

The Case of the Missing Supercomputer Performance:

Achieving Optimal Performance on the 8,192 Processors of ACSI Q

5/26/11
Danni Als

Goal

- The way to improve the effective performance of ASCI Q (the world's second-fastest supercomputer)
- Detected performance problems
- Problem elimination
- Demonstrating a factor 2 improvement

Introduction

- ASCI Q machine installed at Los Alamos National Laboratory (LANL)
- 8,192 processors
- 20 Tflop/s
- Performance gain but is it big enough?

Test Enviroment

SAGE:

- Compressable Eulerian hydrodynamics code
- Is a multi-dimensional (1D, 2D, and 3D), multi-material, massively parallel, Eulerian hydrodynamics code for use in solving the Euler equation
- 150.00 of Fortran + MPI code.

ASCI Q:

- Consist of 2.048 HP ES45 Alpha Server SMP nodes
- Each containing four EV68 Alpha processors
- Interconnected with Quadrics QsNet network
- Installed in steps

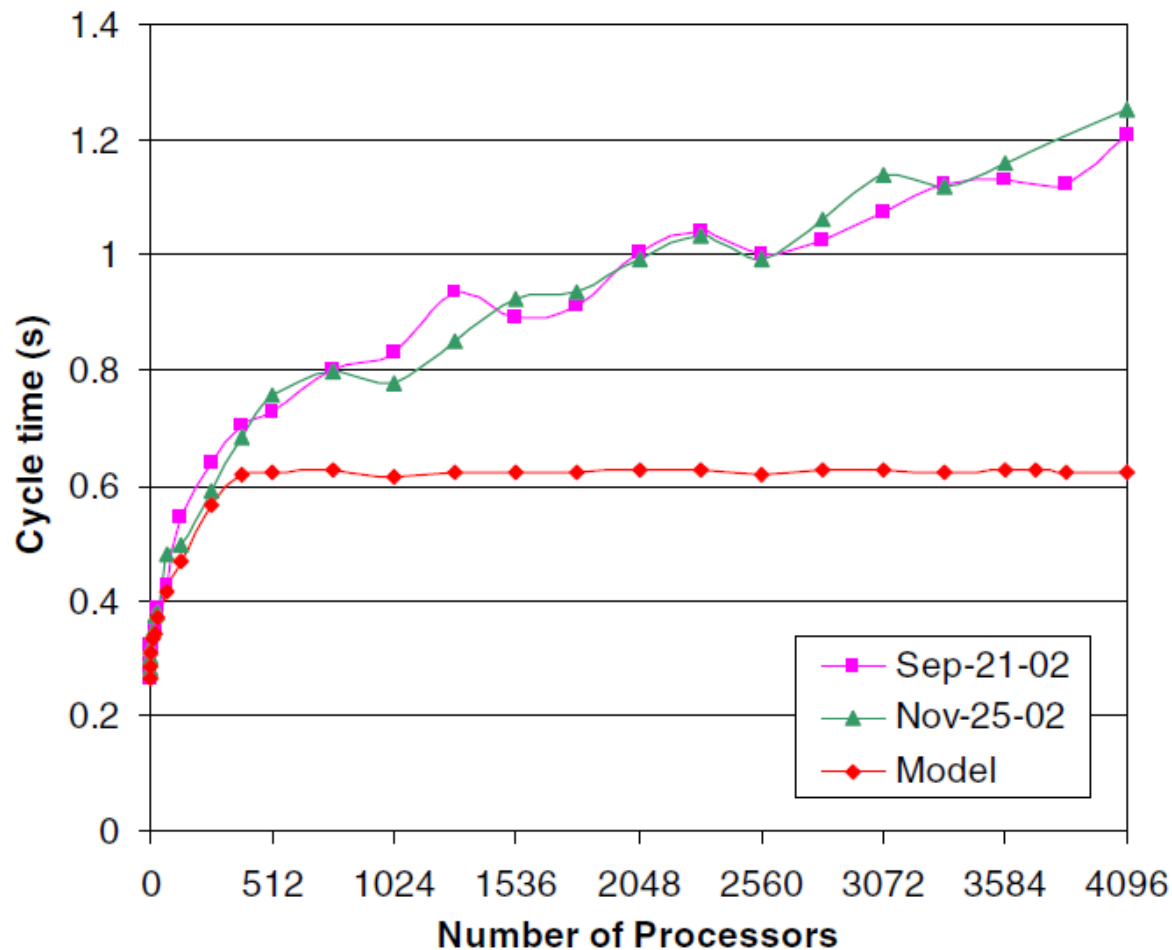
Things to be aware of

- Weak scaling
- Strong scaling
- Memory hierarchies

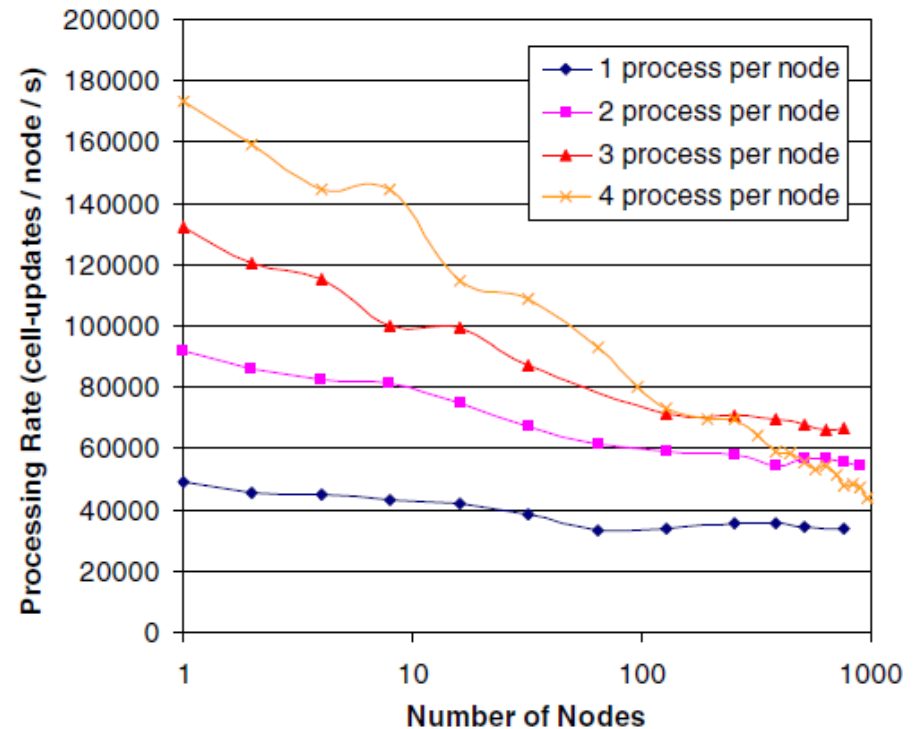
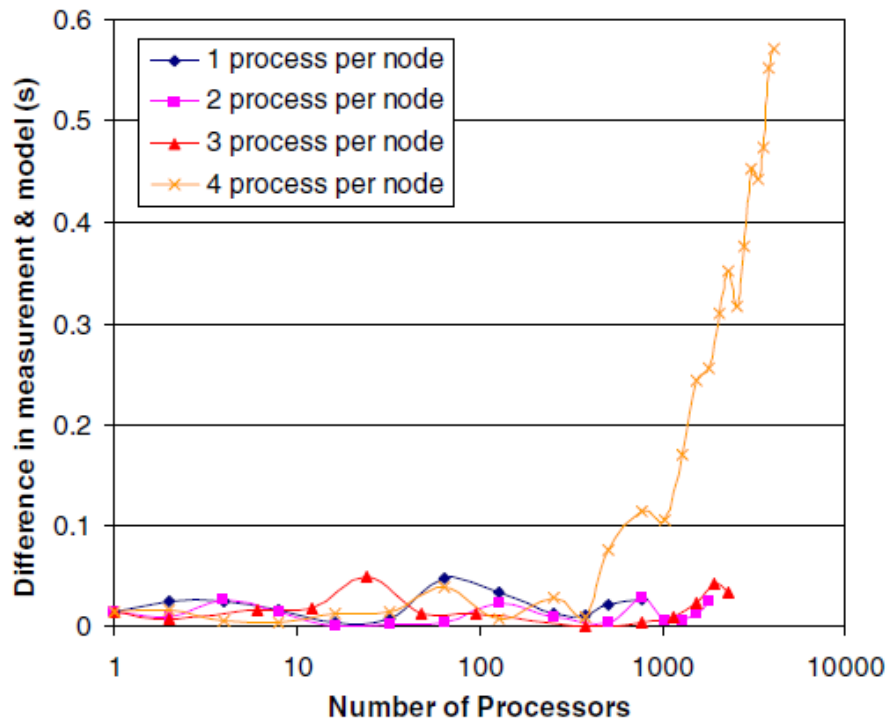
Their approach

