SAND2017-2187C

Understanding and Avoiding Performance Variability in High Performance Networks

Presenter: Ryan E. Grant (SNL)

Taylor Groves (SNL and University of New Mexico), Kevin Pedretti SNL), Anne Gentile (SNL) and Dorian Arnold (University of New Mexico)

Sandia National Laboratories is a multi-mission laboratory managed and operated by Sandia Corporation, a wholly owned subsidiary of Lockheed Martin Corporation, for the U.S. Department of Energy's National Nuclear Security Administration under contract DE-AC04-94AL85000. SAND NO. 2015-10083 C

Frequently Asked Question

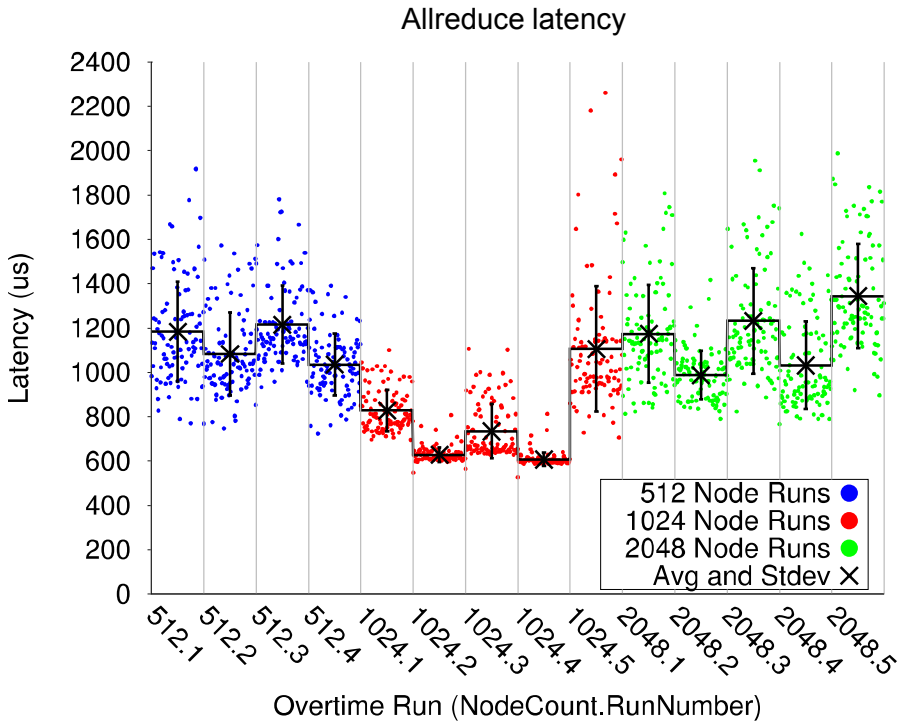
Sandia National Laboratories

Why is my code running so slow from time to time? Could it be the network/MPI?

Answer: It’s complicated

Performance Variation in MPI

- Performance variation in MPI can have significant impact on code performance
- Latencies can range almost 4X for a single allreduce operation over different runs
- Understanding this variation is key to leveraging system performance for jobs
- Network conditions need to be understood when jobs are placed on a system
- Understanding the intersection of different jobs communications is difficult



Performance Variation in MPI

Sandia National Laboratories

- Not all performance variation is due to network congestion/interference
- OS noise can cause this issue as well
- What variation is caused by the network and what variation is caused by other factors?
- Goal:
  - Determine impact on MPI point to point and collectives over time on a production system
  - Mitigate OS noise impact by not fully loading all CPUs on nodes under test
  - Correlate network performance counter data with observed performance
  - Characterize network interference over time with observed causes

Network Performance Variation

Sandia National Laboratories

- Difficult to attribute to a single factor
- Normally caused by an intersection of multiple jobs behaviors
- Jobs utilize the network at different times during execution
  - Makes determining network conditions from job list alone difficult
  - Communication frequency dependent on application and workload
- Shared network resources can be hard to reason about
- Some network topologies make reasoning easier
  - Using a 3D torus
    - Easier to reason about job placement and traffic patterns

Assessment of Variation

- In order to assess network performance variation from an MPI perspective – We developed the Overtime benchmark

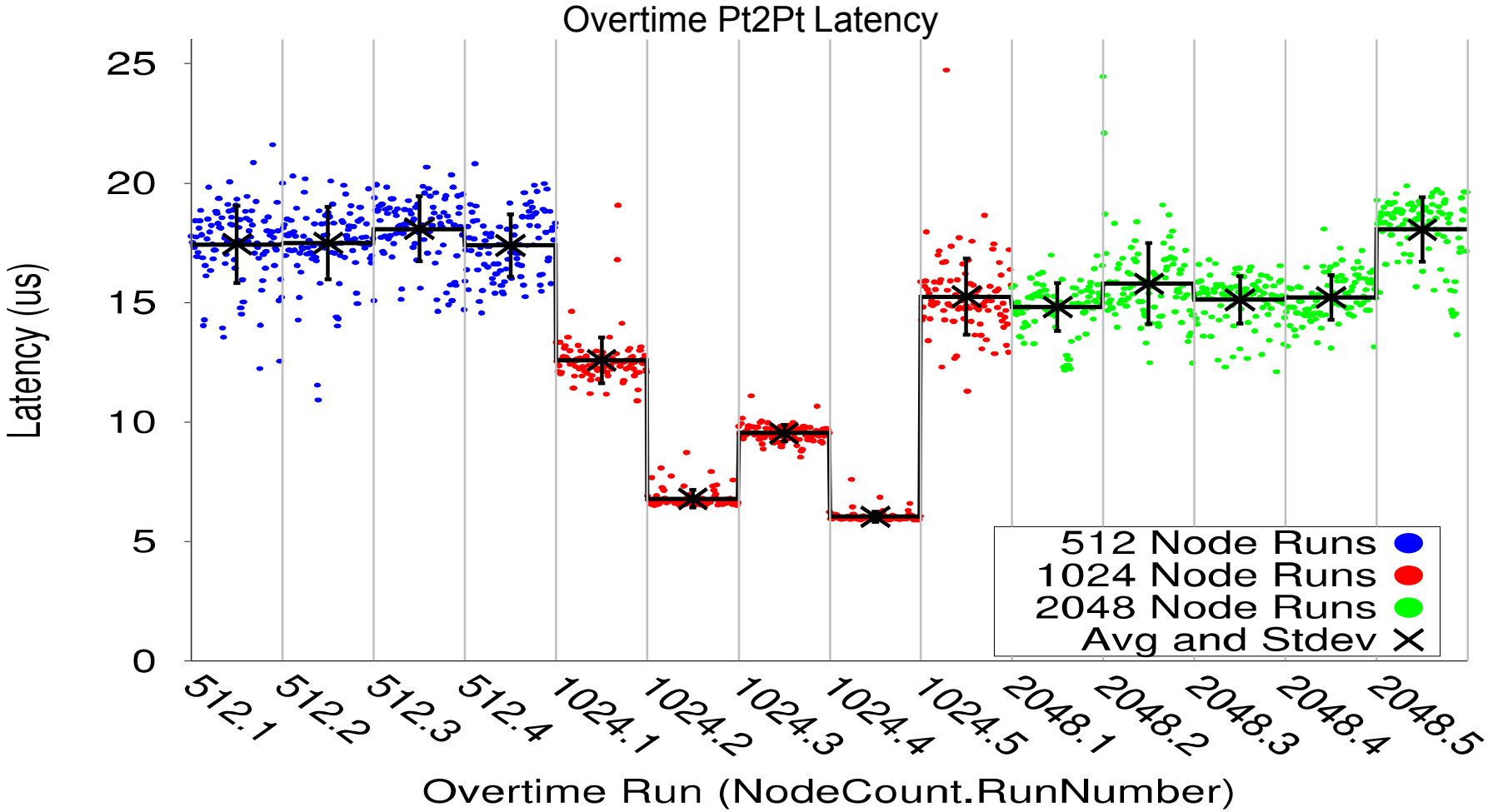
- Overtime is a tool that measures performance and record network performance counters
  - MPI pt2pt latency, bandwidth, and all-reduce performance
  - Alternates between MPI performance and observation of system with no communication
  - Sleep periods are adjustable – default to exact time period of previous MPI tests so network counters are comparable
  - Leverages rich set of network performance counters for Cray Gemini networks

