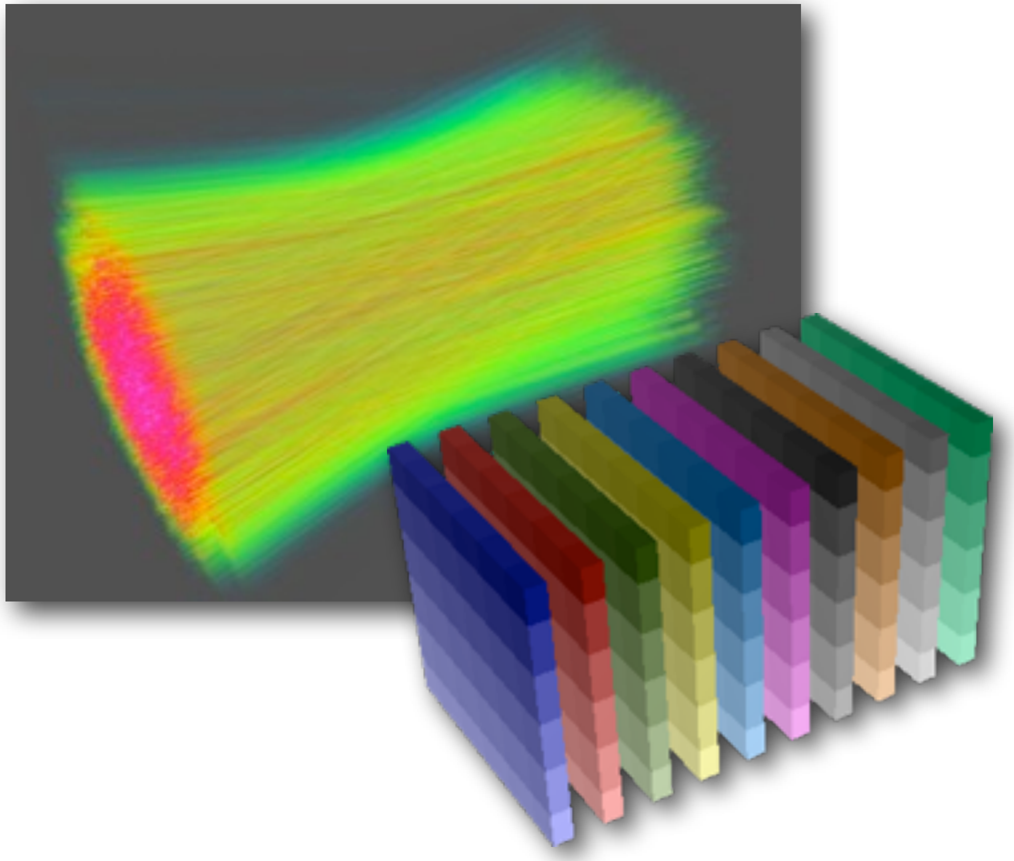
Mapping pF3D

- A laser-plasma interaction code used at the National Ignition Facility (NIF) at LLNL
- Three communication phases over a 3D virtual topology:
  - Wave propagation and coupling: 2D FFTs within XY planes
  - Light advection: Send-recv between consecutive XY planes
  - Hydrodynamic equations: 3D near-neighbor exchange



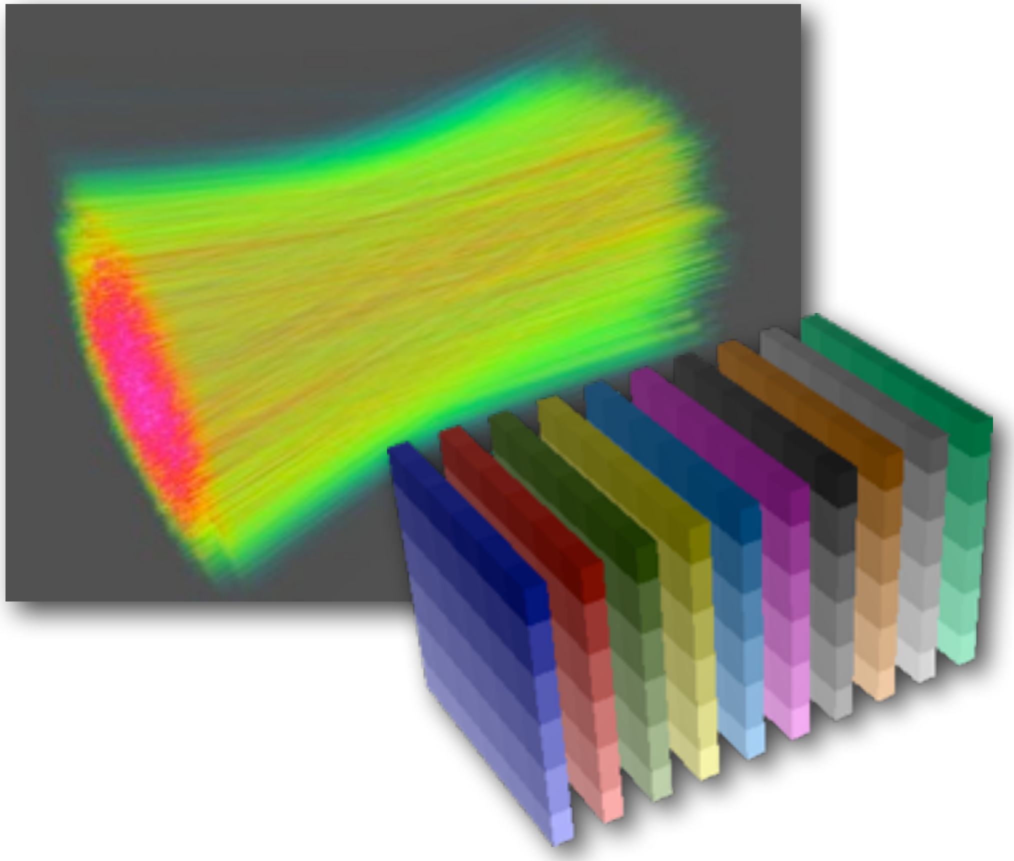
Mapping pF3D

- A laser-plasma interaction code used at the National Ignition Facility (NIF) at LLNL
- Three communication phases over a 3D virtual topology:
  - Wave propagation and coupling: 2D FFTs within XY planes
  - Light advection: Send-recv between consecutive XY planes
  - Hydrodynamic equations: 3D near-neighbor exchange



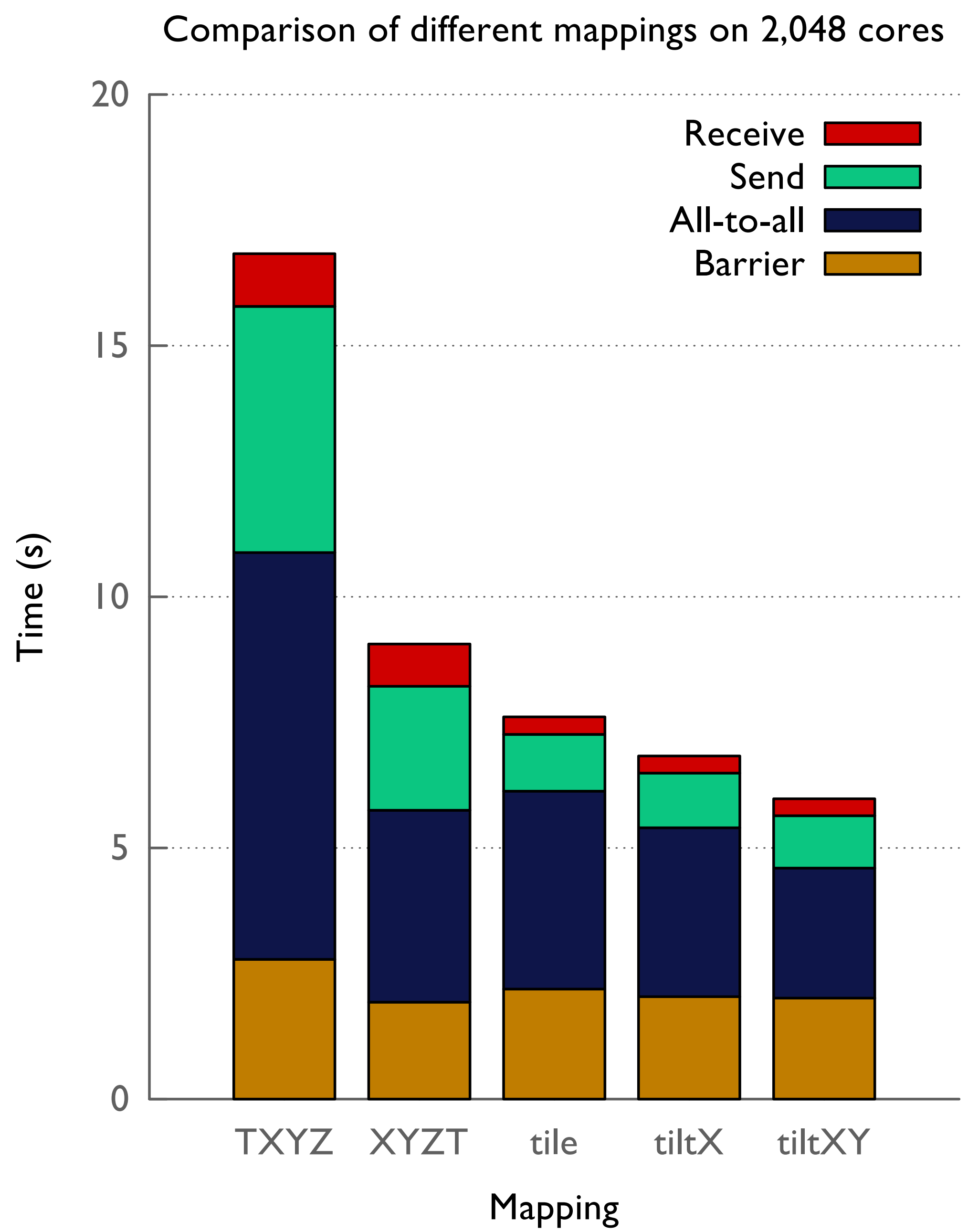
Mapping pF3D

- A laser-plasma interaction code used at the National Ignition Facility (NIF) at LLNL
- Three communication phases over a 3D virtual topology:
  - Wave propagation and coupling: 2D FFTs within XY planes
  - Light advection: Send-recv between consecutive XY planes
  - Hydrodynamic equations: 3D near-neighbor exchange



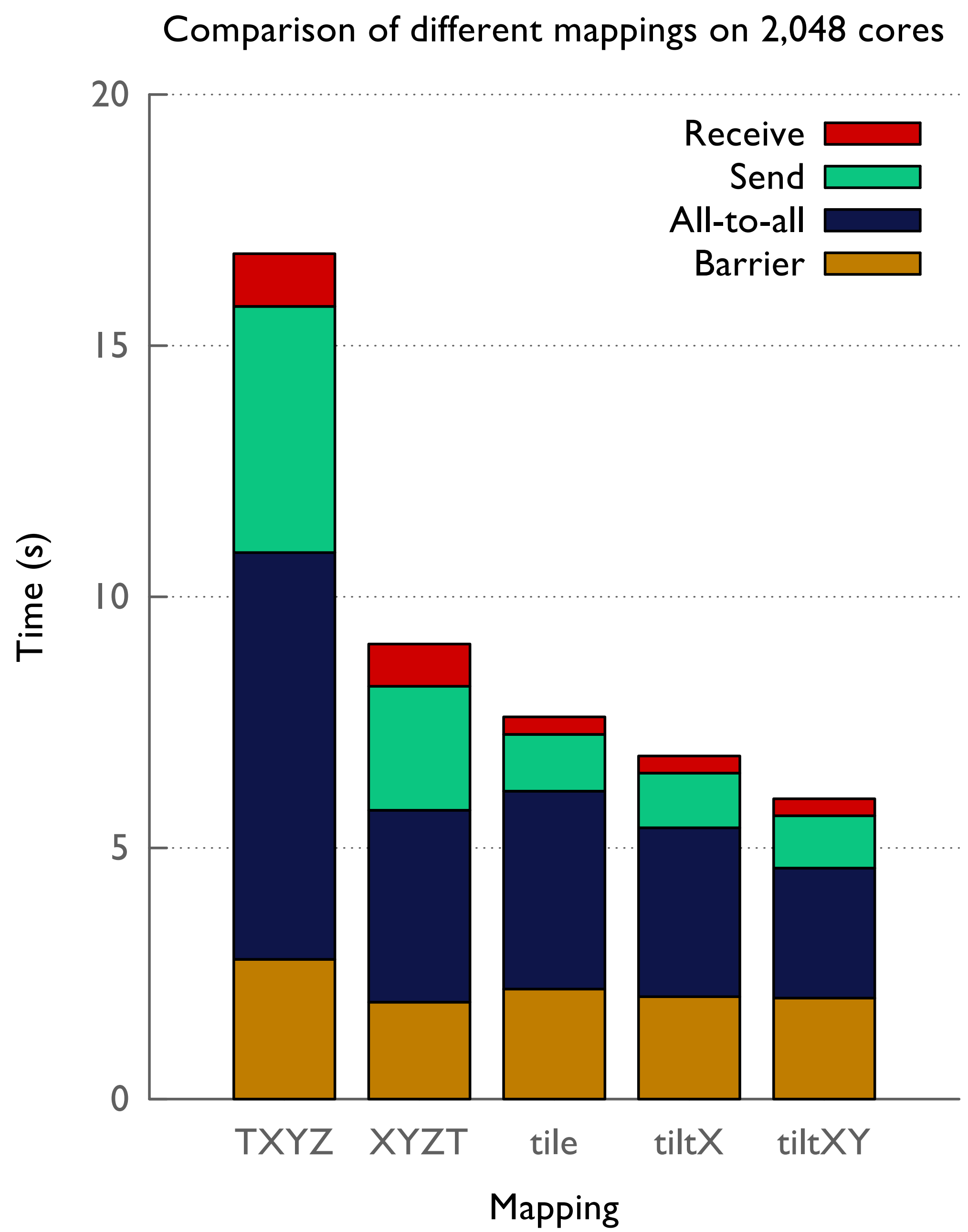
	2048 cores		16384 cores	
MPI call	**Total %**	**MPI %**	**Total %**	**MPI %**
Send	4.90	28.45	23.10	57.21
Alltoall	8.10	46.94	7.30	18.07
Barrier	2.78	16.10	8.13	20.15

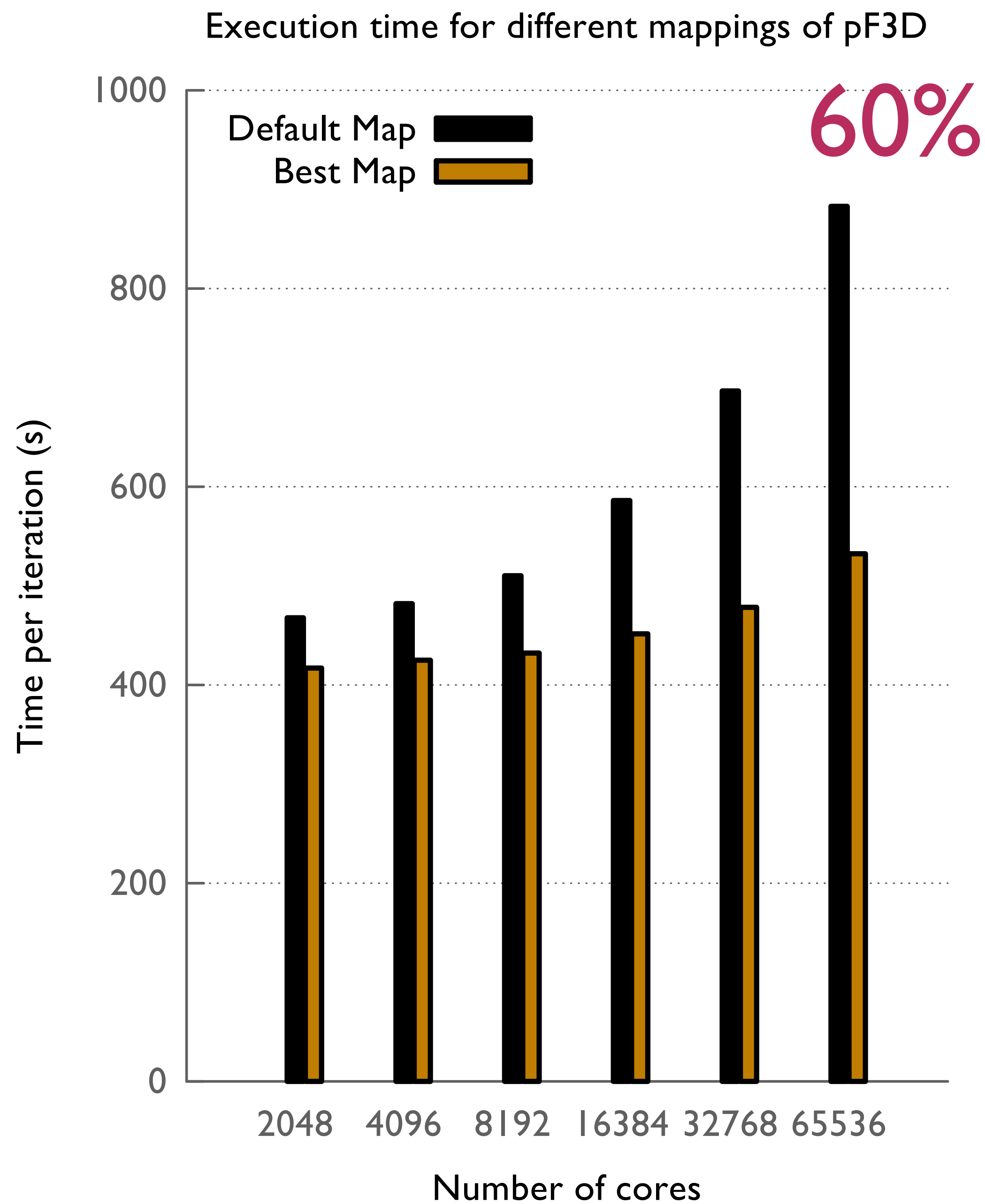
Performance benefits



A. Bhatele et al. Mapping applications with collectives over sub-communicators on torus networks. In *Proceedings of the ACM/IEEE International Conference for High Performance Computing, Networking, Storage and Analysis*, SC '12. IEEE Computer Society, November 2012.