1. Detailed information about both application and system and use performance modeling to determine performance.
2. If measured performance is less then determine source.
3. Eliminate the cause of the suboptimal performance.
4. Repeat from step 2 until expected result is achieved.

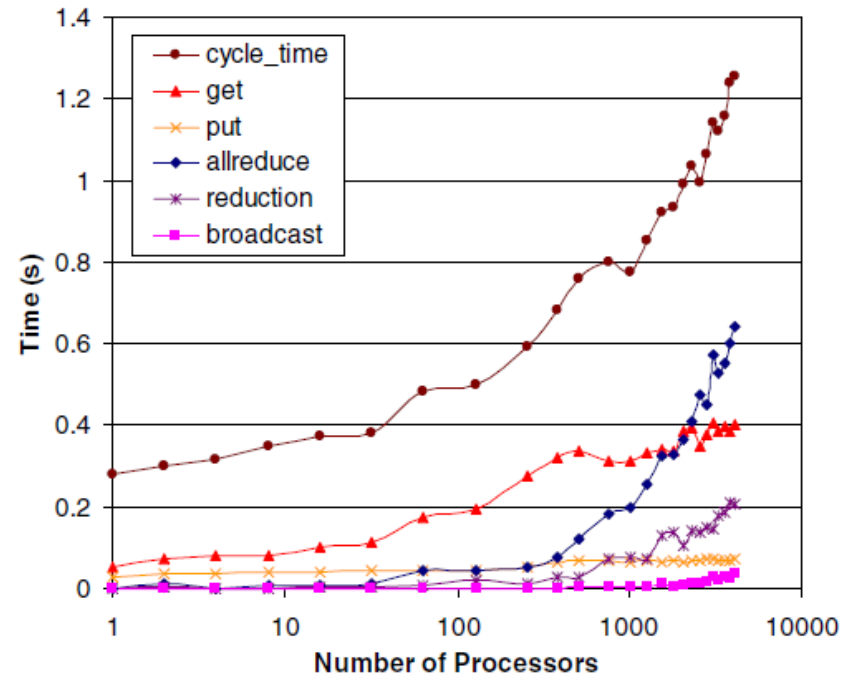
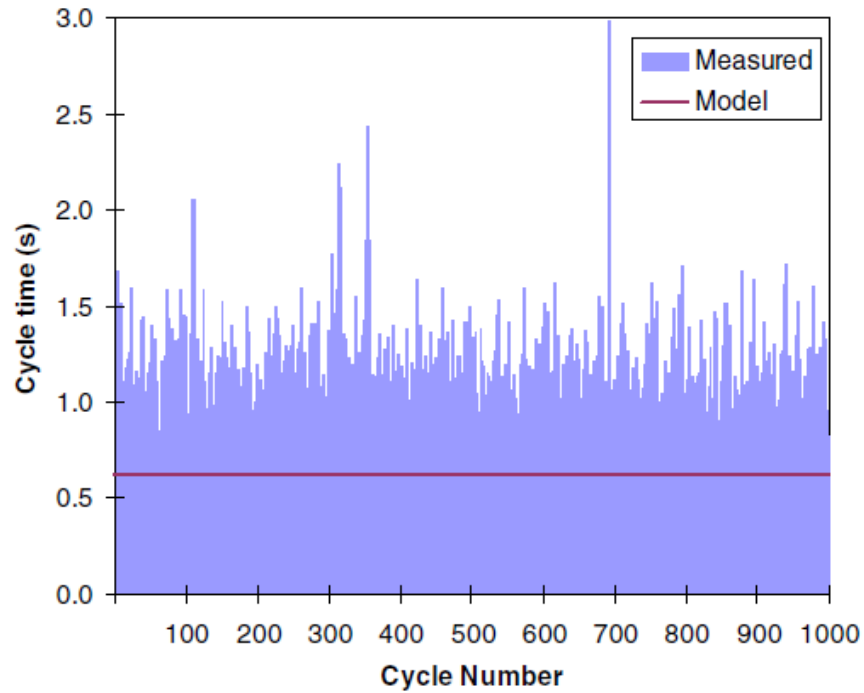
First SAGE performance measurements