Experimental Setup

Sandia National Laboratories

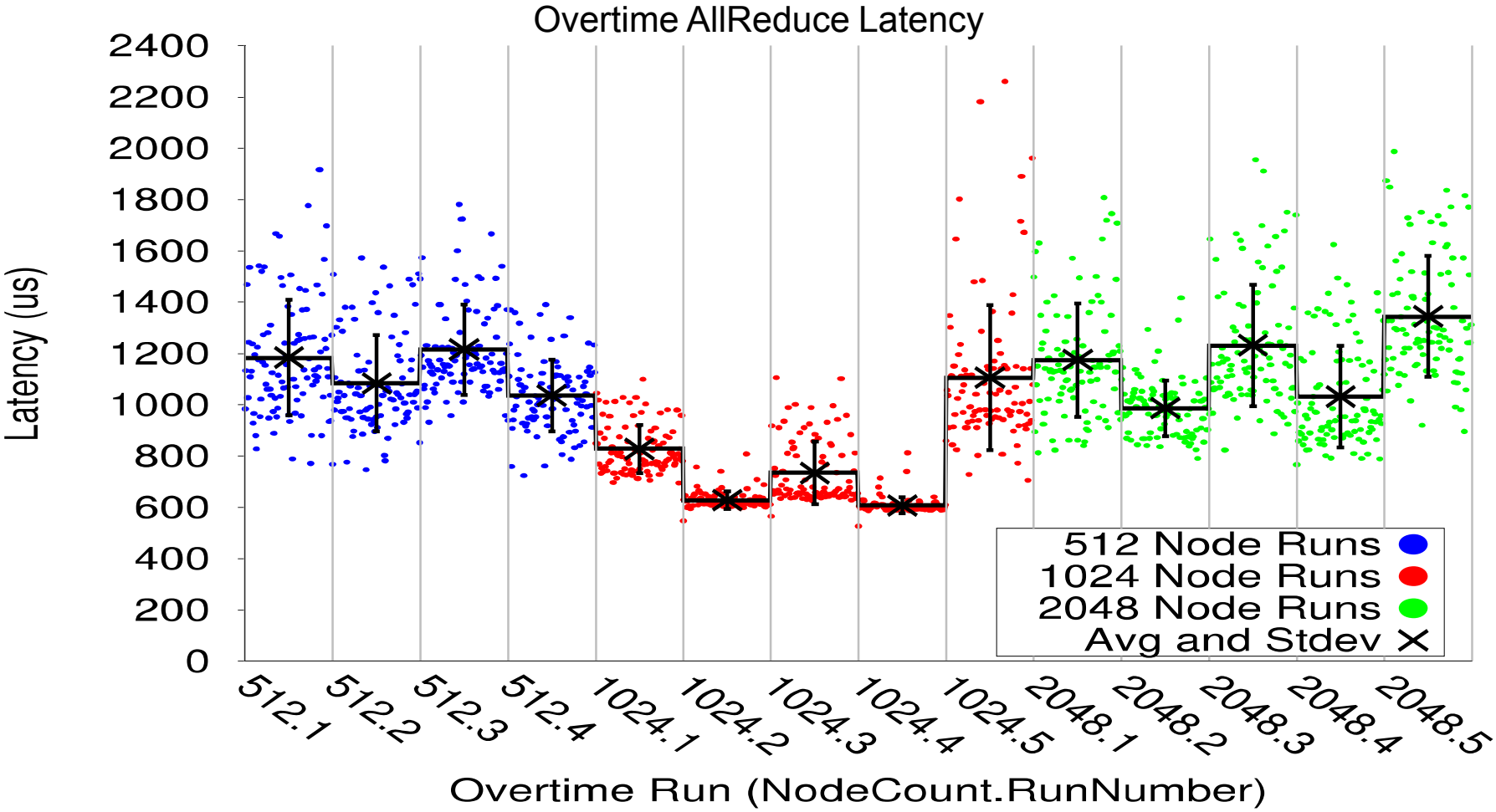
- Testing performed on the Blue Waters system as the National Center for Supercomputing Applications
- 22,640 Cray XE nodes and 4,228 Cray XK7 nodes
  - Only used XE nodes
- 237 XE cabinets, 44 XK cabinets, 13.34 PF peak
- All tests performed during regular production time
- Cray Gemini 3D torus network
  - 24x24x24
  - 13,824 Gemini chips in the system
    - Each Gemini connected to 6 neighbors
  - 2 compute nodes share a Gemini
  - Peak injection bandwidth 9.6 GB/s