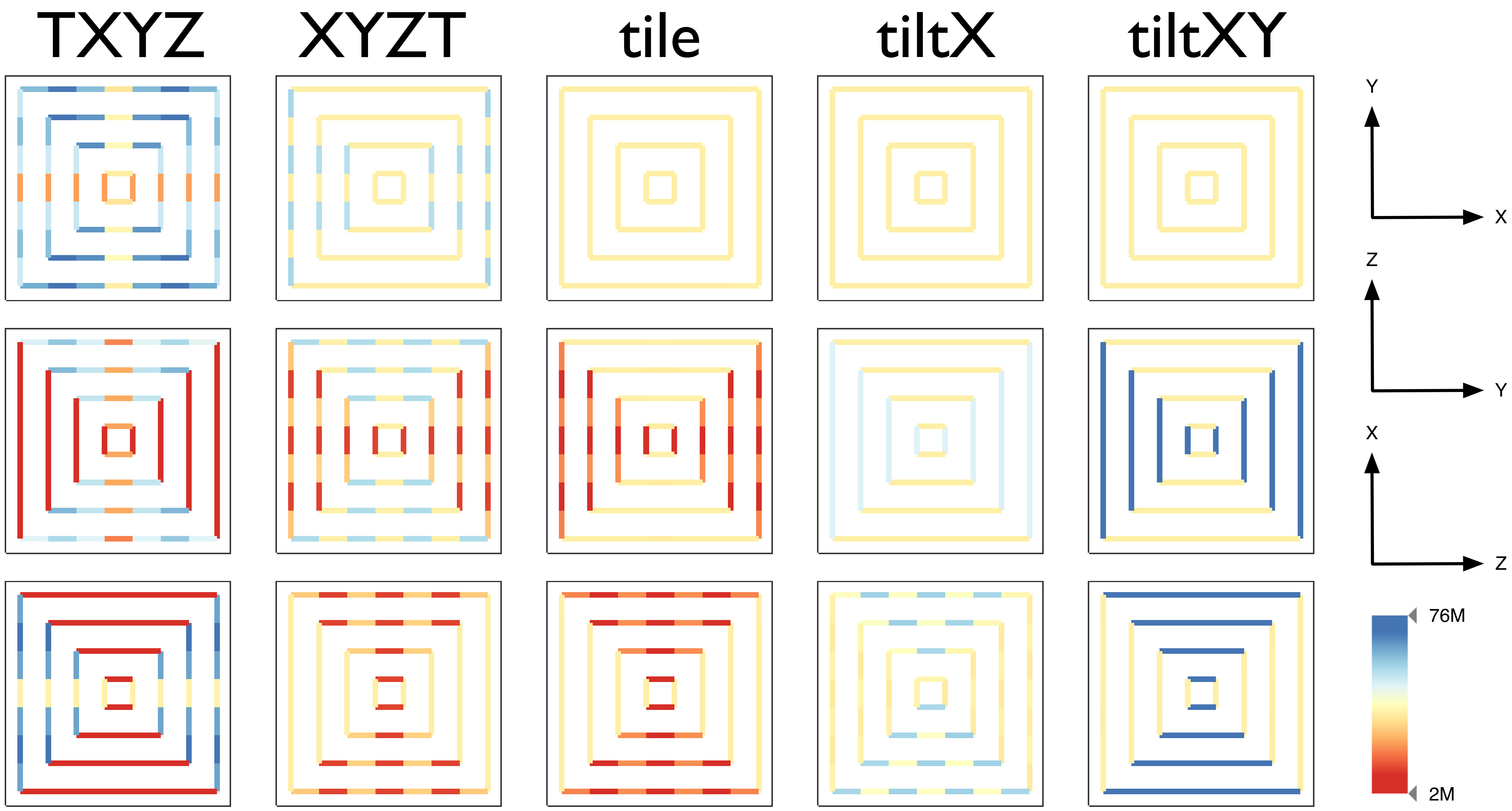
Performance benefits



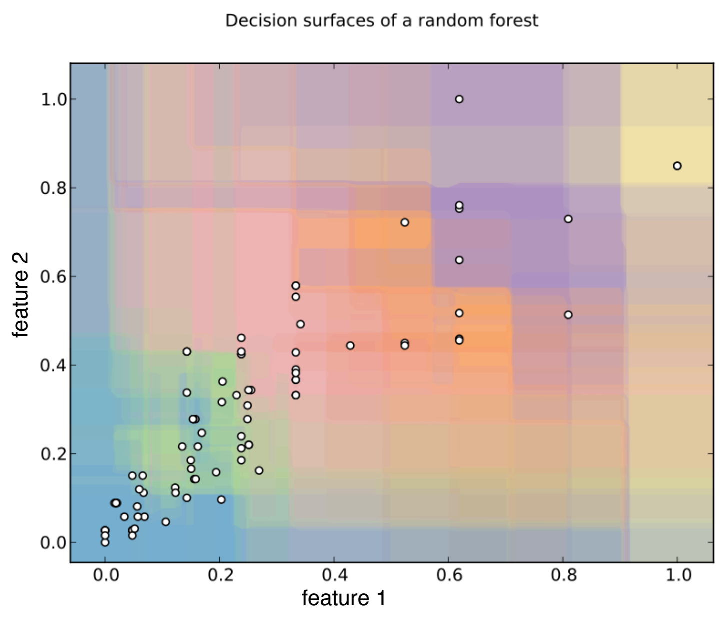


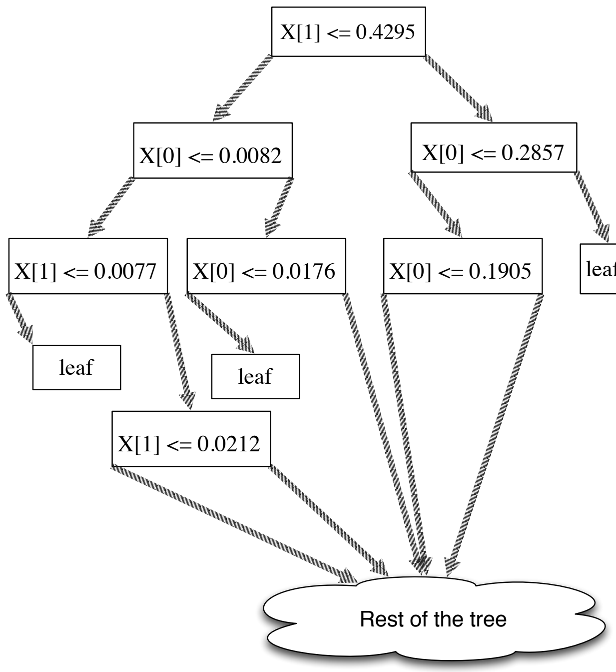
A. Bhatele et al. Mapping applications with collectives over sub-communicators on torus networks. In *Proceedings of the ACM/IEEE International Conference for High Performance Computing, Networking, Storage and Analysis*, SC '12. IEEE Computer Society, November 2012.

Visualizing network traffic using Boxfish



MODELING & SIMULATION





Predicting execution time without executing the code

- Goal: find which mapping gives the best performance
- Offline metrics: maximum hops, average bytes, maximum bytes
- Use network hardware counters to propose new metrics
- Supervised learning algorithms to predict performance

N. Jain et al. Predicting application performance using supervised learning on communication features. In *Proceedings of the ACM/IEEE International Conference for High Performance Computing, Networking, Storage and Analysis*, SC '13. IEEE Computer Society, November 2013.

Why don't we run all the mappings?

Why don't we run all the mappings?

- Wasted allocation hours

	2012	2013
Intrepid	4.16M	0.73M
Mira	0.17M	7.67M
Total	4.33M	8.40M

Why don't we run all the mappings?

- Wasted allocation hours

	2012	2013
Intrepid	4.16M	0.73M
Mira	0.17M	7.67M
Total	4.33M	8.40M

13 million core hours!

Why don't we run all the mappings?

- Wasted allocation hours
- Wasted time in the queue

	2012	2013
Intrepid	4.16M	0.73M
Mira	0.17M	7.67M
Total	4.33M	8.40M

13 million core hours!

Why don't we run all the mappings?

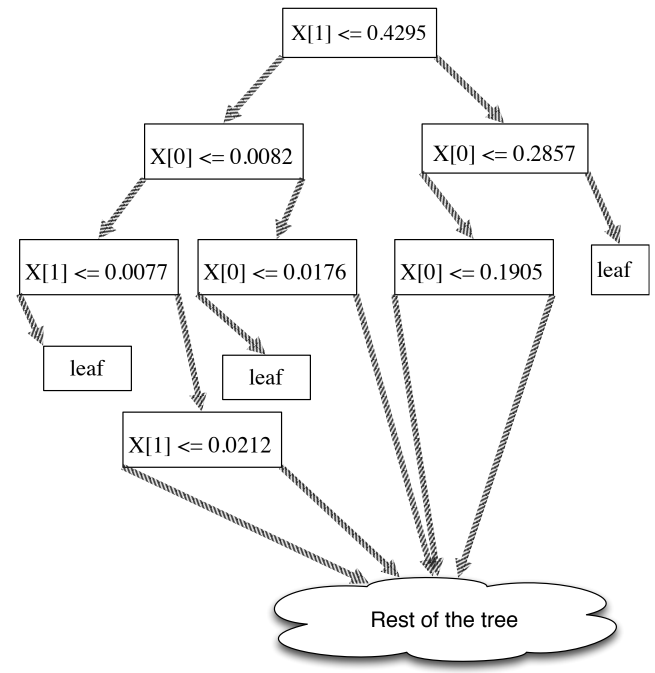
- Wasted allocation hours
- Wasted time in the queue
- All we need is - which is the best mapping?

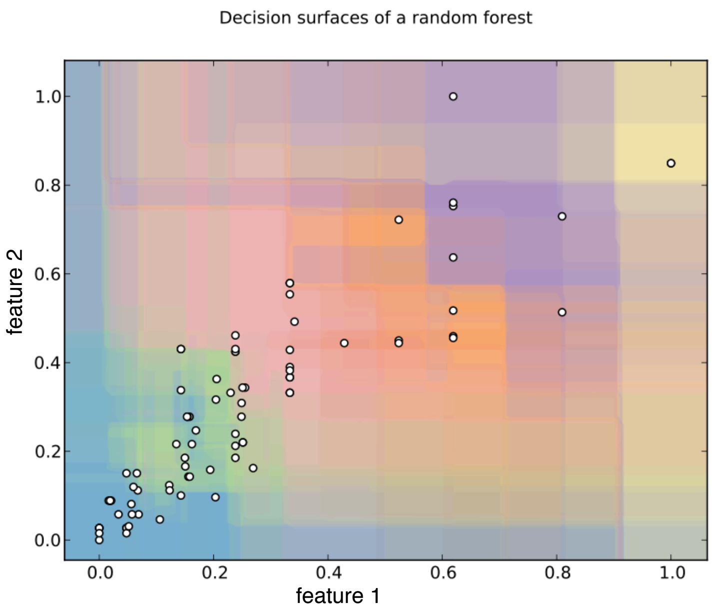
	2012	2013
Intrepid	4.16M	0.73M
Mira	0.17M	7.67M
Total	4.33M	8.40M

13 million core hours!

Supervised learning: scikit-learn

- Use simulation and other tools to obtain network counters and other contention parameters
- Exploit supervised learning algorithms for performance prediction
  - forests of randomized decision trees

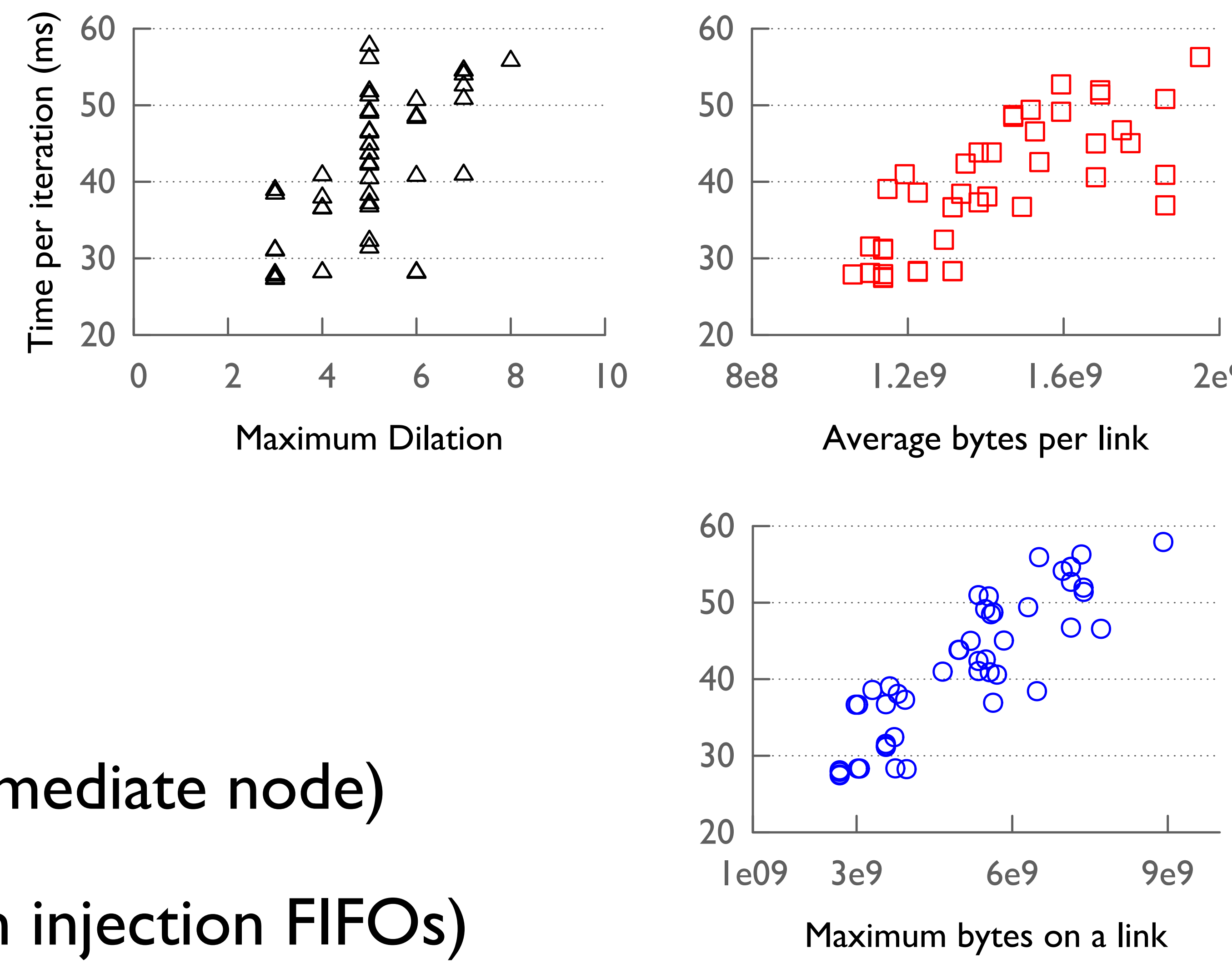




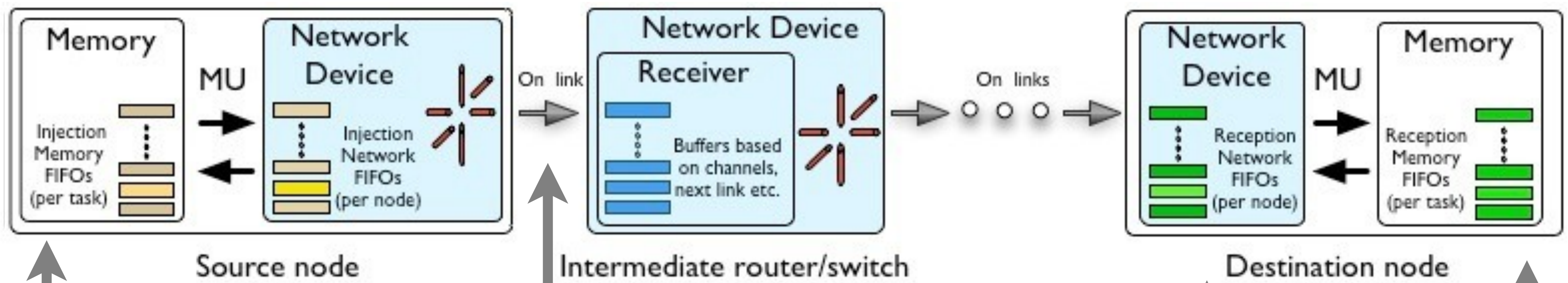
scikit learn http://scikit-learn.org

Existing and new metrics

- Existing metrics
  - maximum hops
  - average bytes
  - maximum bytes
- New metrics:
  - Buffer length (on intermediate node)
  - FIFO length (packets in injection FIFOs)
  - Delay per link (packets in buffers / #received packets)



Message life cycle on Blue Gene/Q



Injection FIFO Contention

Receive Buffer Contention

Reception FIFO Contention

Memory Contention

Link Contention

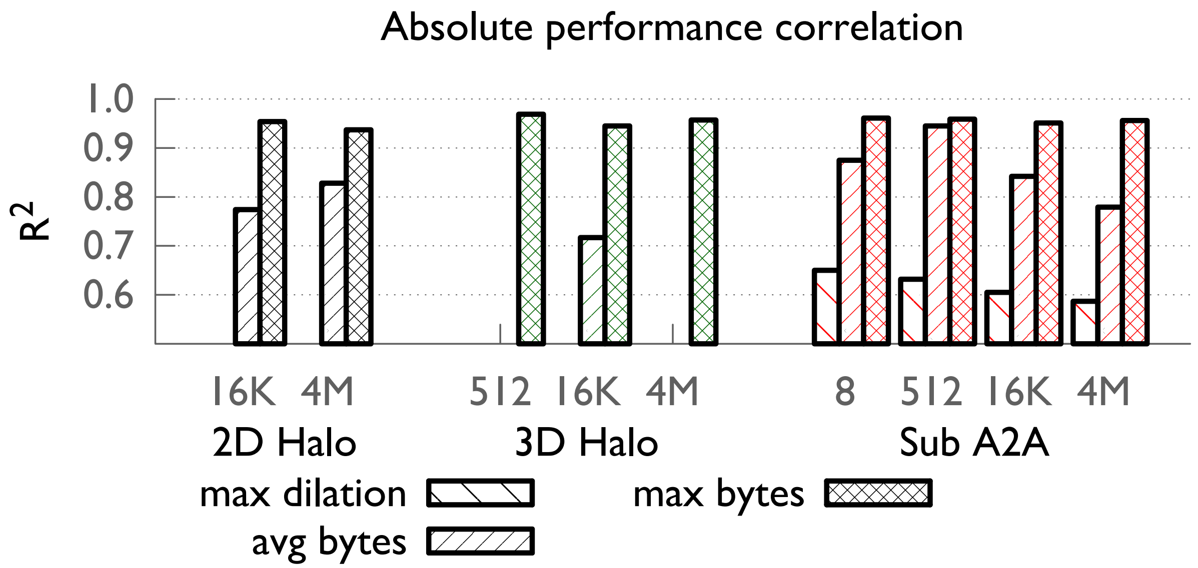
Memory Contention

Results

- Three communication kernels
  - Five-point 2D Stencil
  - 14-point 3D Stencil
  - All-to-all over sub-communicators