Identify the problem



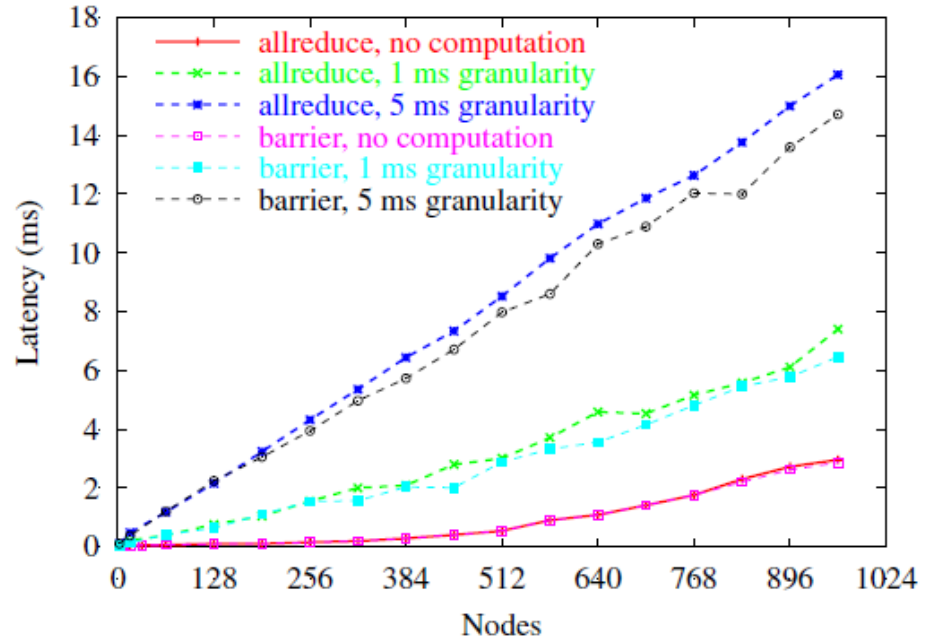
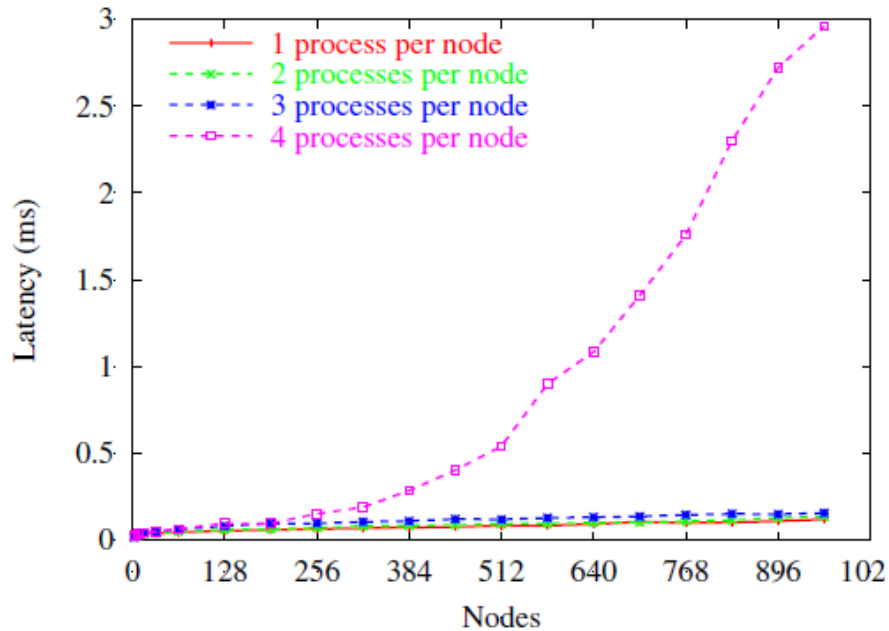
Identify the Problem