Results and Observations



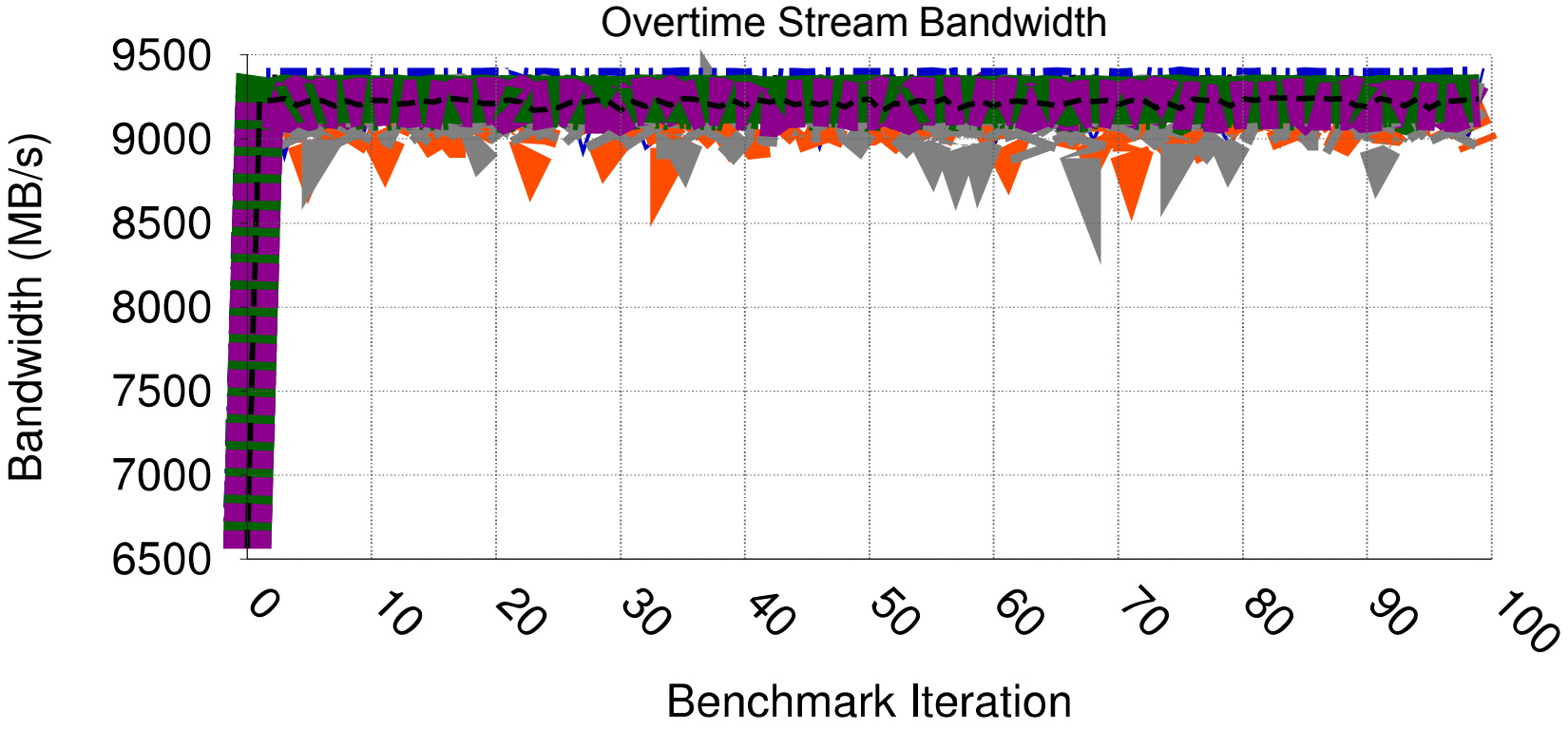
- Latency swings over time can be significant
- Although relatively stable in a sampling period (1 hour)

Results and Observations



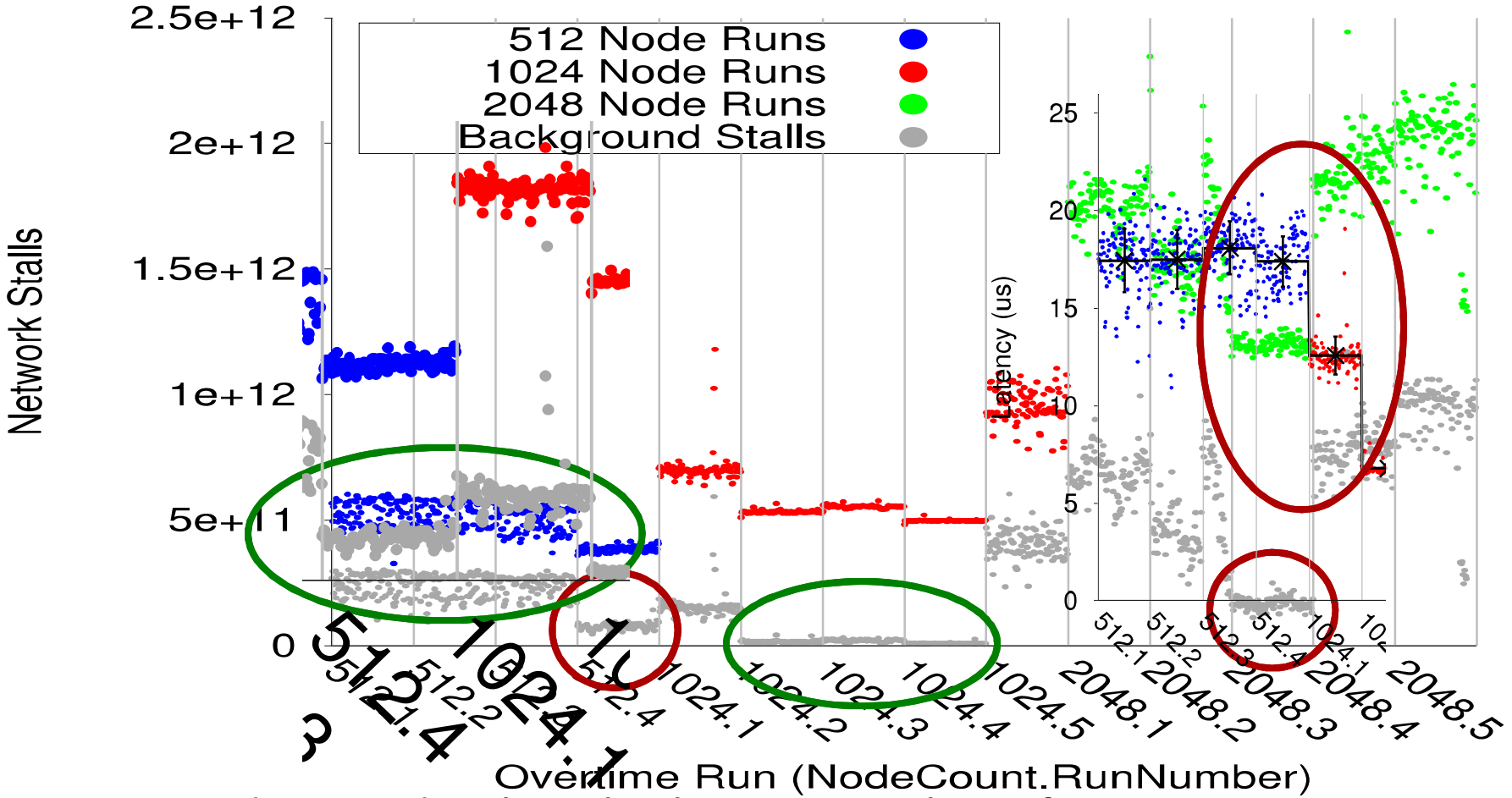
- All-reduce sees variation, but with higher std. dev.
- No major trends with obvious changes within time periods
- Bandwidth is much less impacted over time