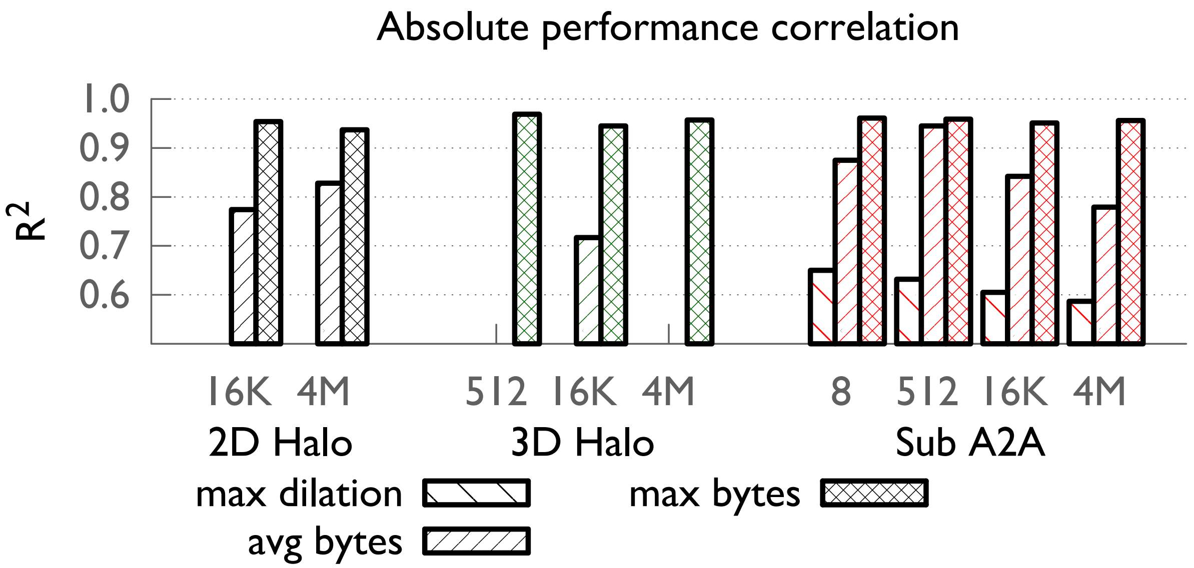
Results

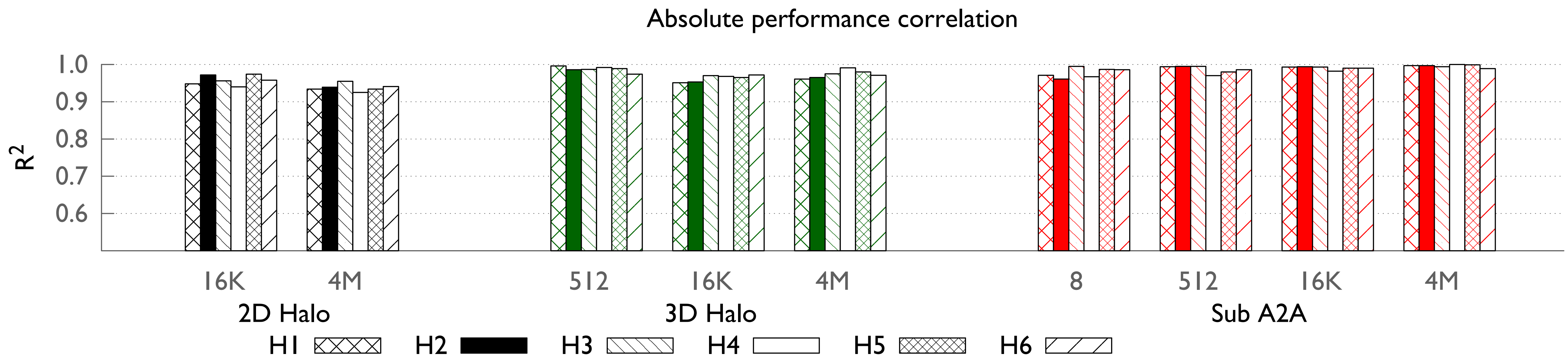
- Three communication kernels
  - Five-point 2D Stencil
  - 14-point 3D Stencil
  - All-to-all over sub-communicators



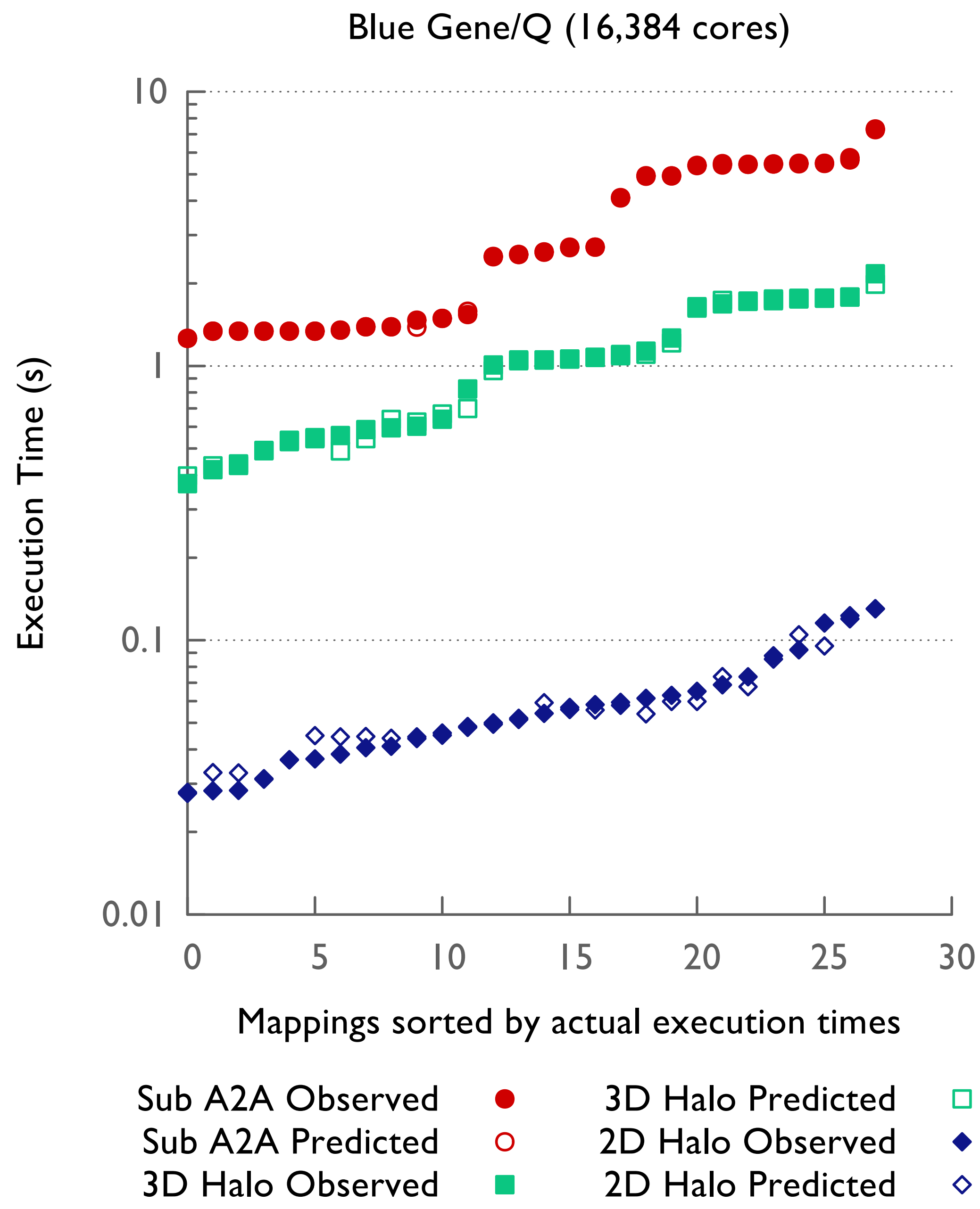
Results

- Three communication kernels
  - Five-point 2D Stencil
  - 14-point 3D Stencil
  - All-to-all over sub-communicators



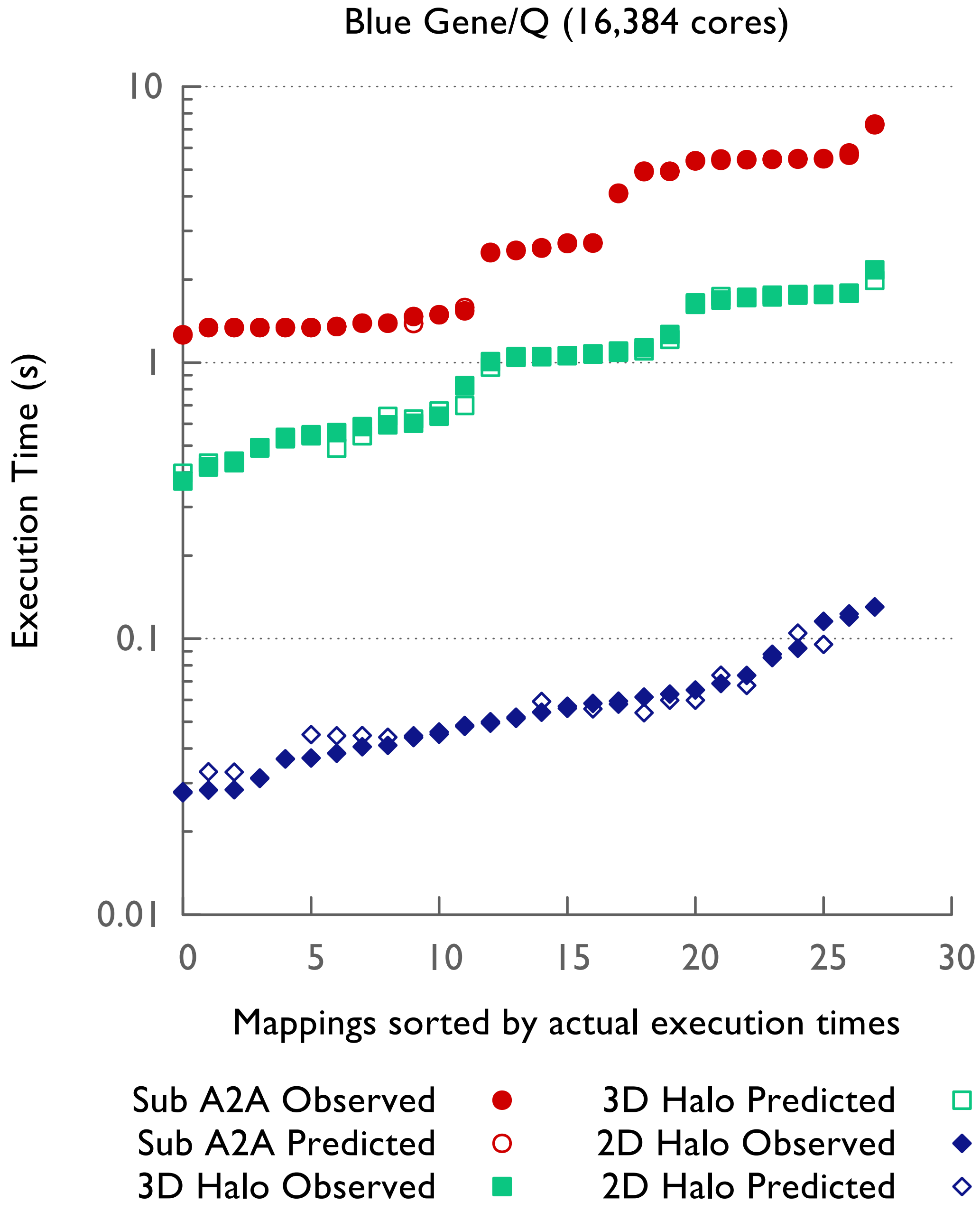


Performance prediction for communication kernels



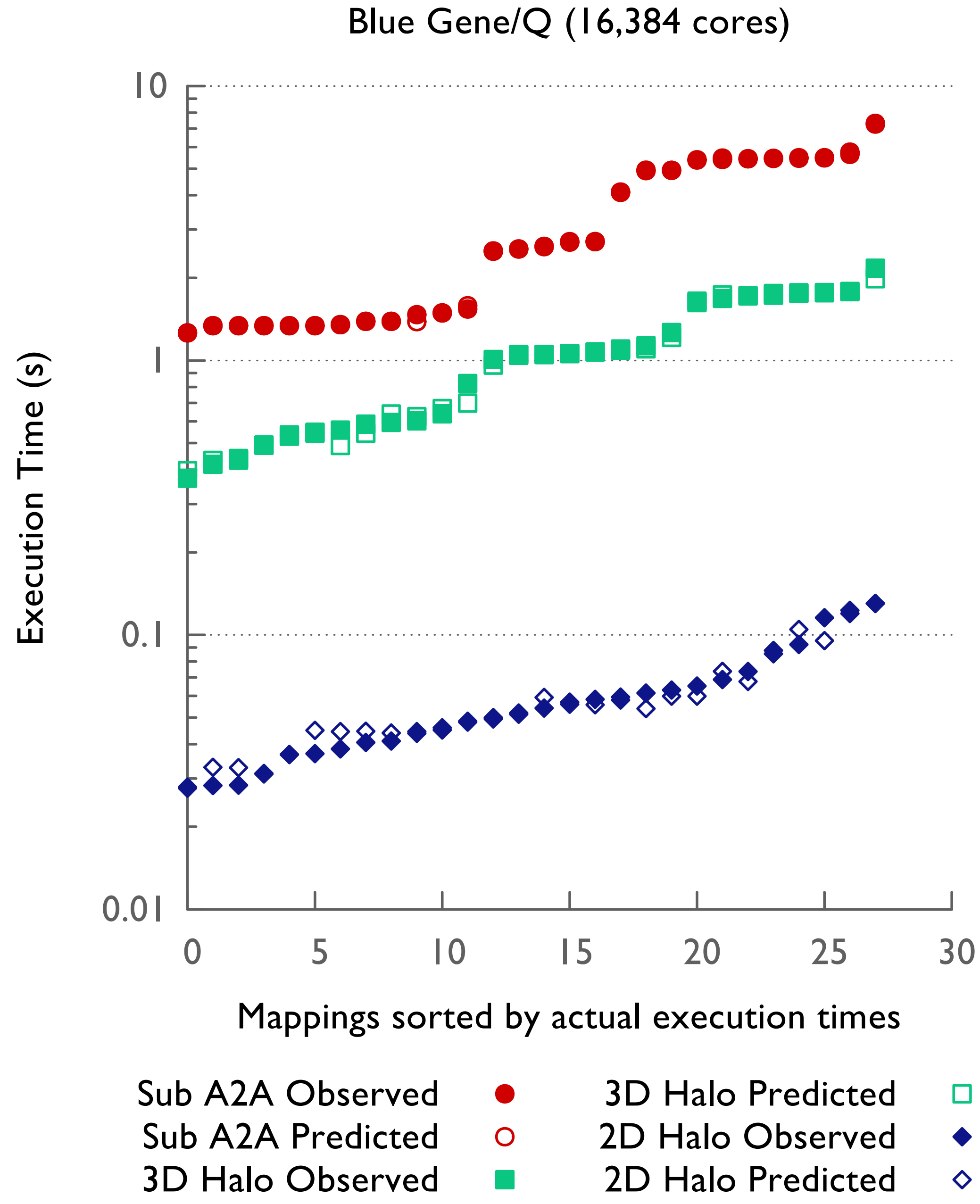
Performance prediction for communication kernels