Problem Conclusion

- Performance difference occurs only when using all four processors per node
- The performance changes alot from cycle to cycle
- It appears to originate from allreduce and reduction operations

Allreduce optimization



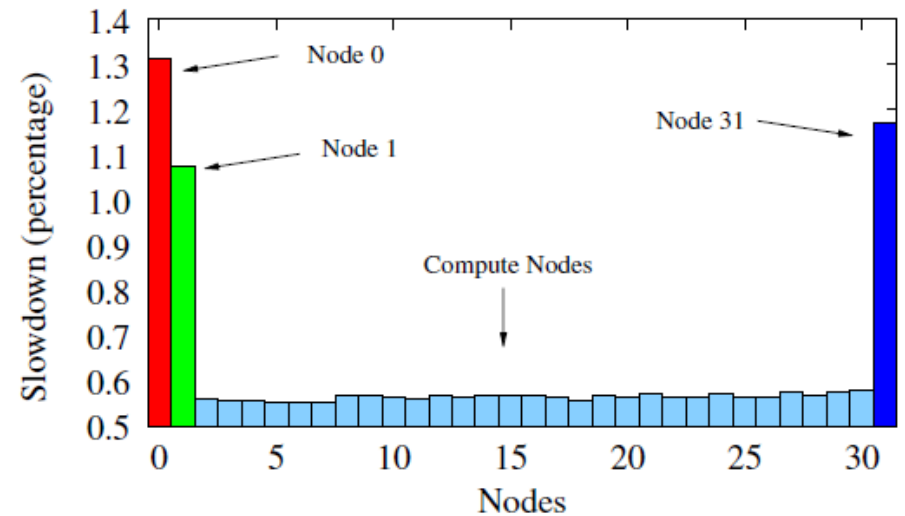
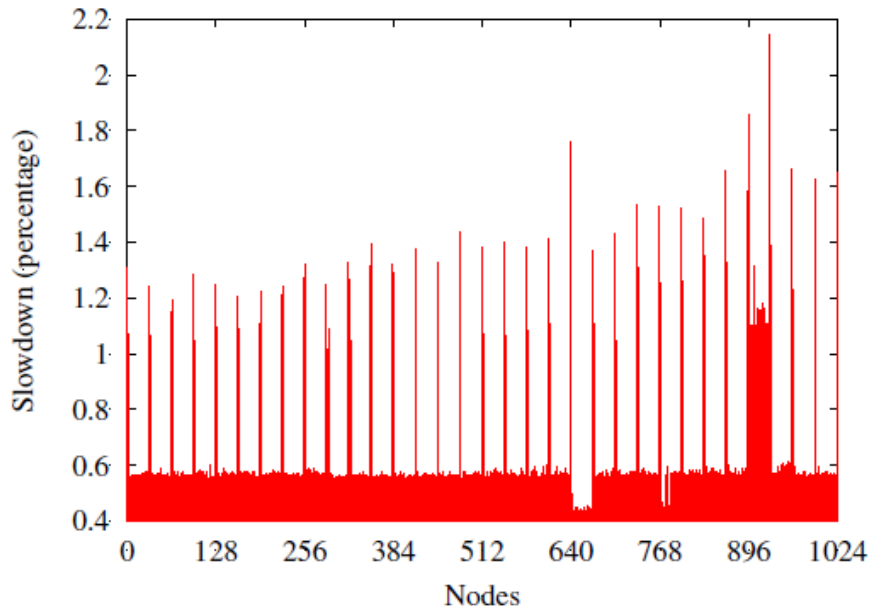
Allreduce Conclusion

- Neither MPI nor the network are responsible for performance problems.
- Performance loss is in the nodes themselves.