Results and Observations



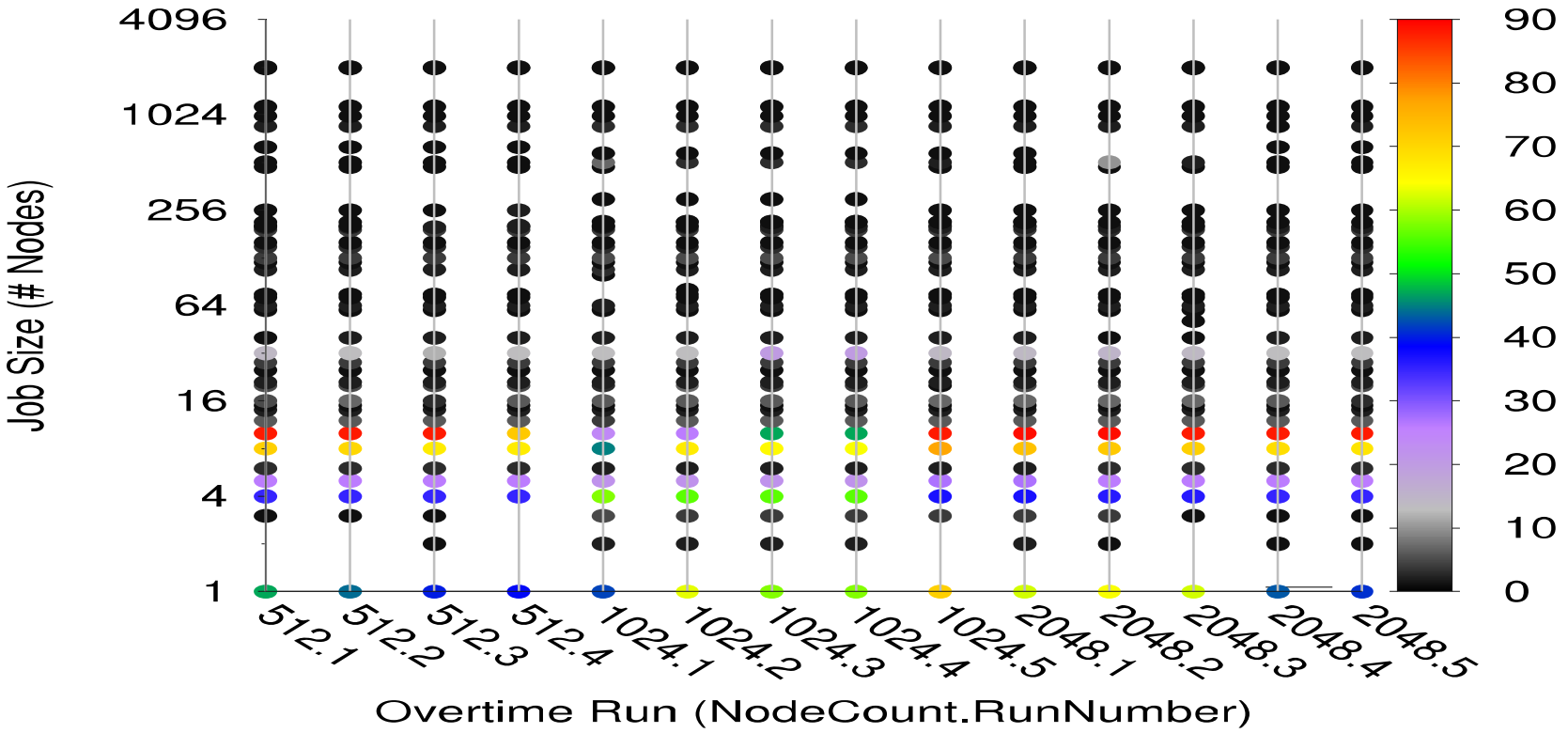
- Bandwidth is much less impacted over time than latency or reductions

Network Performance Counters



- To understand, take a look at network performance counters
- Some correlation between idle stalls and observed perf
- Important exceptions to this observation – multiple factors