- Better correlation than with existing metrics such as average or maximum bytes



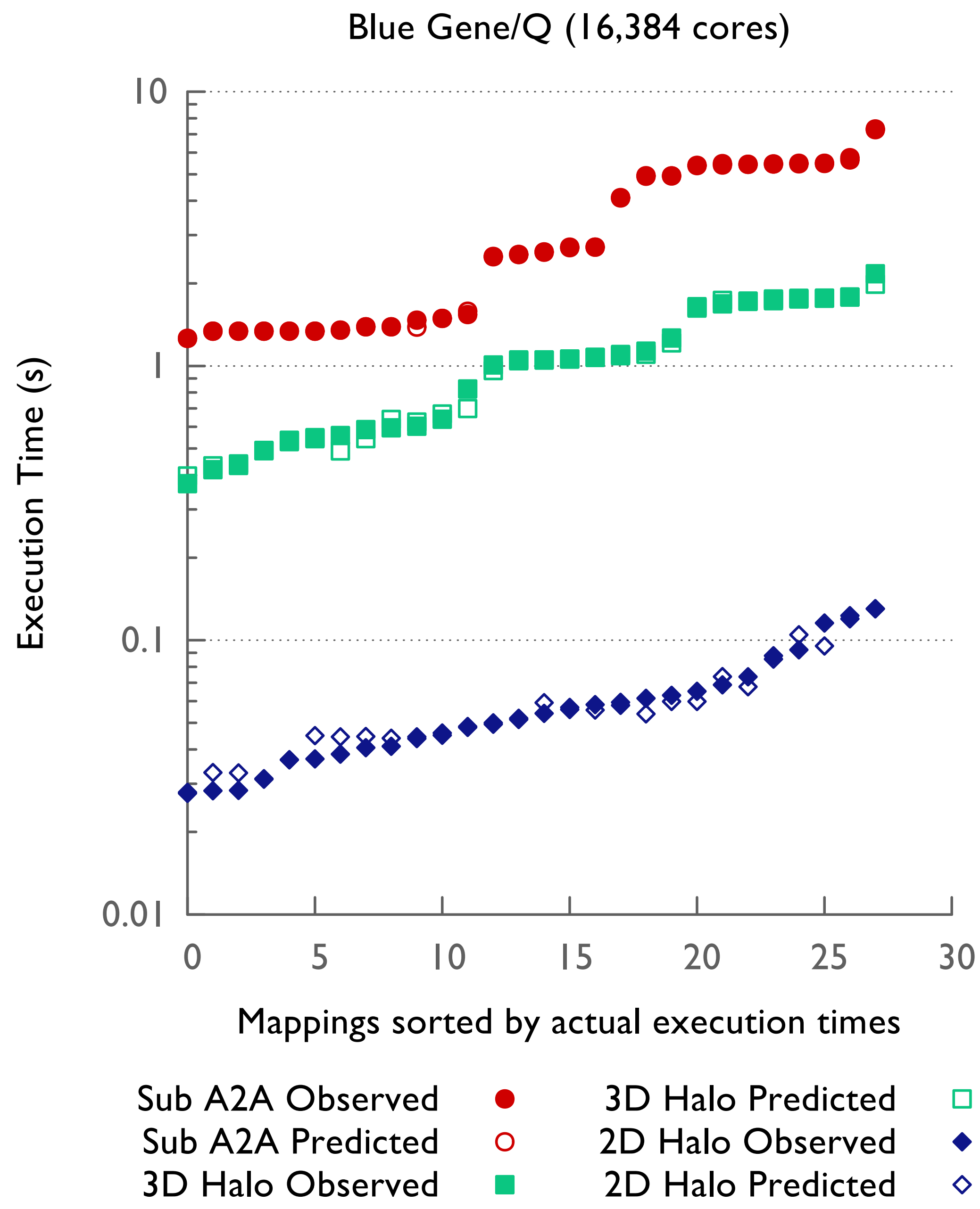
Performance prediction for communication kernels

- Better correlation than with existing metrics such as average or maximum bytes
- Hybrid metric:
  - average bytes + maximum bytes + average buffer length + maximum FIFO length



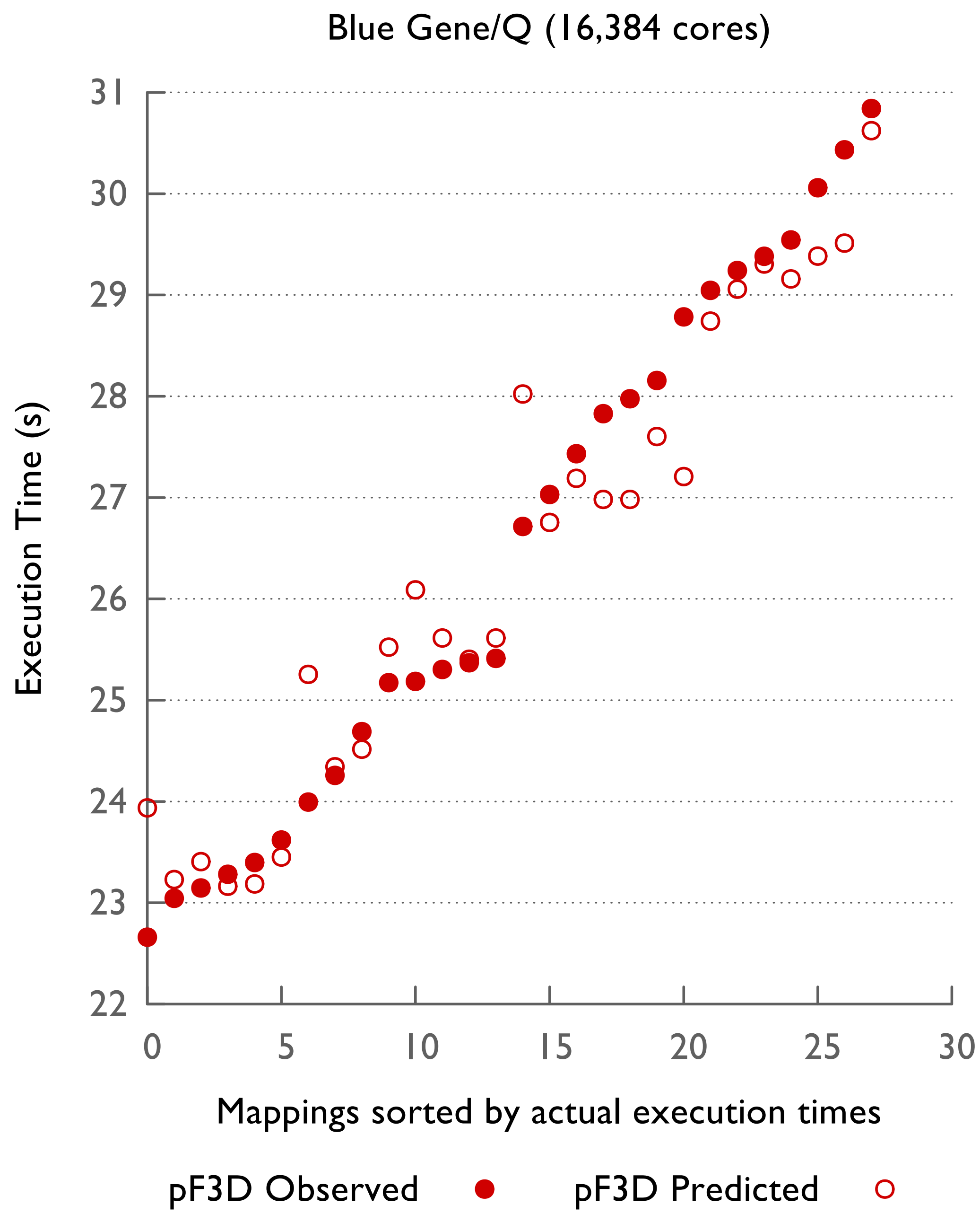
Performance prediction for communication kernels

- Better correlation than with existing metrics such as average or maximum bytes
- Hybrid metric:
  - average bytes + maximum bytes + average buffer length + maximum FIFO length
- Crazy things:
  - combine all training sets
  - use 16k training set to predict 64k performance

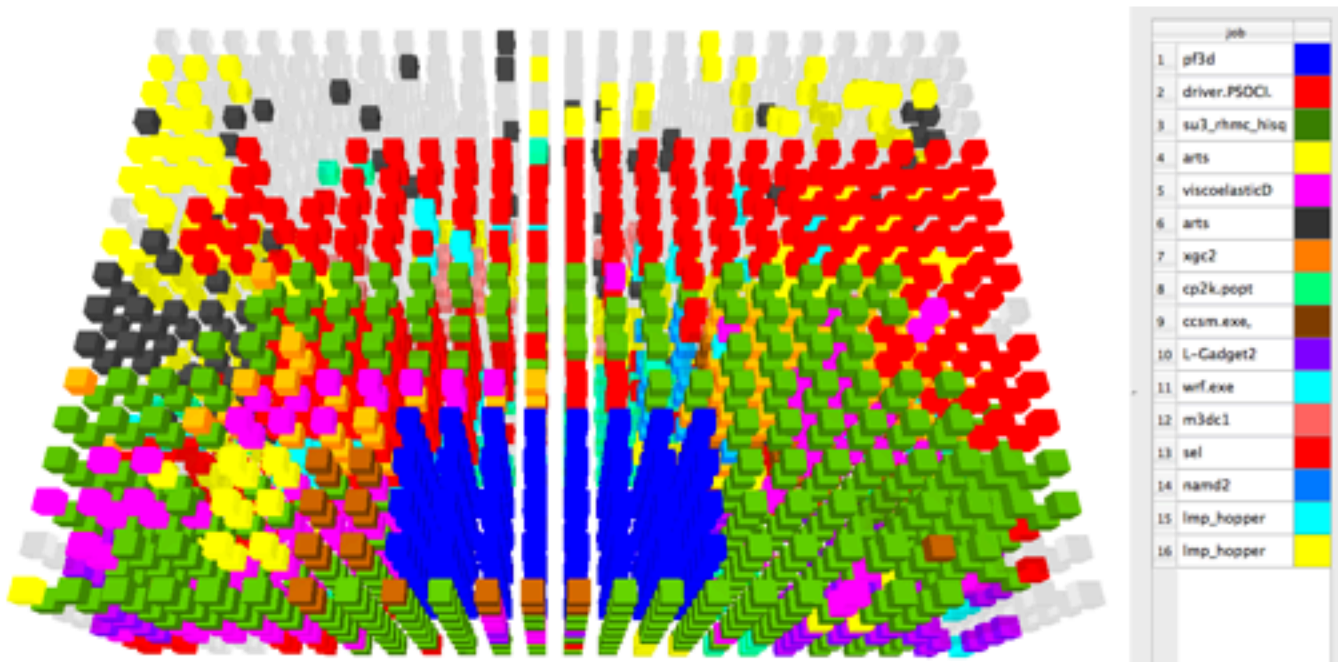


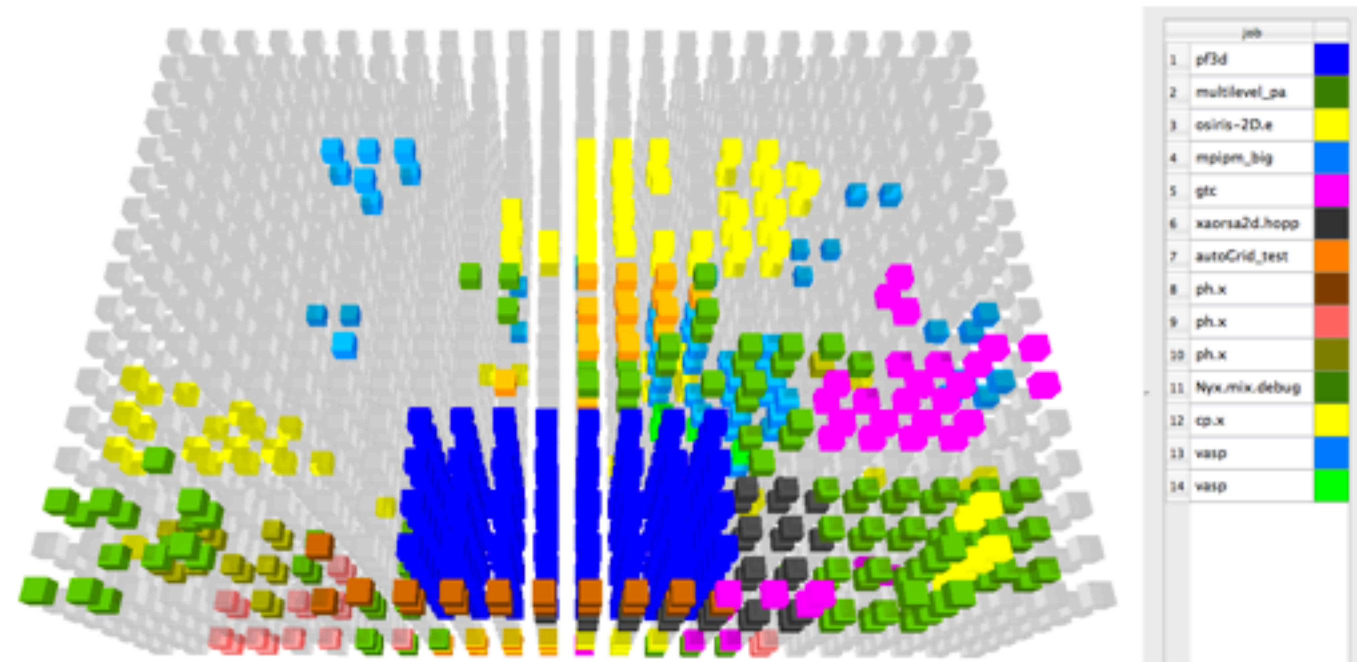
Predicting the performance of pF3D

- Production application
  - has computation
  - and multiple phases of communication
- Hybrid metric:
  - average bytes + average buffer length + average delay + sum of hops + maximum FIFO length



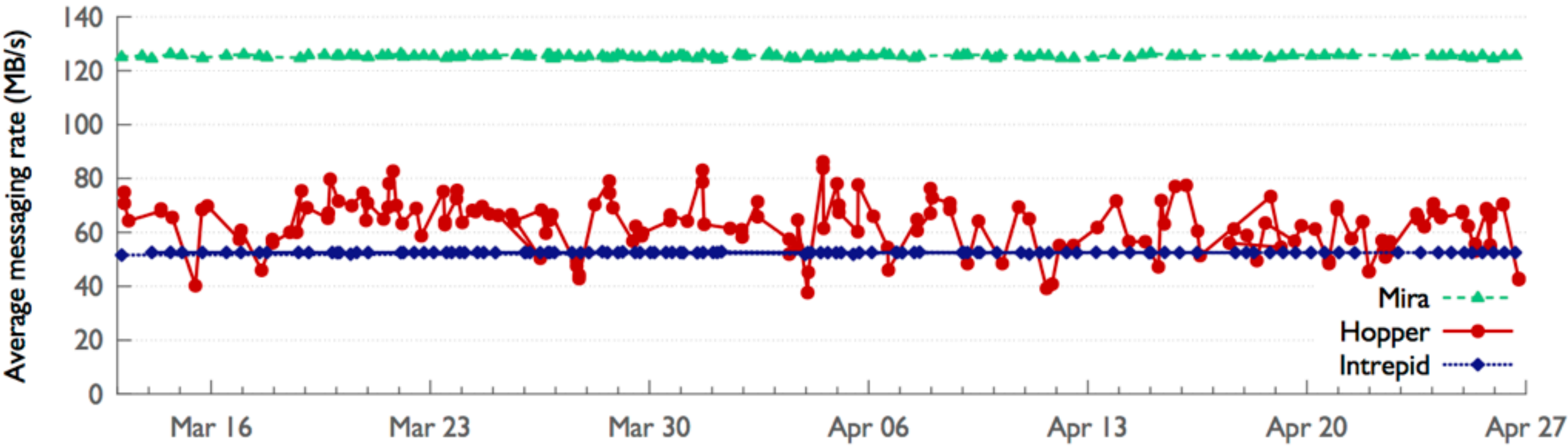
JOB PLACEMENT & ROUTING





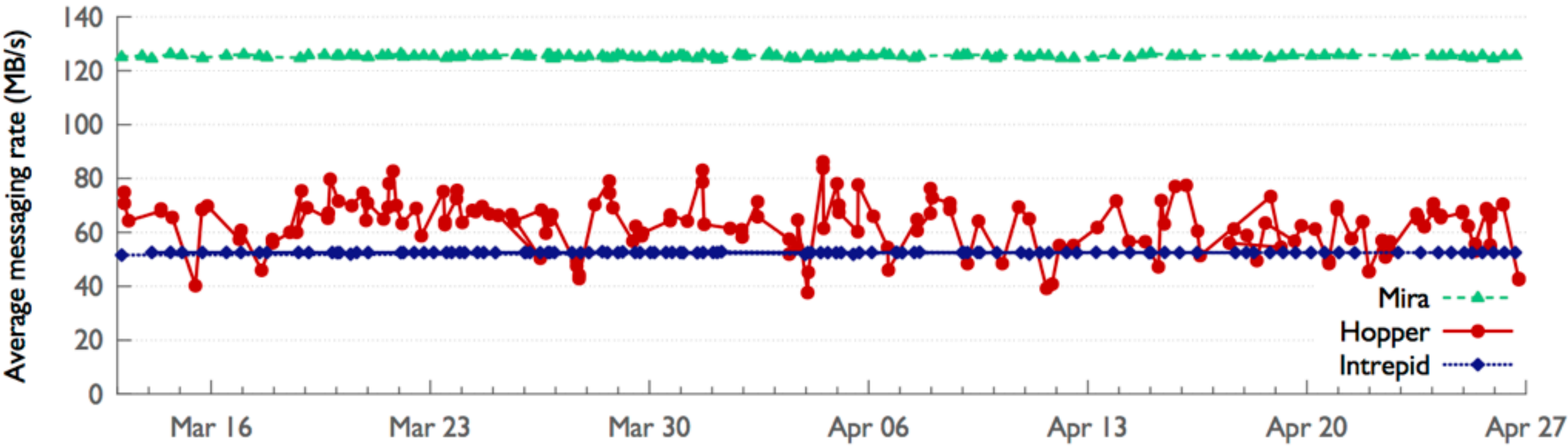
Performance variability

Average messaging rates for batch jobs running a laser-plasma interaction code



Performance variability

Average messaging rates for batch jobs running a laser-plasma interaction code



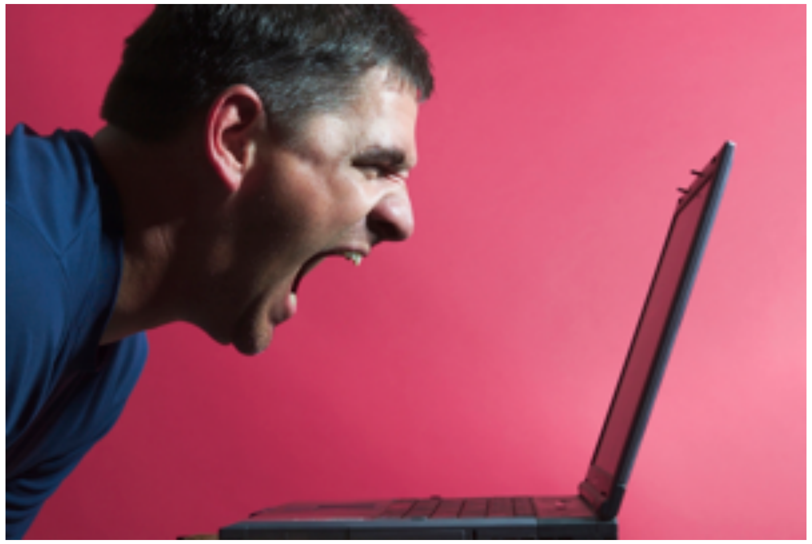
$$\frac{\text{Total number of bytes sent on the network}}{\text{Time spent sending the messages}}$$

Leads to several problems ...