Computational Noise

- System activity can run without interfering with the application as long as there is a spare processor available in each node to absorb them
- No spare processor, a processor is temporarily taken from the application to handle system activity

Computational noise



Computational Noise

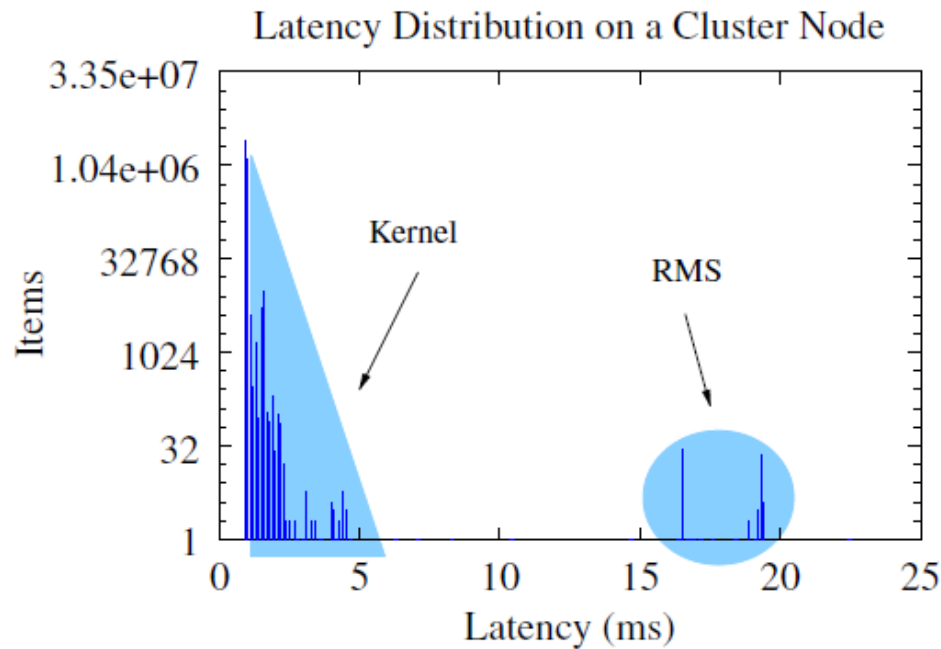
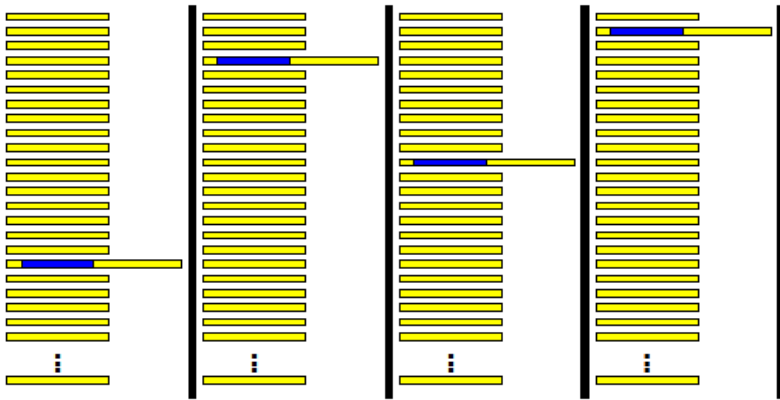
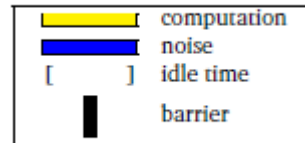