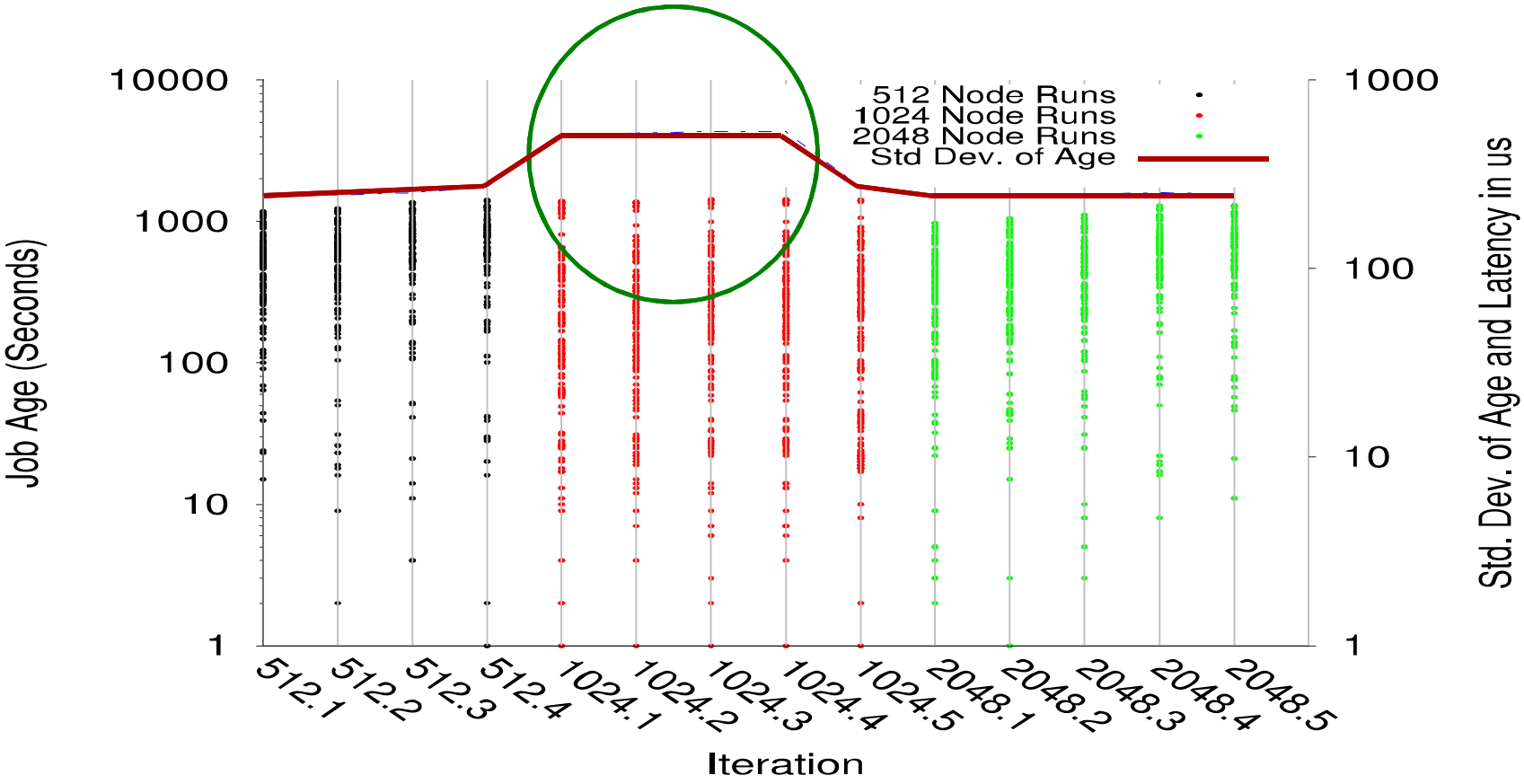
Job Mix



Job size with color coding of number of jobs of that size executing

- Is job mix playing a role?
- Job sizes are relatively regular

Job Age

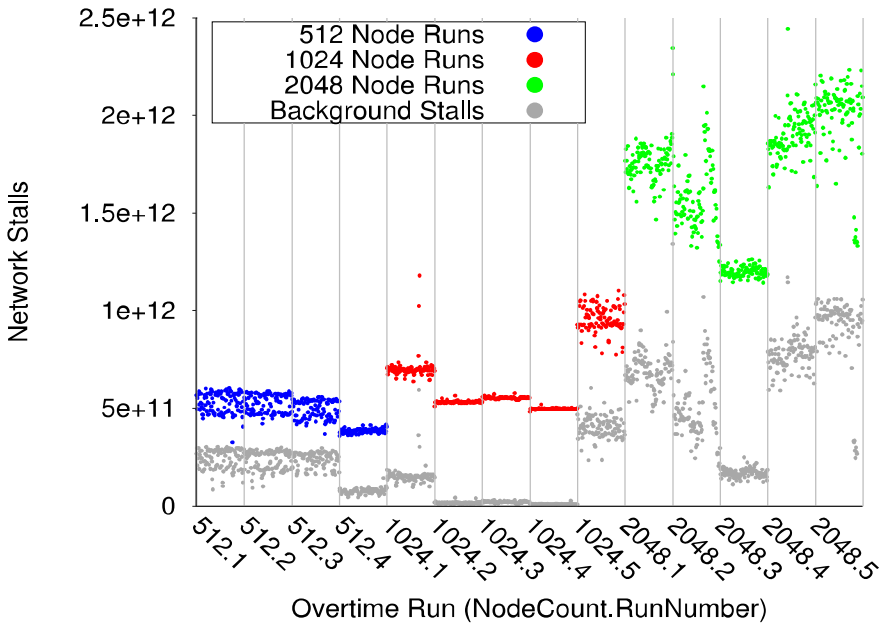


- Could age of the jobs running be a factor?
  - Jobs starting up could cause network variation

Observed Stalls vs. Job Age

- If job age correlates well with best runs why don't we see a change in resting stalls over other runs?
  - If job startup/completion is causing network interference, it should show up in the observed stalls over time…
- Conclusion:

  Network stalls are not sufficient to understand or predict network performance
- Separating stalls out by links (x,y,z) doesn't provide further insight either



The Good News

- Although multiple factors are at play when trying to predict performance based on job age/network counters…
- Network performance is relatively steady for 10-60 minute periods
- Actively measuring network performance provides reliable feedback
- Network measurements during idle periods provide reasonable feedback
  - With some false positives (optimistic network prediction)
- Overtime could be used to assess a potential job allocation
  - Determine if the predicted network performance matches requirements

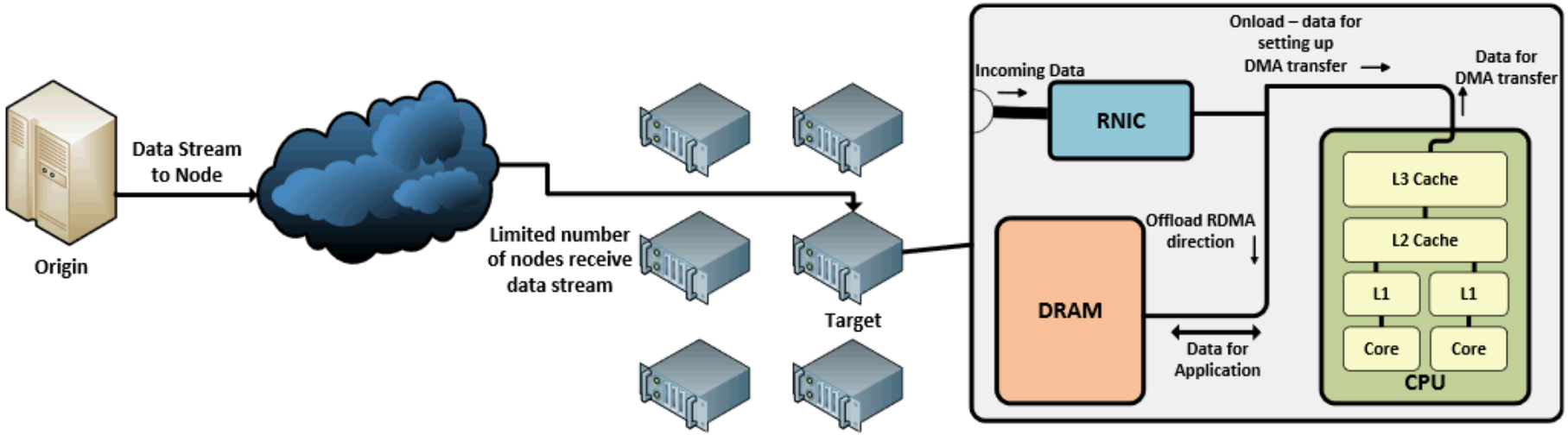
That's It?

- Not quite, there are other sources of network interference that can occur, even on node.

- Using RDMA traffic we can encounter a condition called Network induced Memory Contention (NiMC)

What is RDMA?

- Remote Direct Memory Access (RDMA)
  - Bypass the CPU and access memory directly
- Facilitates overlap between communication and computation



- However, there's a downside.

Small Scale Results (Sandy-Onload)