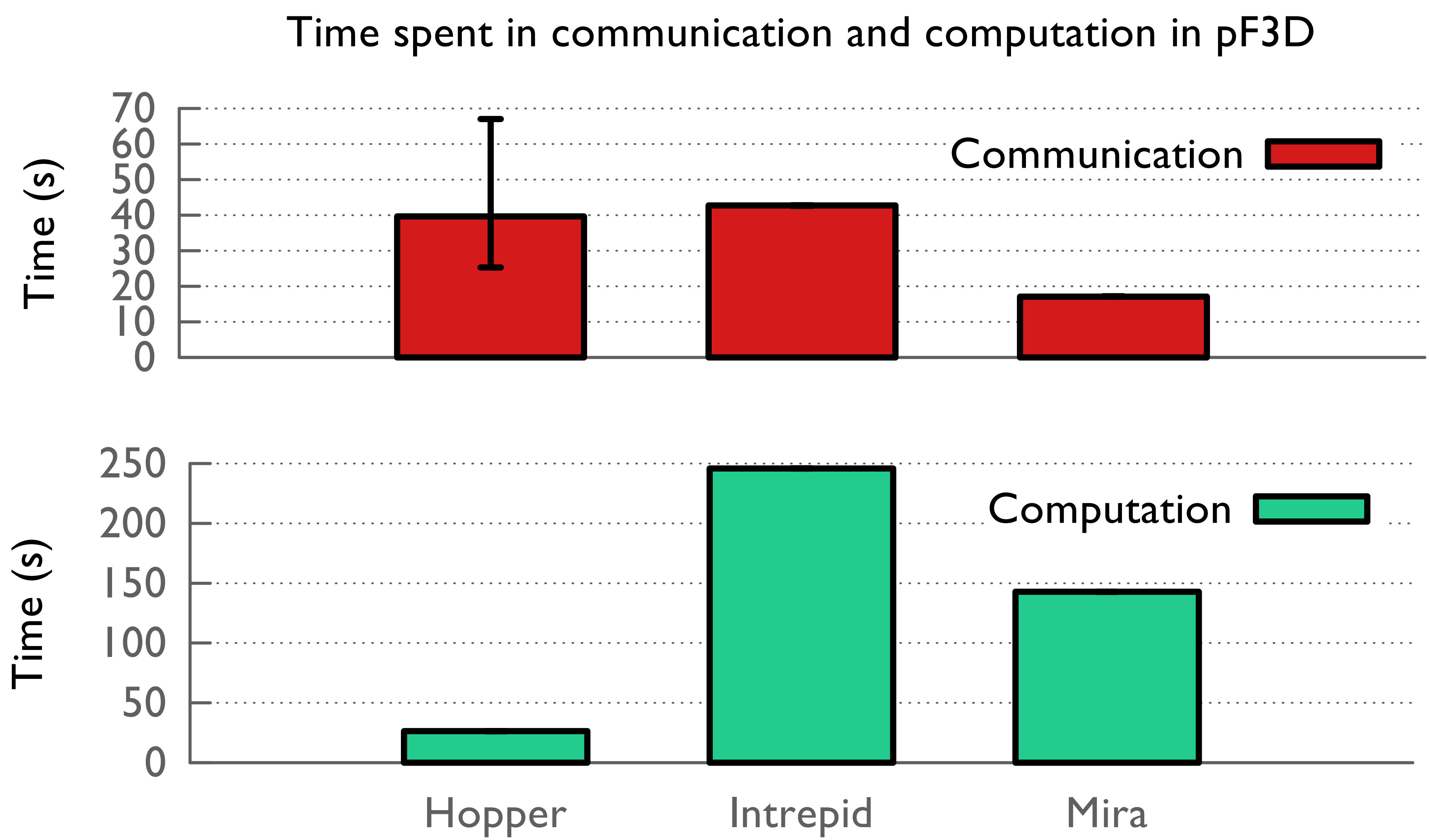
- Individual jobs run slower:
  - More time to complete science simulations
  - Increased wait time in job queues
  - Inefficient use of machine time allocation/core-hours
- Overall lower throughput
- Increase energy usage/costs

Also affects software development

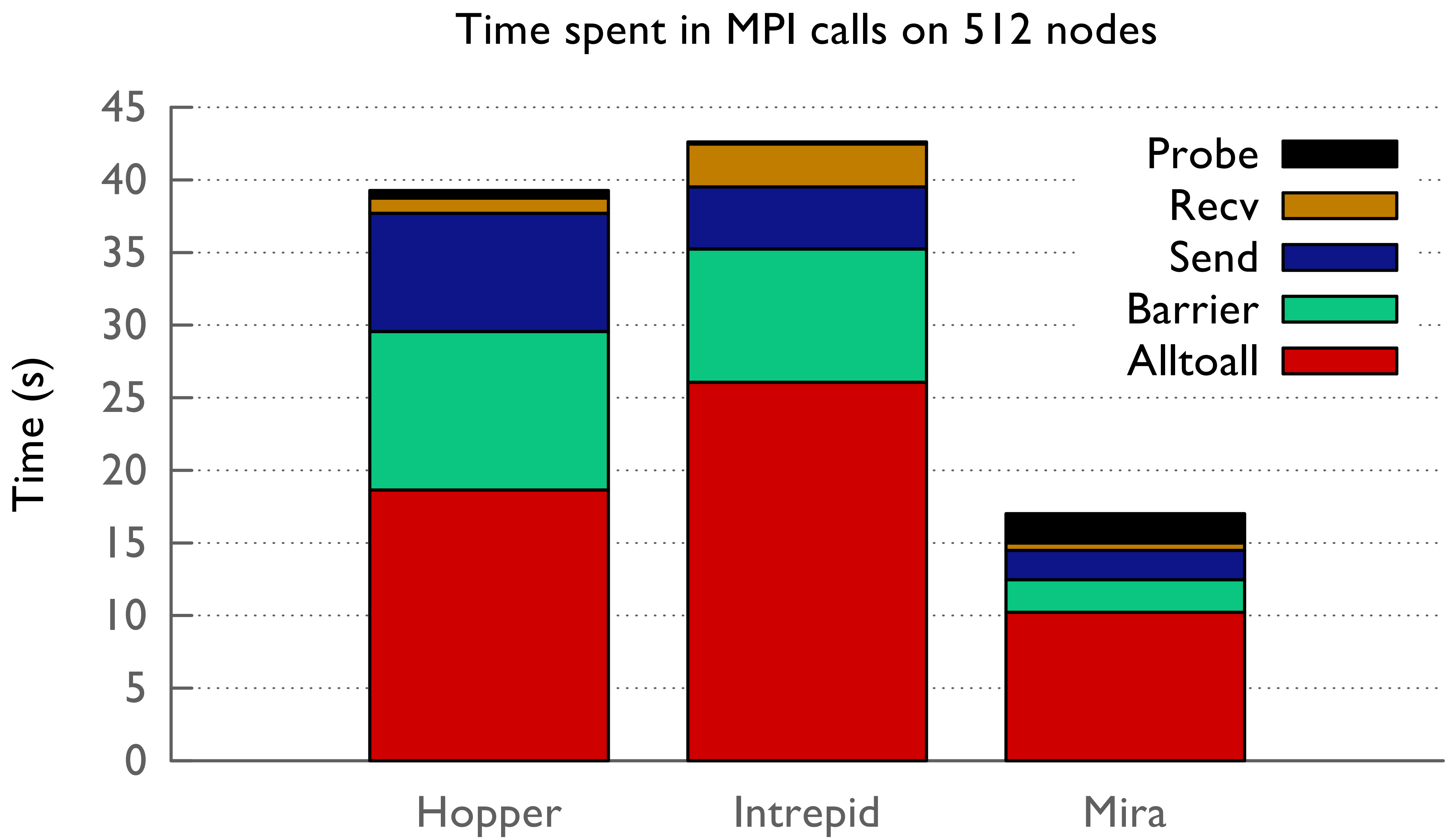
- Debugging performance issues
- Quantifying the effect of various software changes on performance
  - code changes
  - compiler/software stack changes
- Requesting time for a batch job
- Writing allocation proposals



pF3D characterization



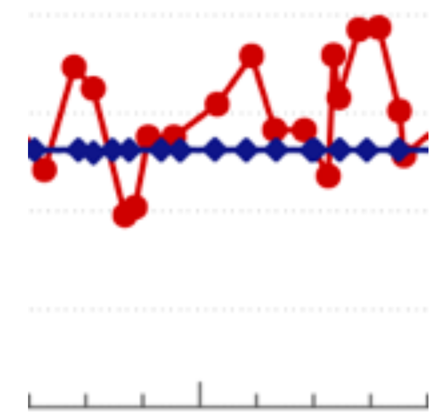
pFD characterization

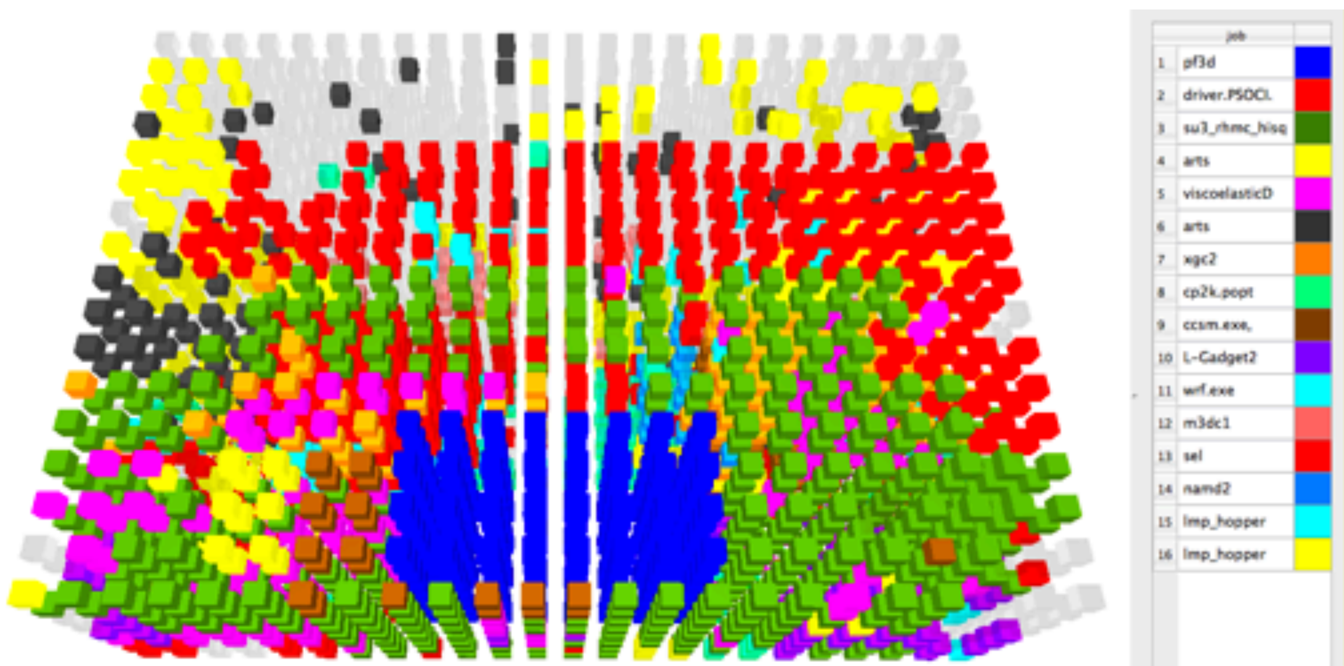


Sources of variability

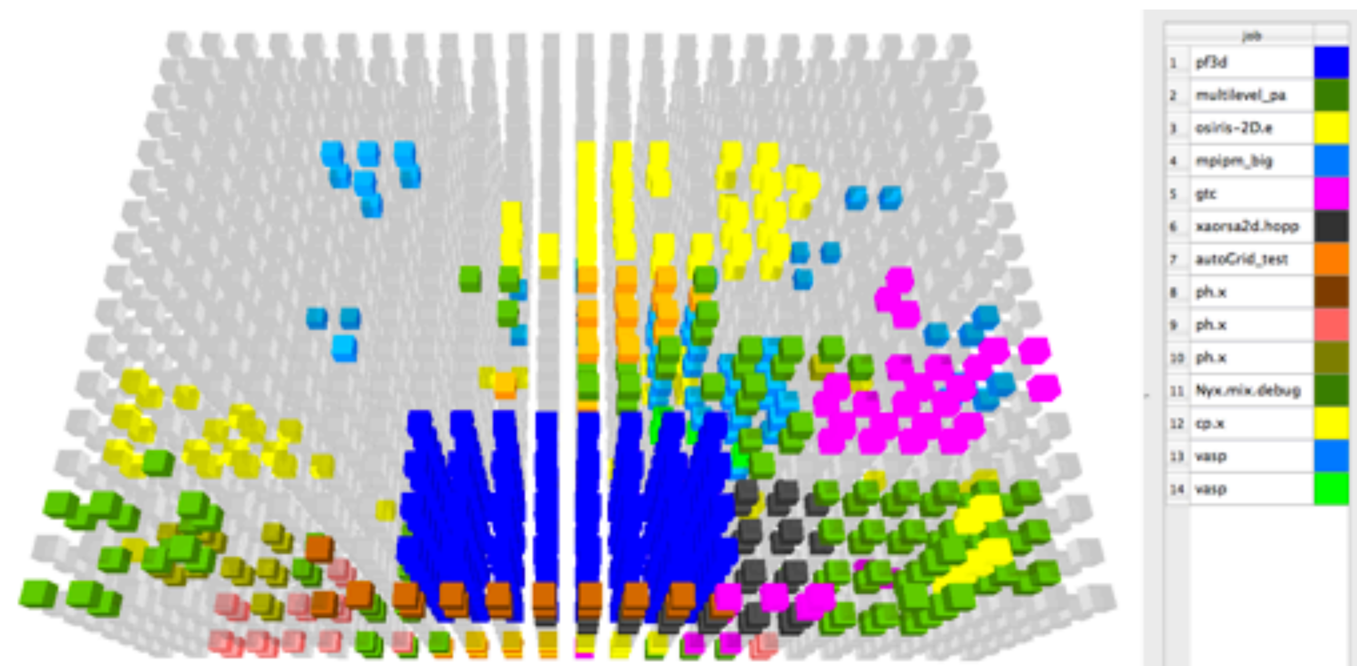
- Operating system noise (OS jitter)
  - OS daemons running on some cores of each node
- Placement/location of the allocated nodes for the job (Allocation shape)
- Contention for shared resources (Inter-job contention)
  - Sharing network links with other jobs

4x8x8-shaped pF3D job





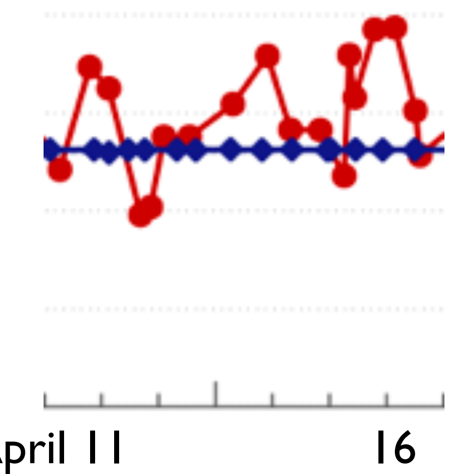
April 11

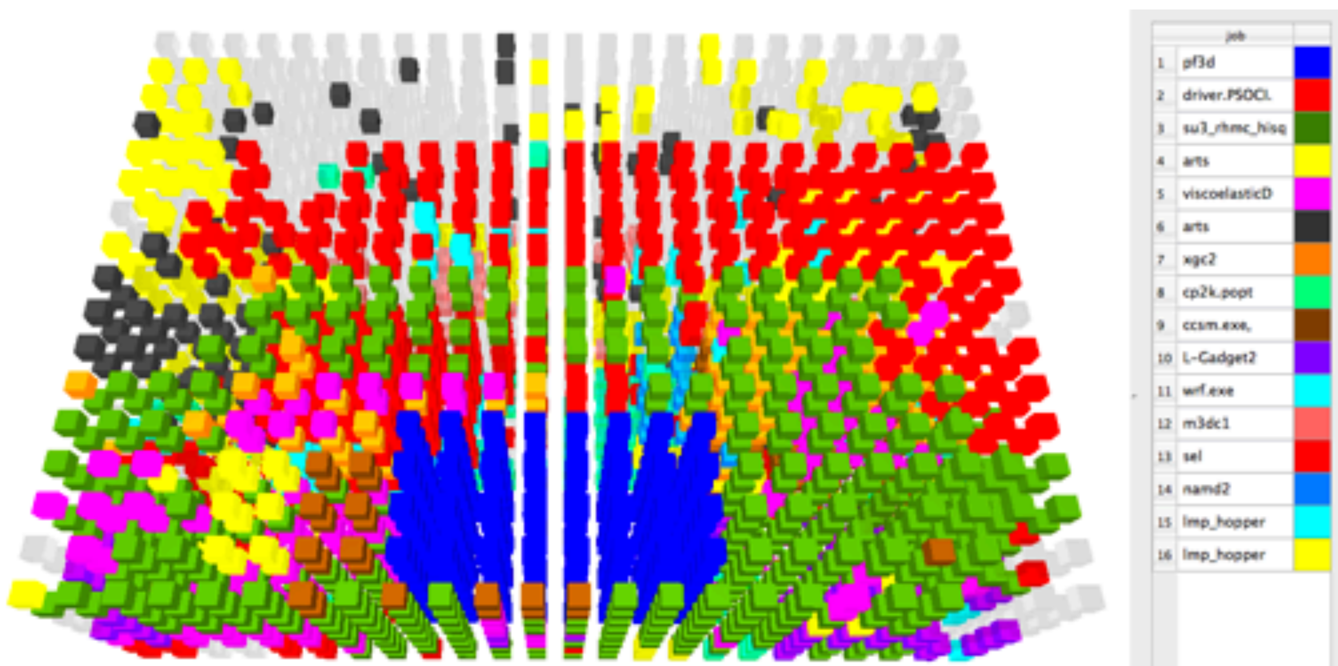


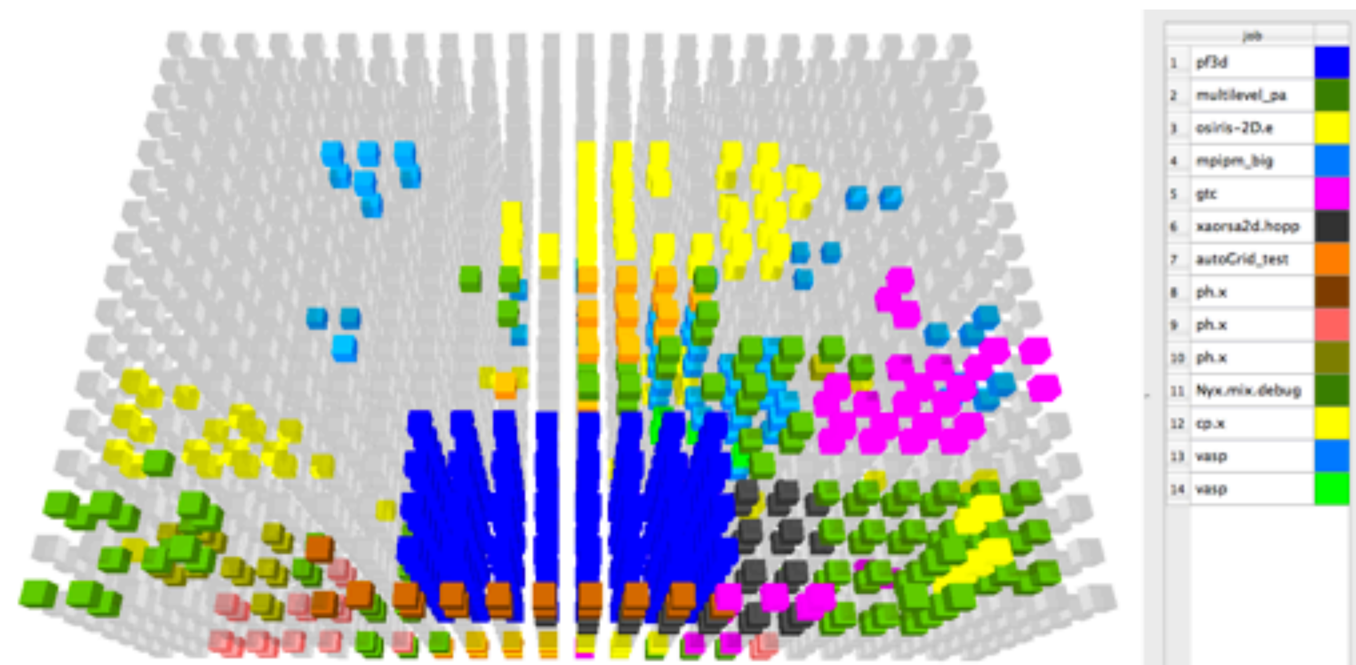
April 16

https://scalability.llnl.gov/performance-analysis-through-visualization/software.php

4x8x8-shaped pF3D job







April 11

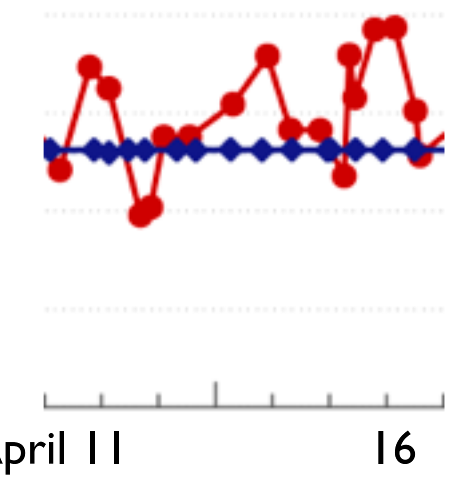
MILC job in green

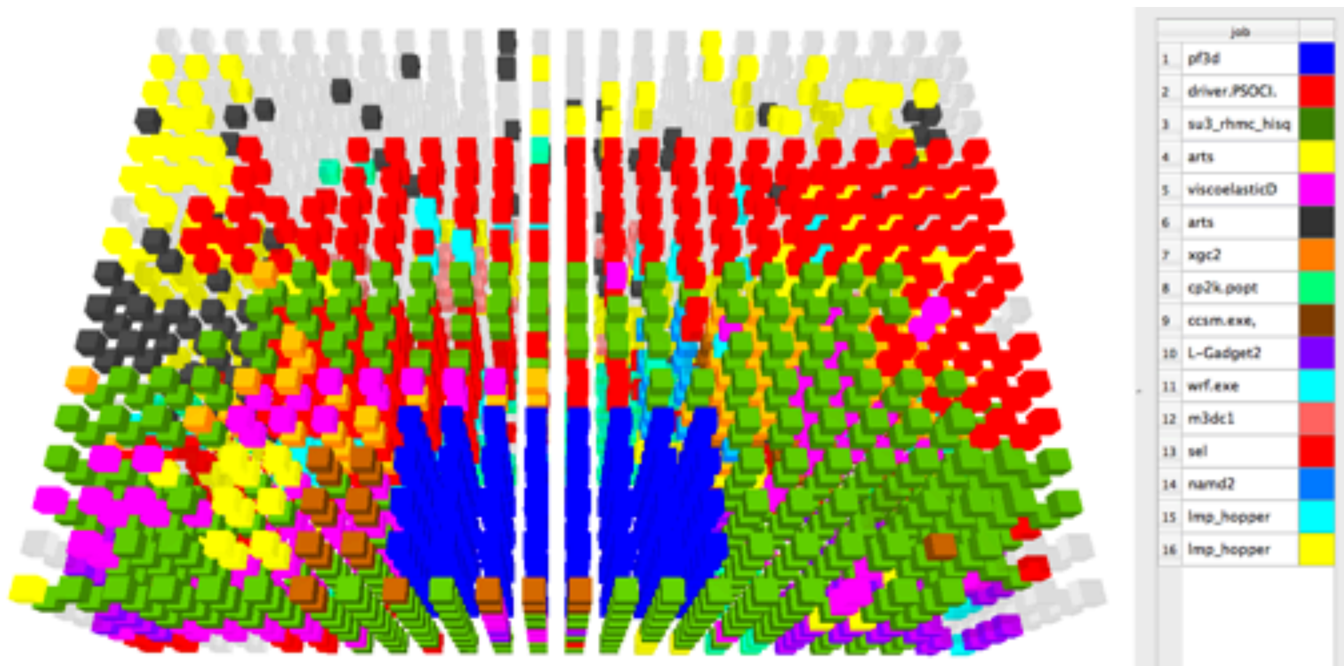
April 16

25% higher messaging rate

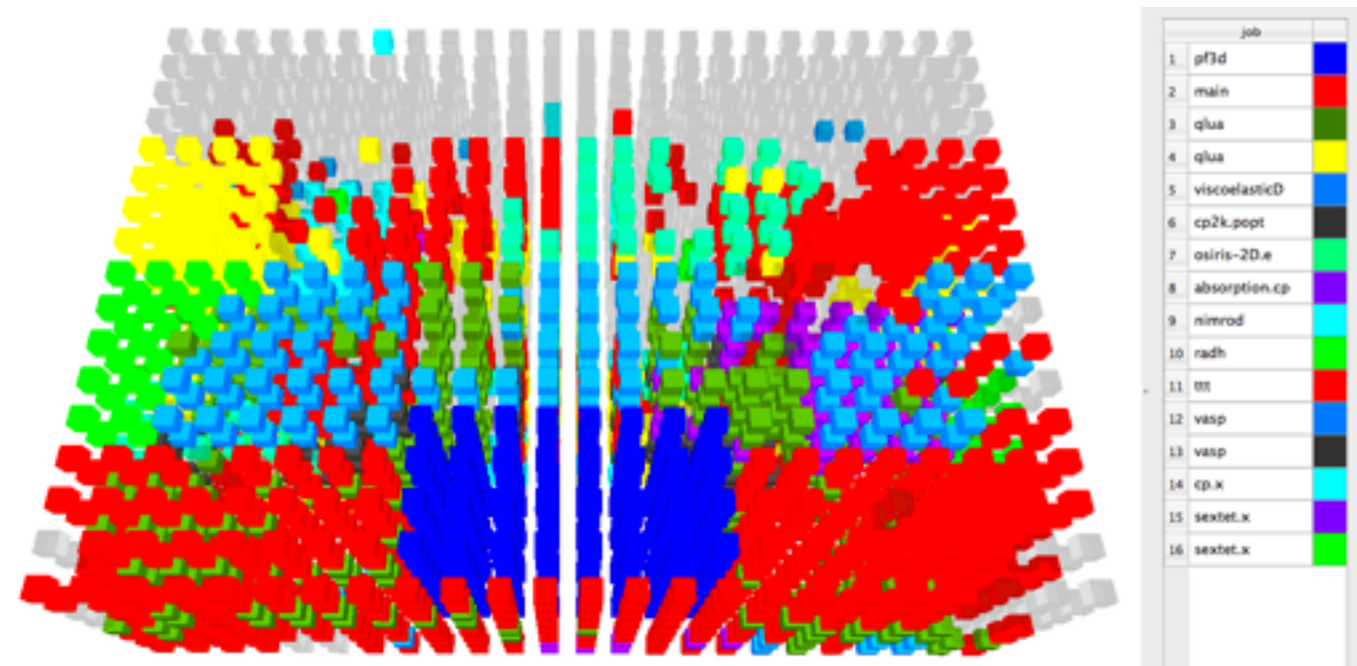
https://scalability.llnl.gov/performance-analysis-through-visualization/software.php