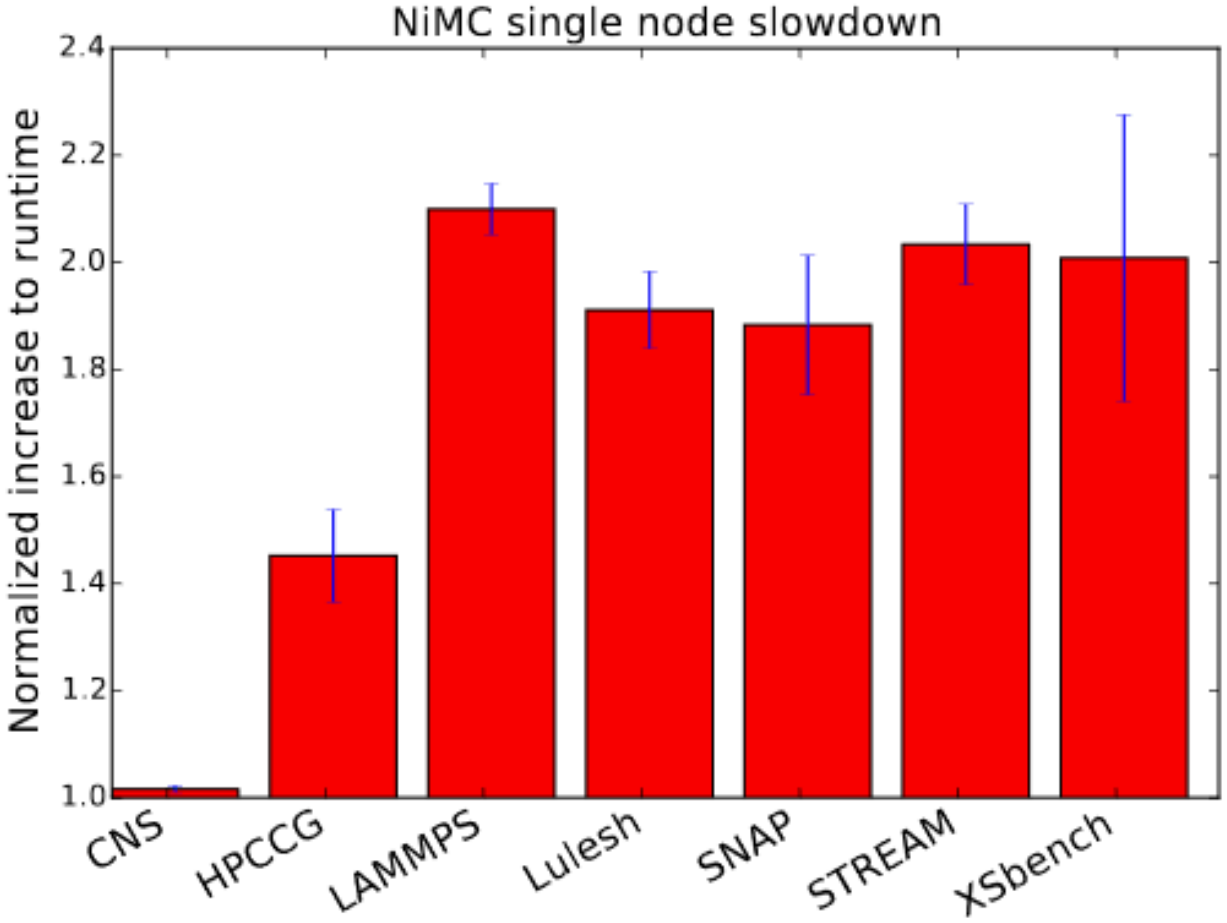


Fig. 3: Normalized impact of NiMC on single node runs.

NiMC hurts every code expect for CNS

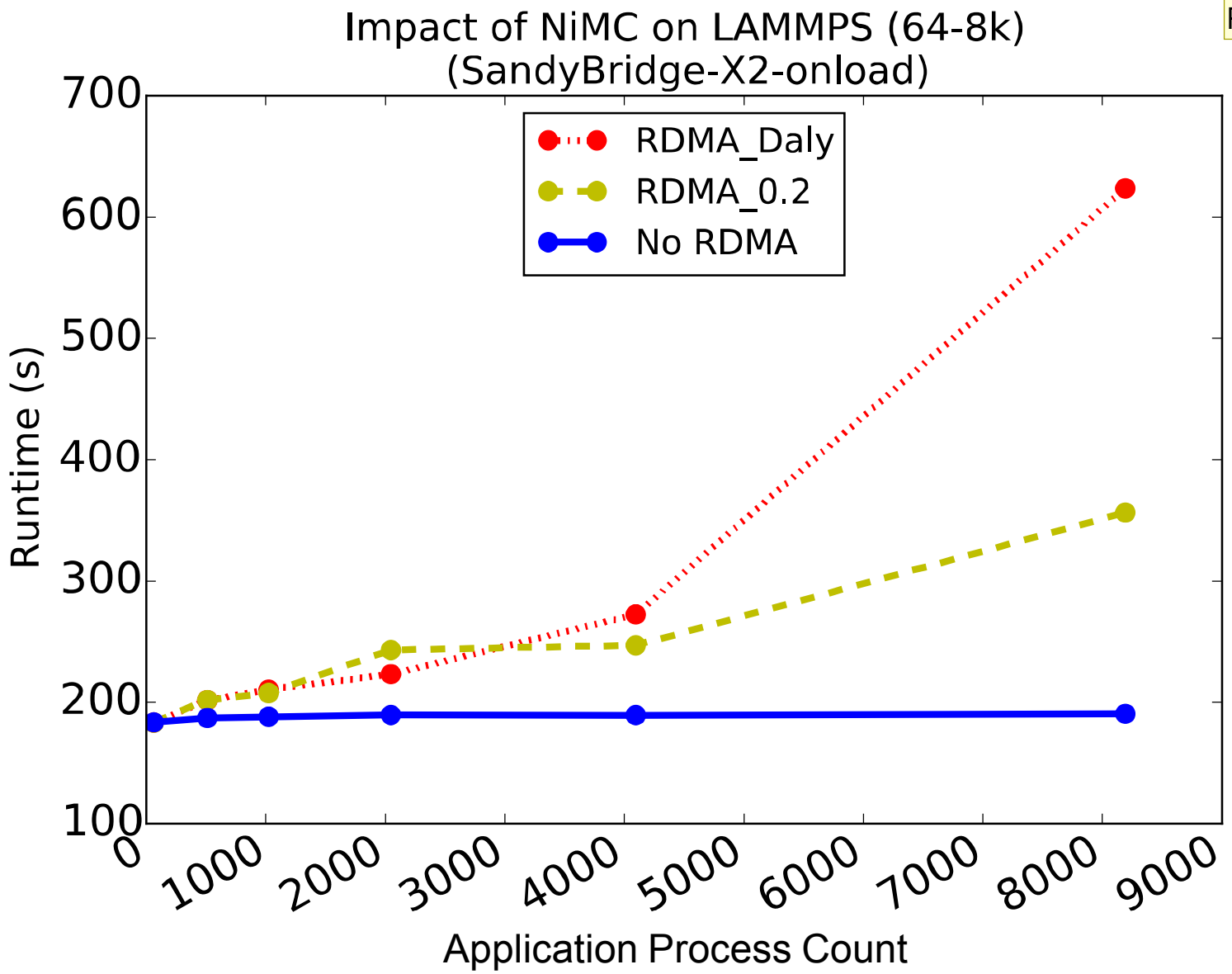
CNS fits into cache comfortably

Two factors in slowdown, main memory bandwidth and cache pollution

Impact at Scale

Sandia National Laboratories

RG1



RG1 Application Process Count

Ryan Grant, 5/17/2016

Evidence of Cache Pollution

- In the **absence** of RDMA writes
  - No real correlation between stalled cycles and any of the cache misses
  - No real correlation between stalled cycles and runtime
- **With** RDMA writes
  - Strong correlation between Stalled Cycles and misses throughout the cache hierarchy
  - Correlation between runtime and L1 Misses becomes larger

TABLE IV: Performance Monitoring Counter Correlations Across All Applications

	Corr. Metric	Stalled Cycle	L1 Miss	L2 Miss	L3 Miss
No RDMA	Time	-0.04	0.941	0.946	0.930
	Stalled Cycles	N/A	0.086	0.030	0.068
RDMA	Time	0.912	0.959	0.978	0.925
	Stalled Cycles	N/A	0.870	0.973	0.997

Can we detect NiMC?