4x8x8-shaped pF3D job



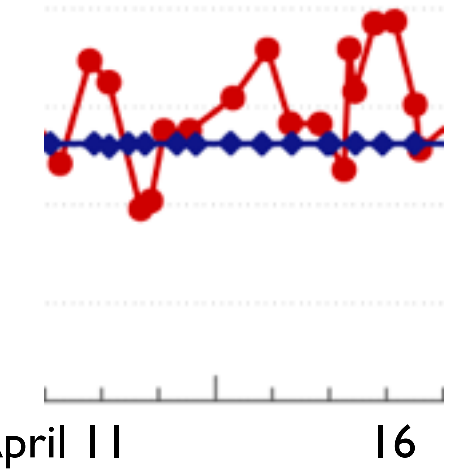


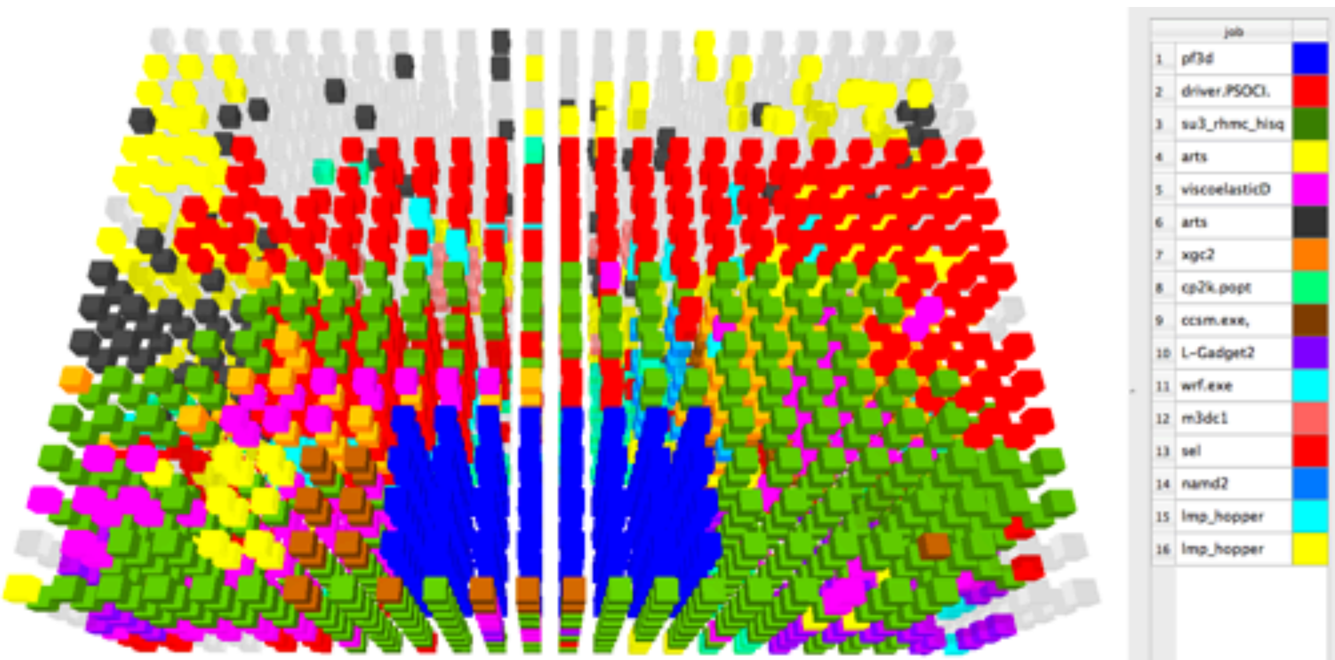
April 11

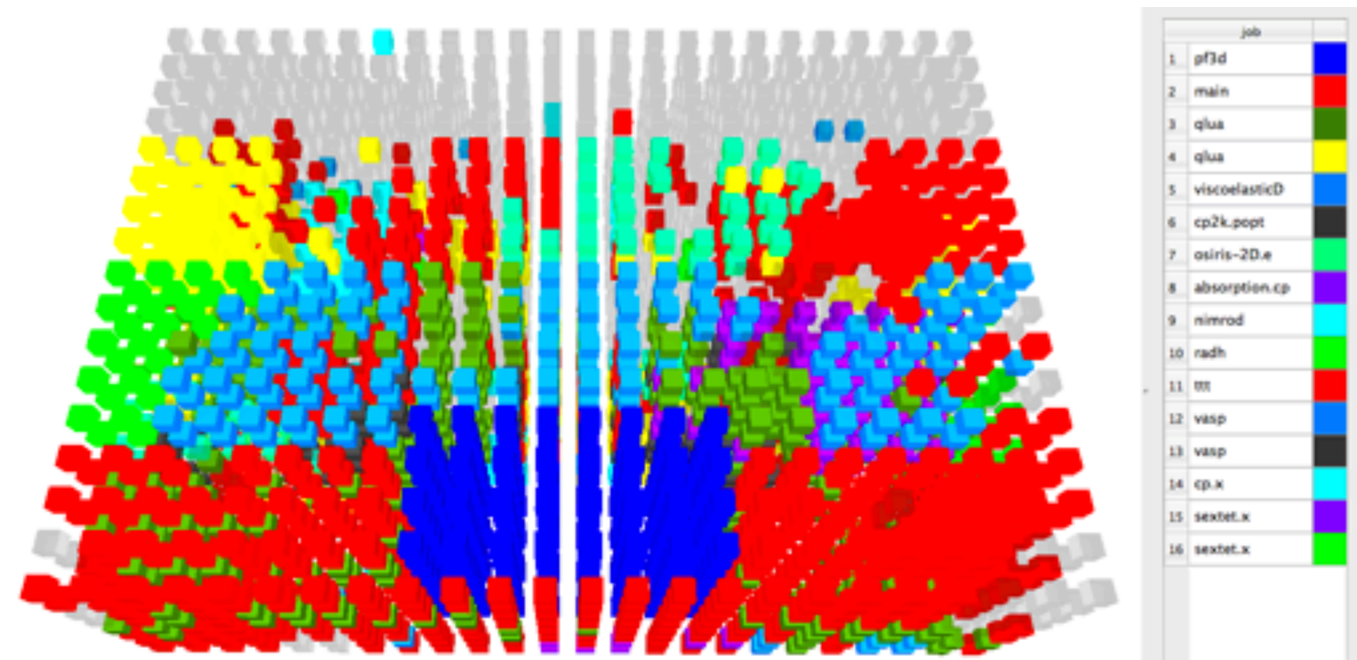


April 16b

4x8x8-shaped pF3D job







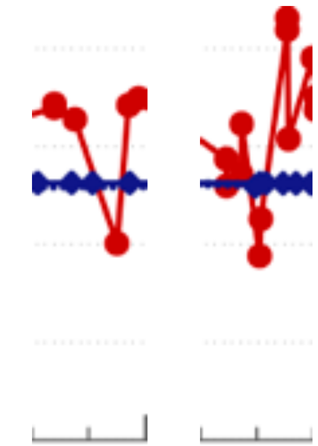
April 11

MILC job in green

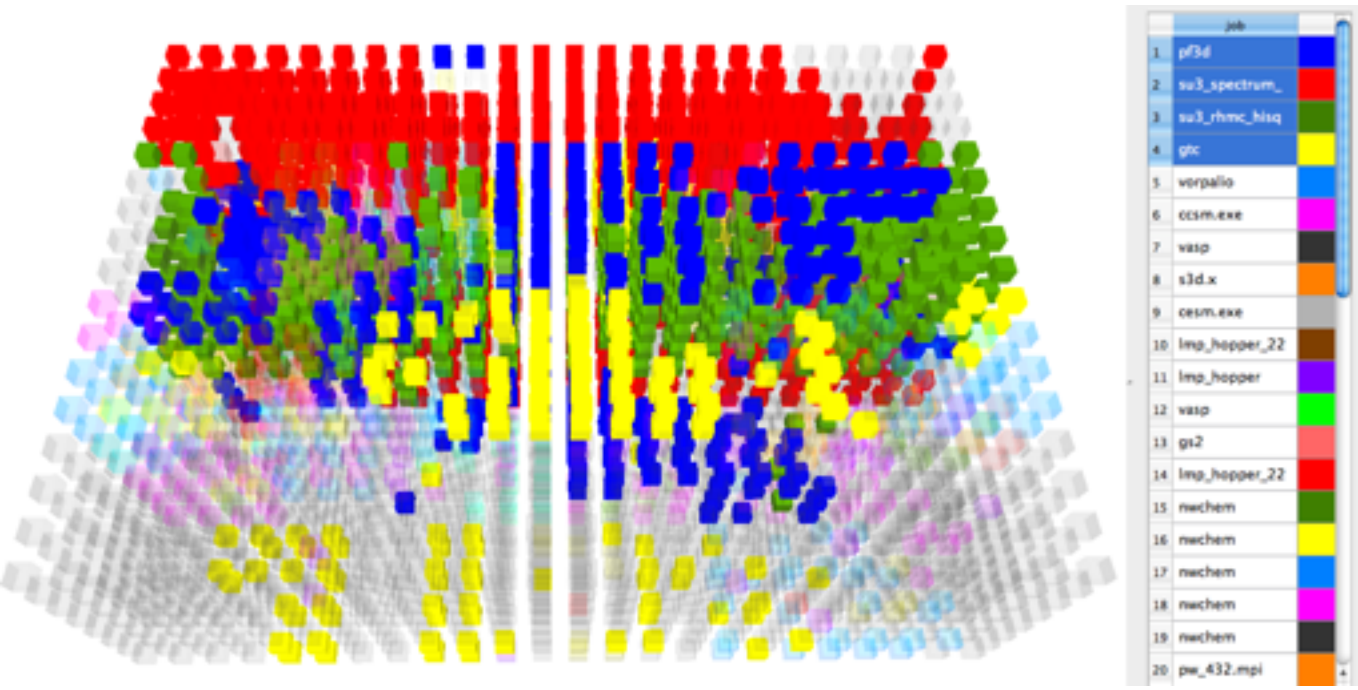
April 16b

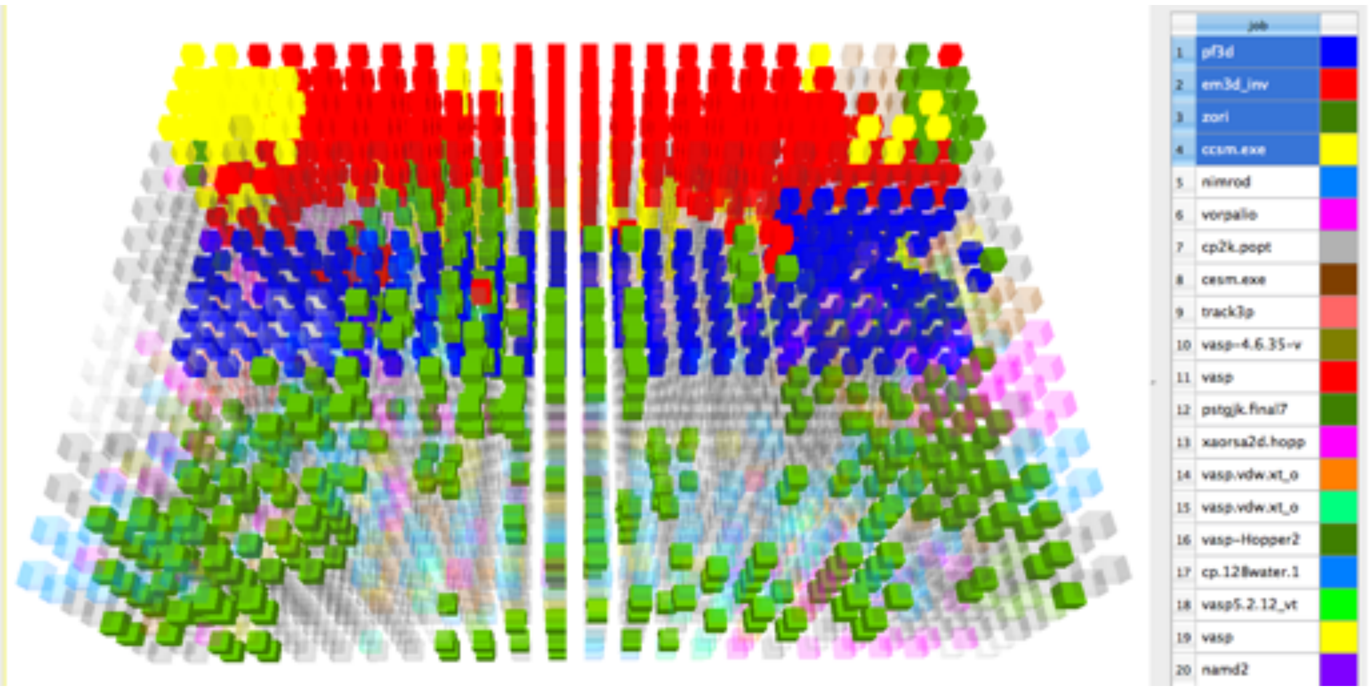
27.8% higher messaging rate, LSMS is not communication-heavy

Slowest vs. fastest job



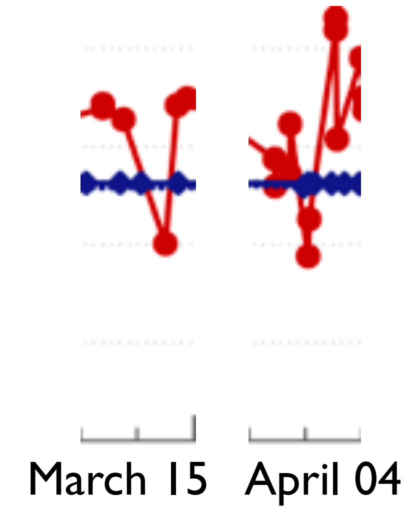
March 15 April 04

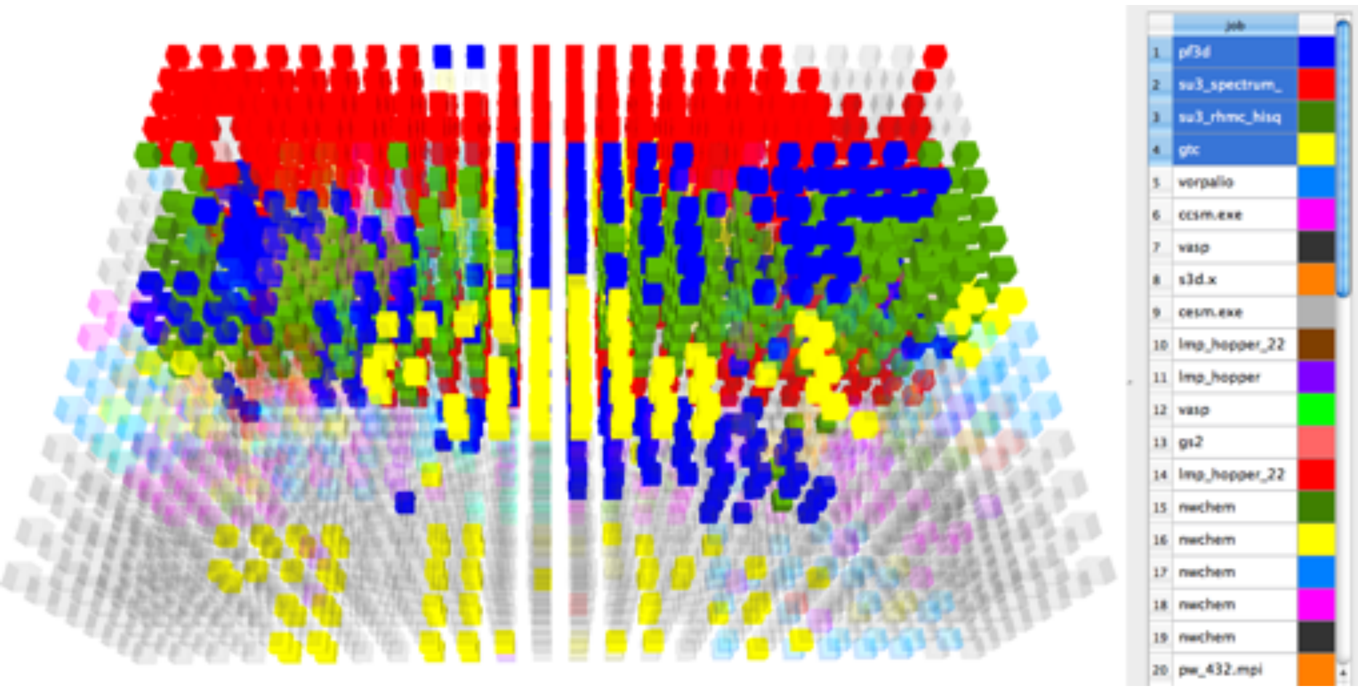




March 15 April 04

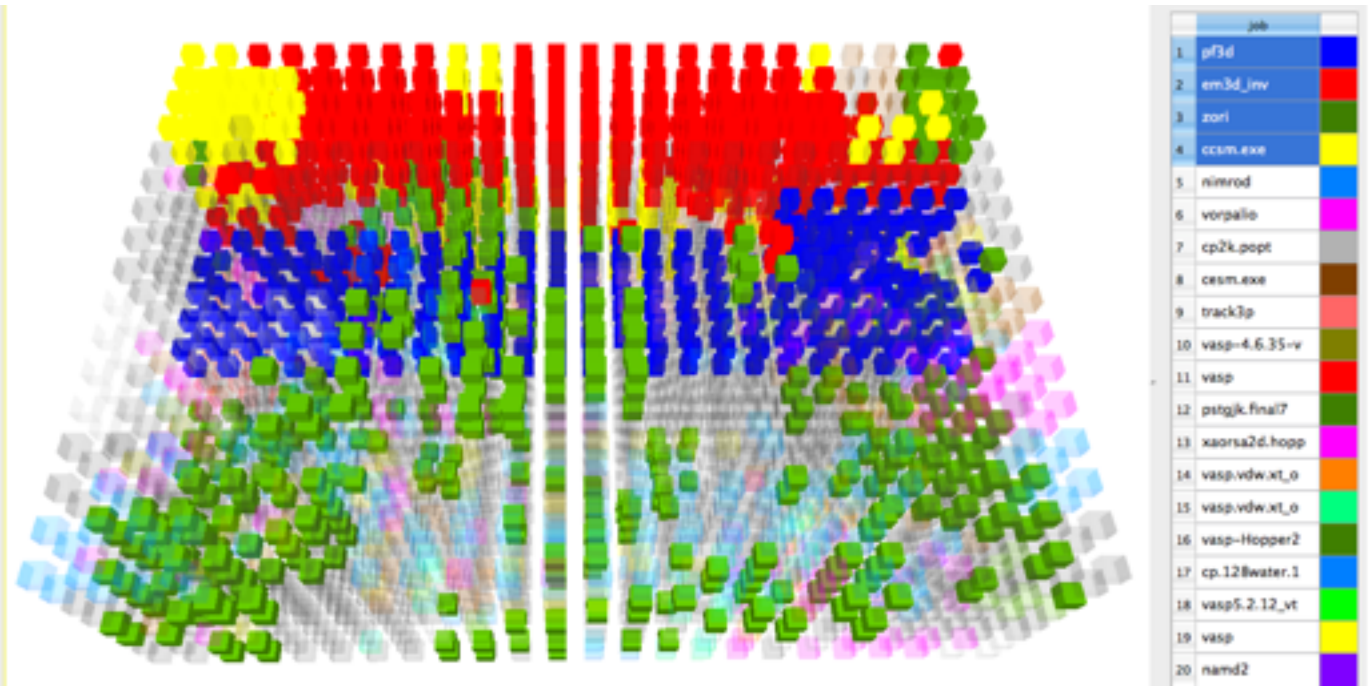
Slowest vs. fastest job





March 15

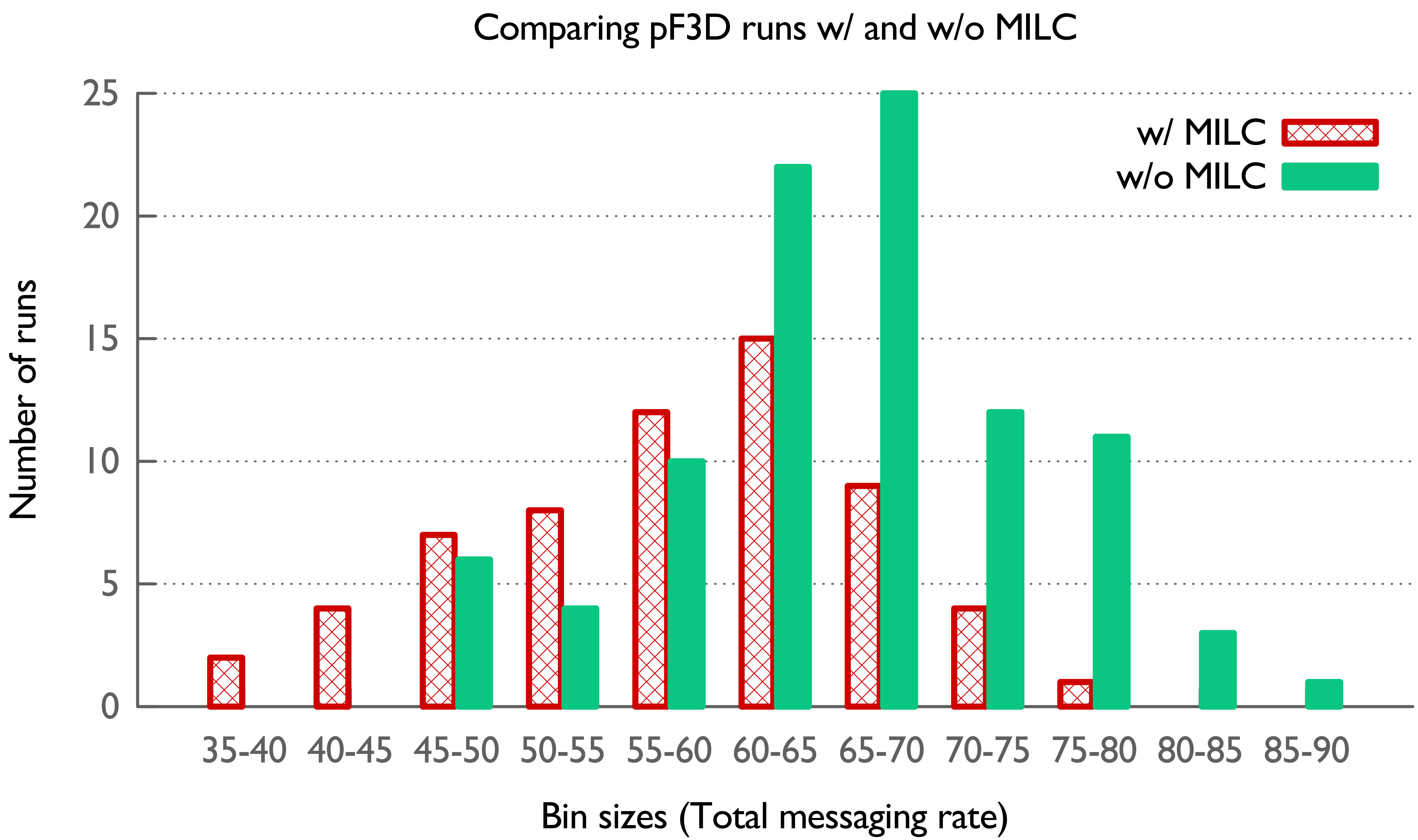
Three conflicting jobs, two MILC



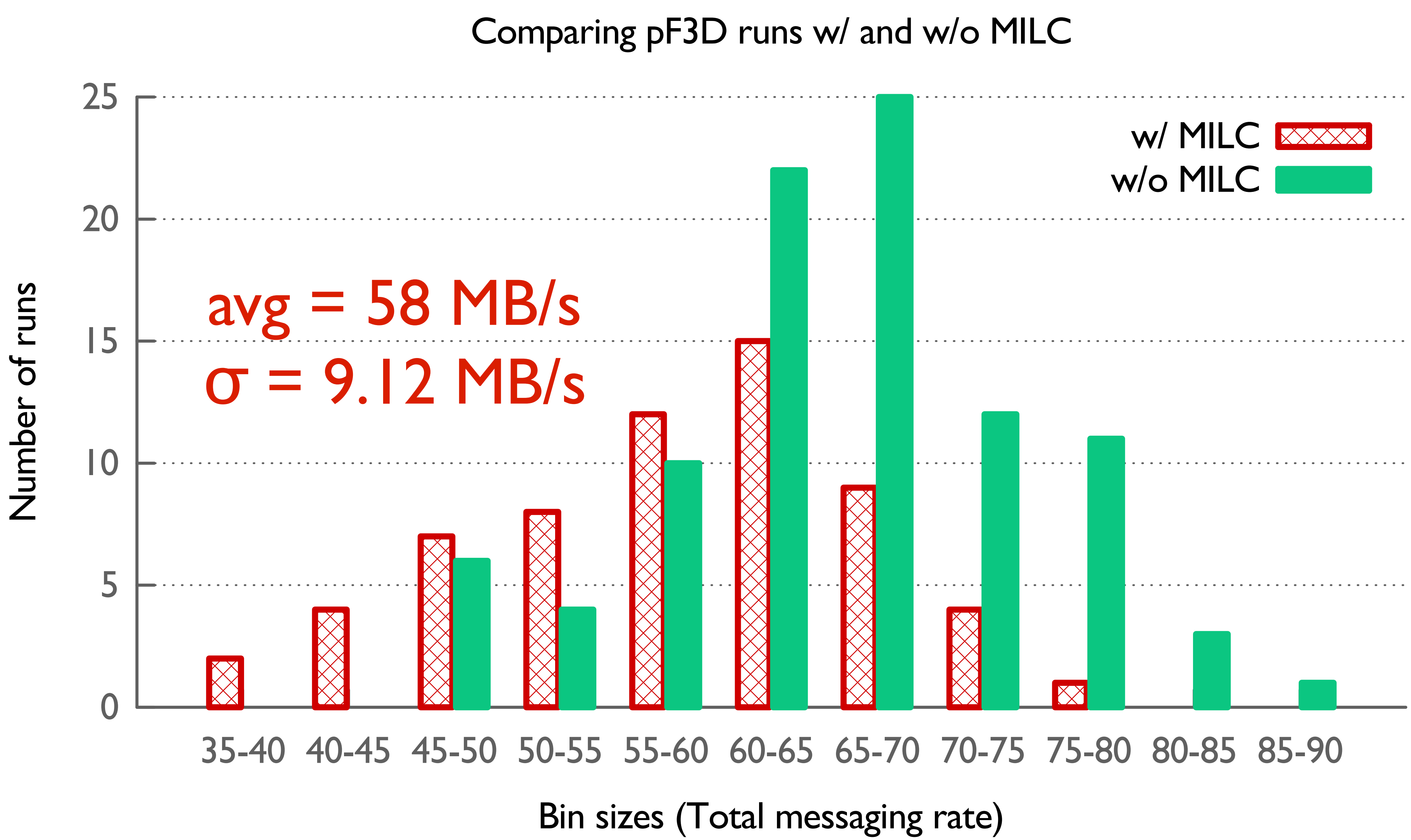
April 04

2.29X higher messaging rate

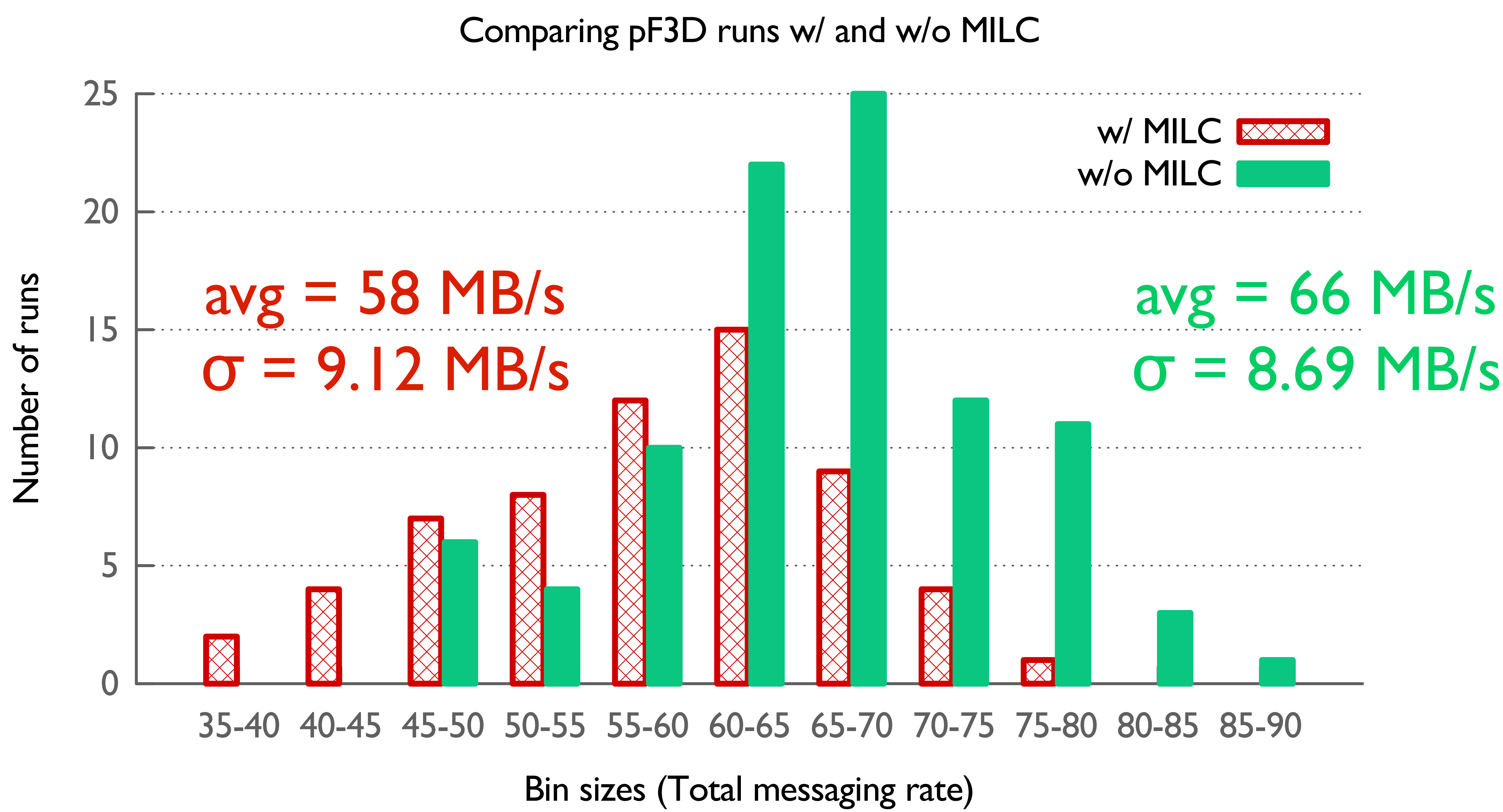
Effect of MILC on pF3D



Effect of MILC on pF3D



Effect of MILC on pF3D



Performance tip!

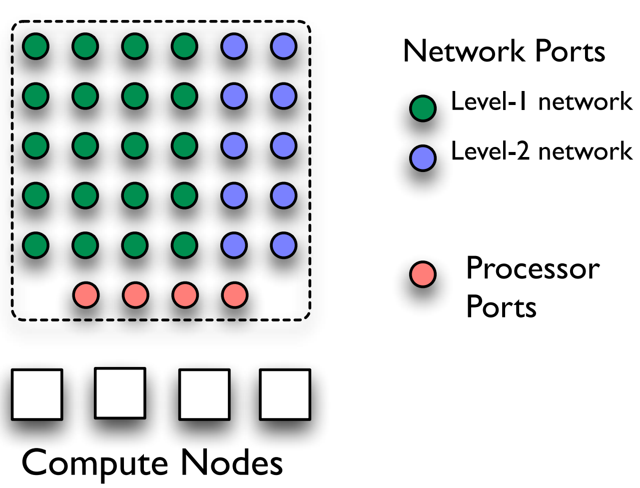
- Variability insignificant on IBM Blue Gene systems
- OS noise and allocation shape have a weak correlation with performance
- The placement of other jobs around a job can affect its performance significantly

http://www.hpcwire.com/2013/11/16/sc13-research-highlight-goes-performance-neighborhood/

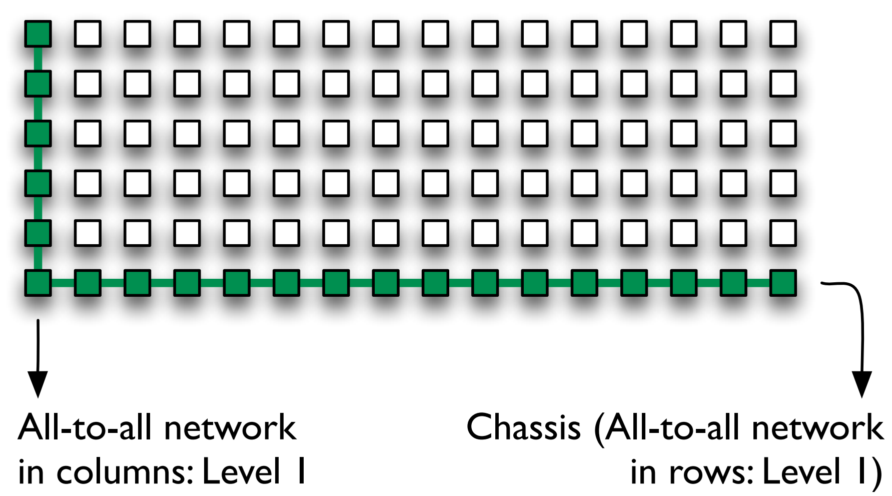
Modeling job placements and message routing

- Dragonfly topology: a two-level hierarchical topology
- Routing choices: static (deterministic) vs. dynamic (adaptive), direct vs. indirect (random jumps)
- Placement options: random, round-robin, blocked

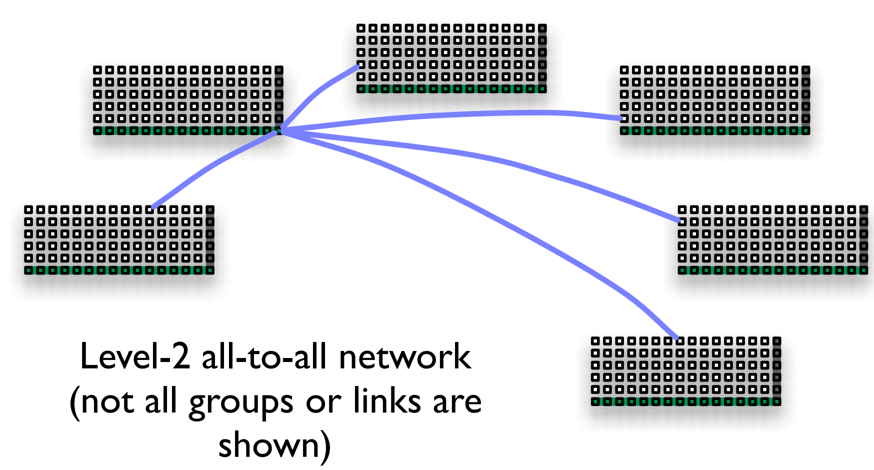
A DRAGONFLY ROUTER



A GROUP WITH 96 ROUTERS

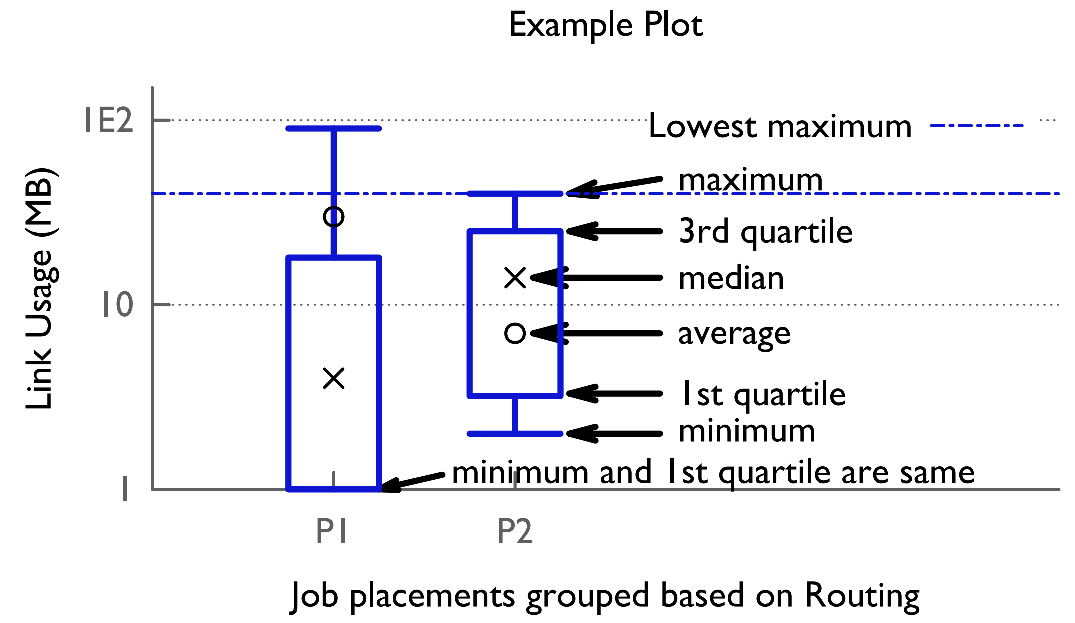


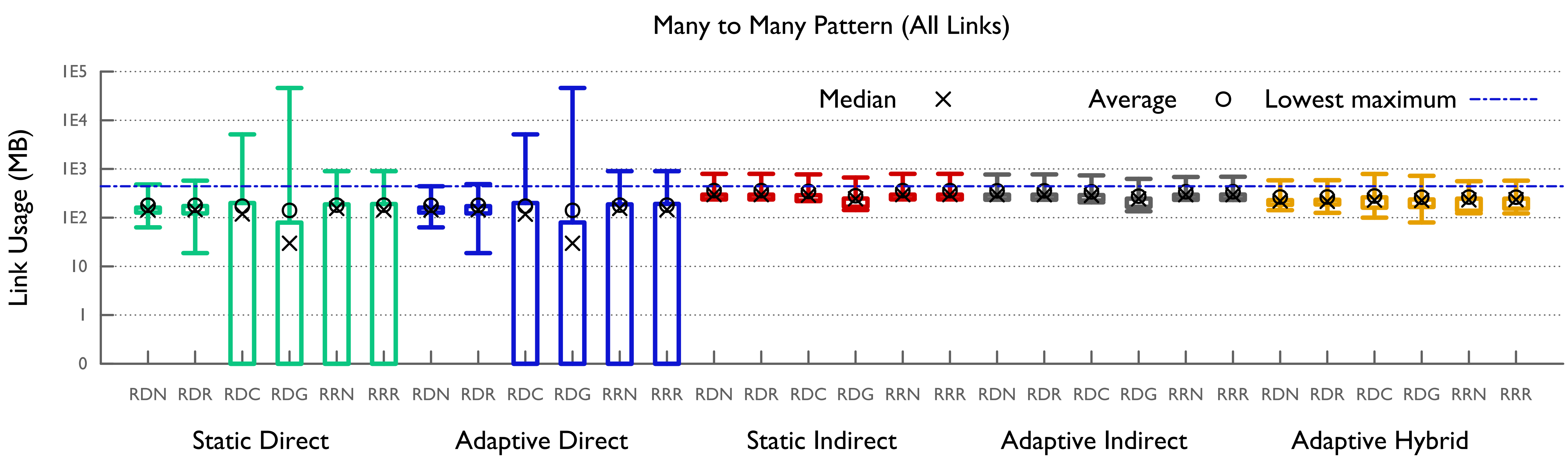
THE DRAGONFLY TOPOLOGY



Single jobs

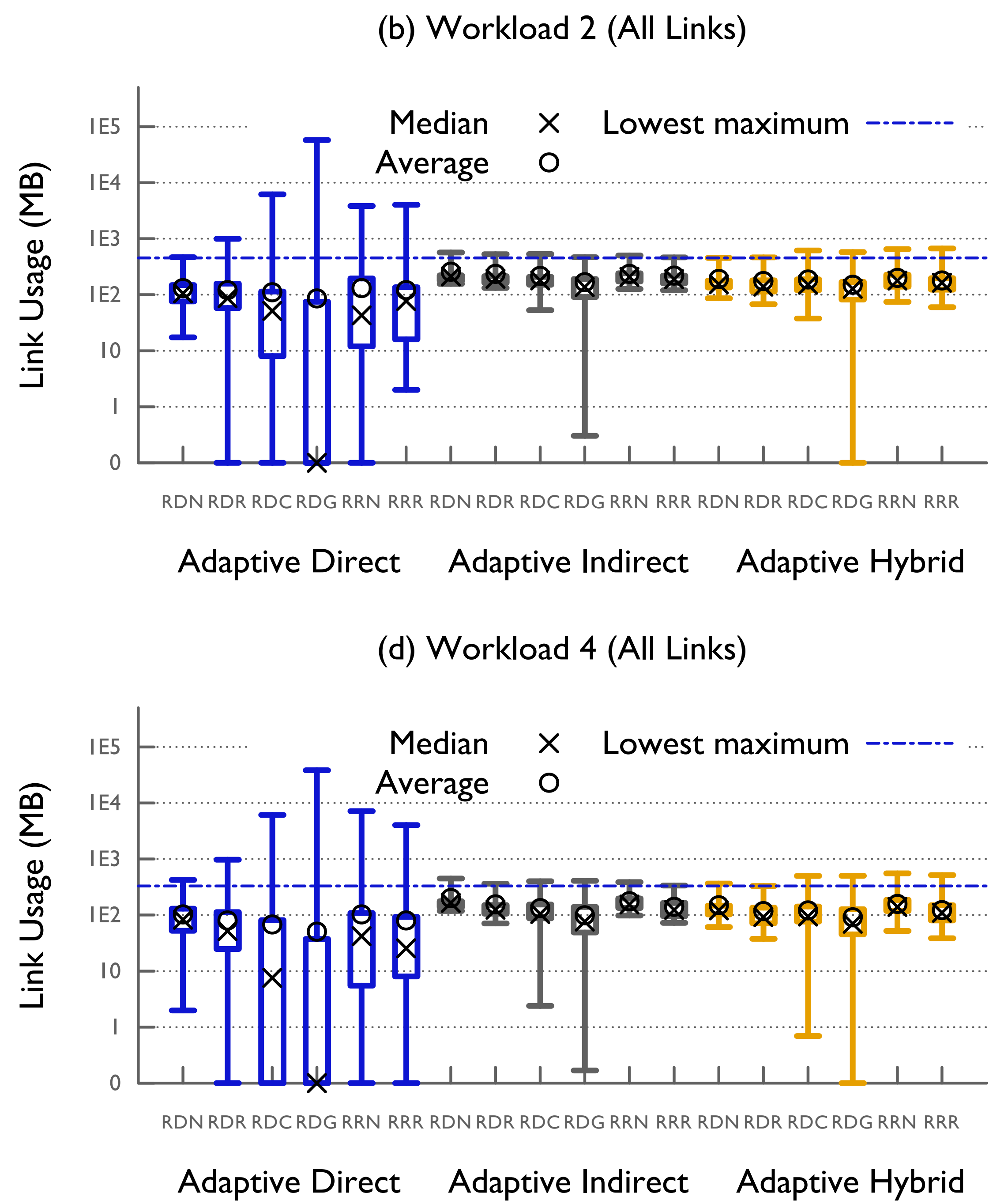
- All-to-all over sub-communicators
- Various traffic metrics





Parallel job workload

- Representative of NERSC workloads
- Static routing out of the question
- Routings with indirect jumps preferred



N. Jain et al. Maximizing network throughput on the dragonfly interconnect. In *submission to the ACM/IEEE International Conference for High Performance Computing, Networking, Storage and Analysis*, SC '14. 2014.

Summary

- Optimizing communication is the #1 priority
  - Minimize off-node communication
  - Map remaining off-node communication carefully
- Job placements and mapping are non-intrusive methods for improving performance
- Going forward: modeling and simulation will be crucial for:
  - designing future networks
  - predicting application performance

http://computation-rnd.llnl.gov/extreme-computing/interconnection-networks.php

This work was funded by the Laboratory Directed Research and Development Program at LLNL under project tracking code 13-ERD-055: STATE - **S**calable **T**opology **A**ware **T**ask **E**mbedding.

Charm++ Workshop ◆ April 30, 2014

LLNL: Abhinav Bhatele, Peer-Timo Bremer, Todd Gamblin, Katherine E. Isaacs, Steven H. Langer, Kathryn Mohror, Martin Schulz

Illinois: Ronak Buch, Nikhil Jain, Harshitha Menon, Laxmikant V. Kale, Michael Robson

Utah: Amey Desai, Aaditya G. Landge, Valerio Pascucci

Purdue: Ahmed Abdel-Gawad, Mithuna Thottethodi

LBL: Brian Austin, Nicholas J. Wright

This work was performed under the auspices of the U.S. Department of Energy by Lawrence Livermore National Laboratory under Contract DE-AC52-07NA27344.