- Yes!
- Ran tests with different feature sets w/ random forest ML
- No one set was best for all apps
- CNS has a bad score, but runtime not impacted from NiMC

App/Benchmark	Set 1 Score	Set 2 Score	Set 3 Score
STREAM-DRAM	1.000	1.000	0.995
STREAM-cache	1.000	1.000	0.990
HPCCG	0.998	0.999	0.999
CNS	0.741	0.747	0.742
LAMMPS	1.000	1.000	1.000

OOB scores for forests predicting the presence of NiMC

NiMC/Machine Learning Conclusion

Sandia National Laboratories

- Each feature set evaluated was able to detect NiMC
  - Feature sets each focused on a level of cache (L1, L2, L3)
- NiMC on onload NICs have far-reaching impact beyond just the local cache, i.e. impact in shared levels
- Furthermore, asynchronous programming models may not provide as much relief as desired
  - Even if we aren't waiting for the slowest process at a synchronization point, imbalance in the system may create bottlenecks for shared resources

NiMC

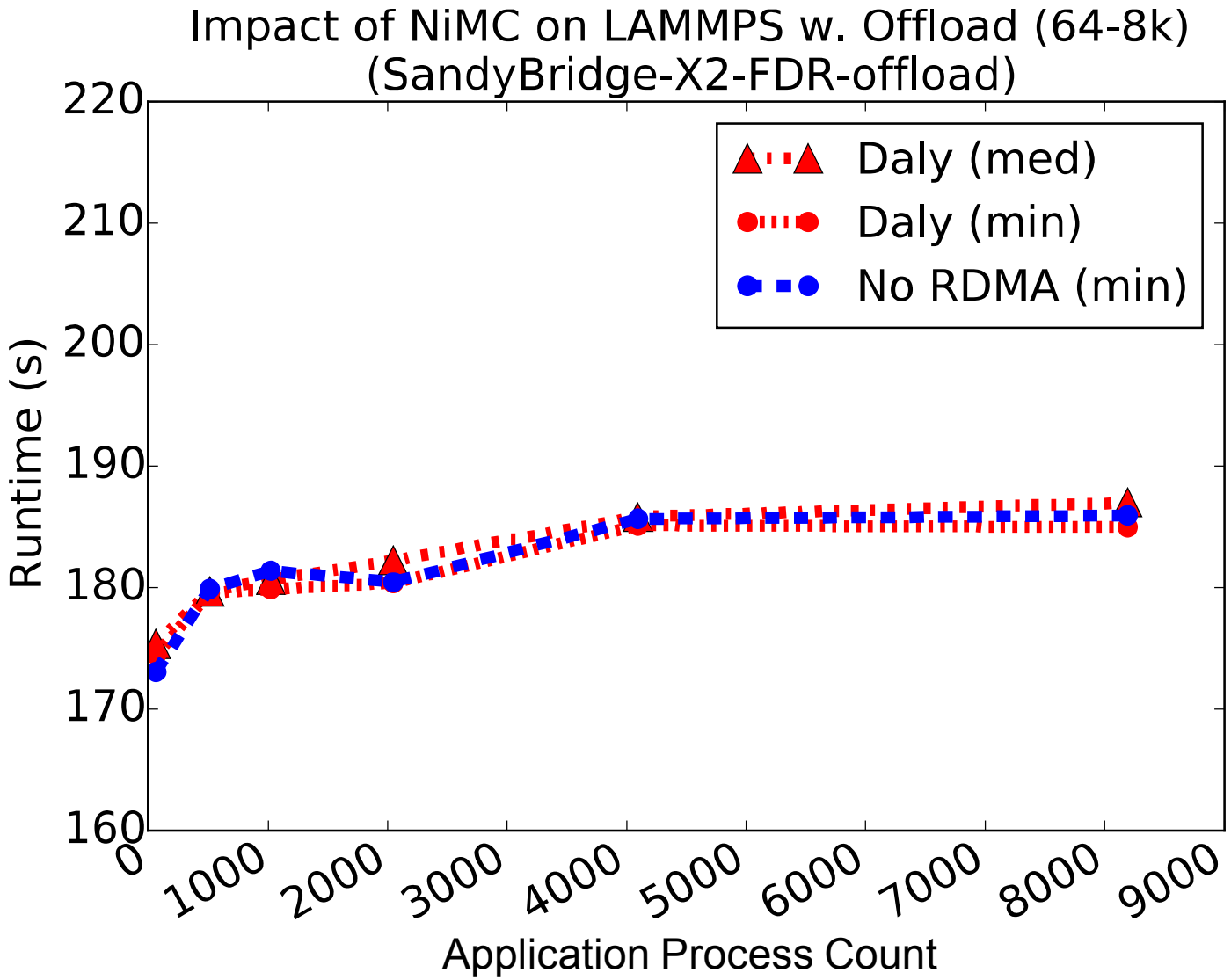
Can we eliminate or mitigate NiMC impact?

Yes.

Offload NIC

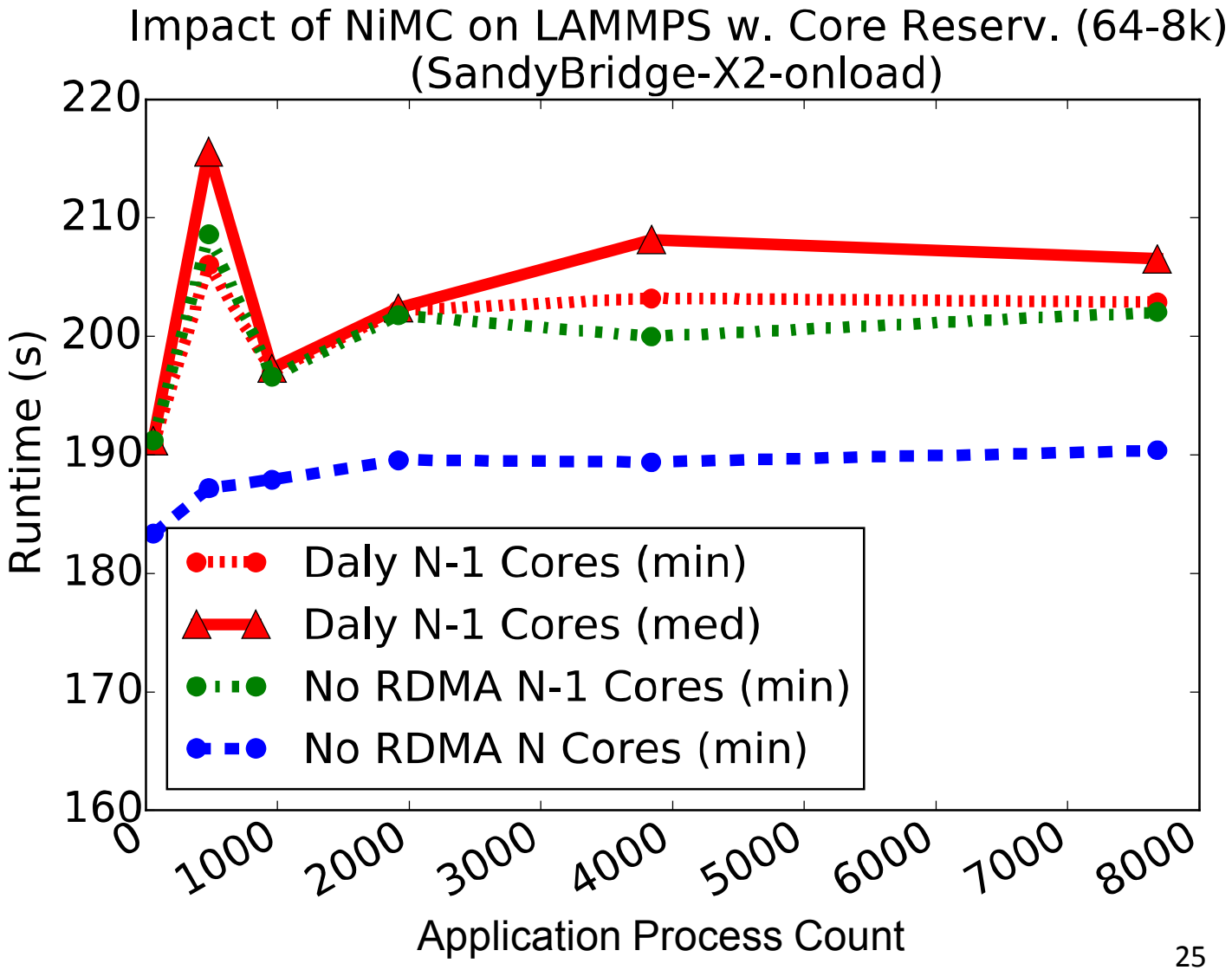
Using offload NICs in system design can help mitigate NiMC

Important to note that in previous generations, memory bandwidth and network balance still see impact on offload NICs



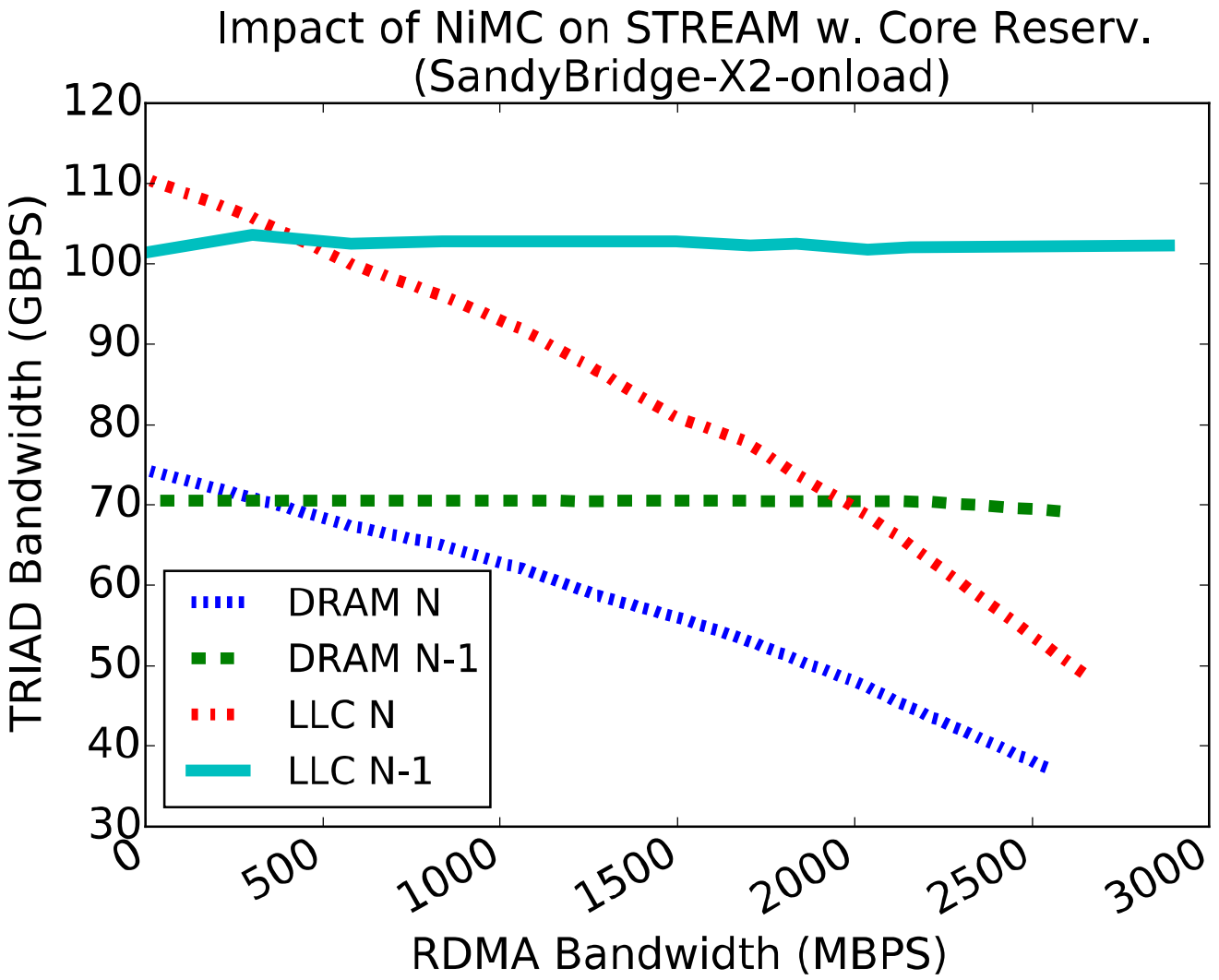
Core Reservation

After a bandwidth threshold, it's better to reserve a core to handle RDMA traffic



Bandwidth Throttling

Under a threshold of ~500 MB/s it is better to just slow traffic at the origin side



Key Takeaways:

- RDMA isn't free:
  - NiMC degraded performance on 6 out of 8 evaluated systems
- NiMC impact depends on architecture + workload:
  - Ranges from no impact to,
  - 3X slowdown in LAMMPS running on an onload system with 8k processes
- We can deal with NiMC, if we are conscious of its impact:
  - Offload NICs (for current CPUs)
  - Network throttling
  - Core reservation

Conclusions

- Multiple factors make network performance prediction difficult but possible to achieve (not 100% of the time)
- Overtime tool available for others to use
  - Part of the Sandia Microbenchmarks
  - http://www.cs.sandia.gov/smb/
- Use for:
  - Assessing job placement for fulfilling networking requirements
  - Composing with application variation studies to understand networking variation independently
  - Studying network interference on other networks (e.g. Aries)
  - Evaluating periods of network variability on other systems

Thank you

Sandia National Laboratories

Questions?

Acknowledgments:

This work was funded through the Computational Systems and Software Environment sub-program of the Advanced Simulation and Computing Program funded by the National Nuclear Security Administration