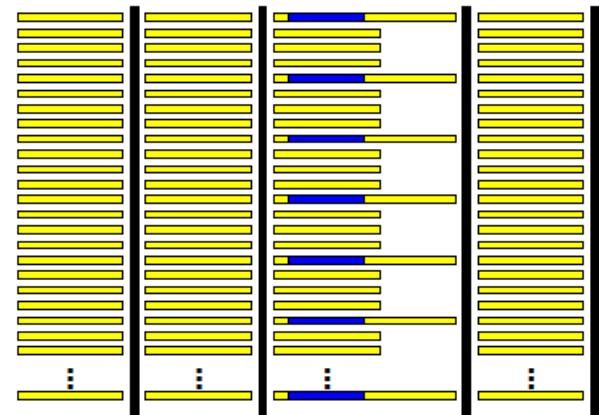
TABLE 2: Summary of noise on each 32-node cluster

Source of noise	Duration (ms)	Location (nodes)			
		0	1	2-30	31
Kernel	0-3	✓	✓	✓	✓
RMS dæmons	5-18	✓	✓	✓	✓
TruCluster dæmons	>18	✓	✓		

Noise Synchronization

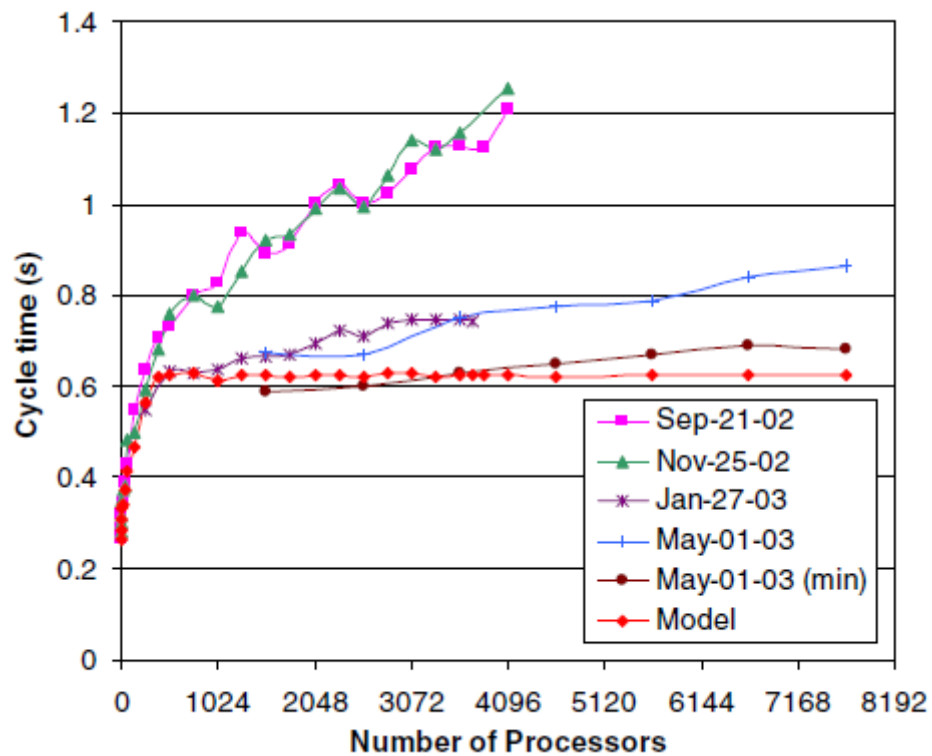


(a) Uncoordinated noise

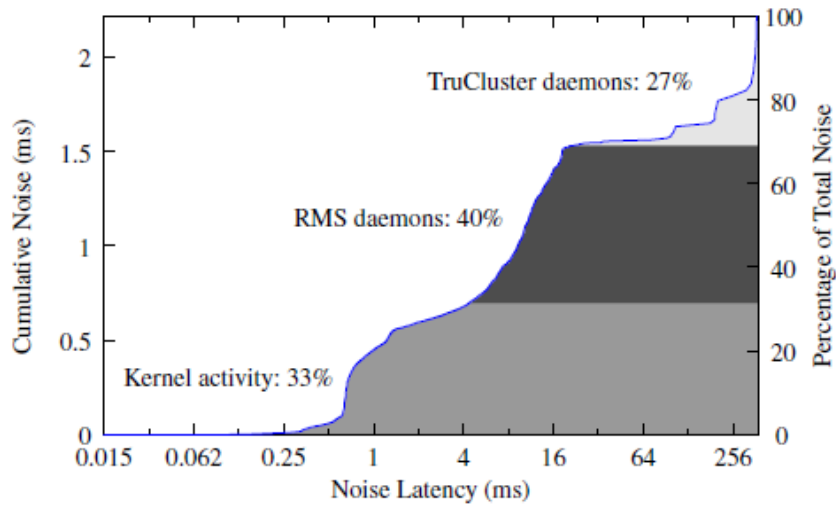


(b) Coscheduled noise

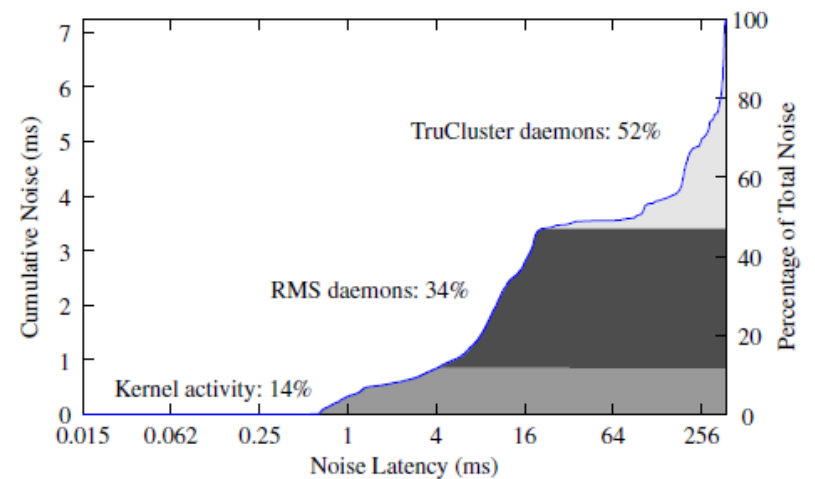
SAGE Performance



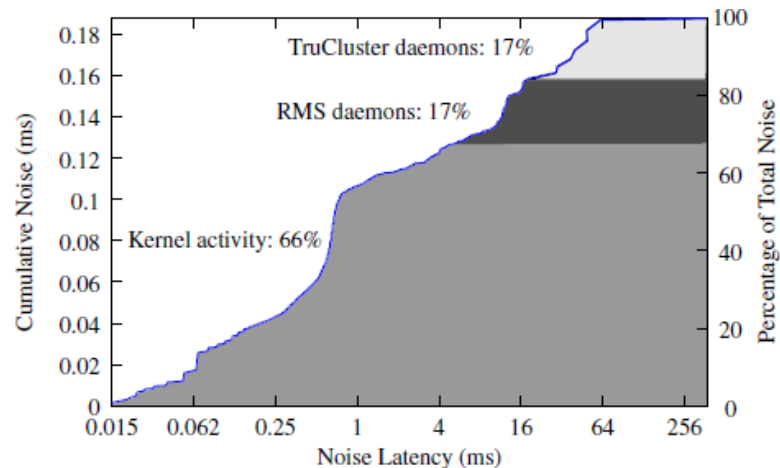
Barrier experiments



(b) 1 ms of intervening computation



(c) 5 ms of intervening computation



(a) No intervening computation

Conclusion

- Created performance-analysis methodology.
 - Analyzes
 - Identifies all sources of noise and categorizes them.
 - Determines the total impact the noise has on the application and which sources of noise has the greatest impact.
- Managed to double the performance of SAGE
- First to determine how fast an application can potentially run.

References

The RAGE radiation-hydrodynamic code

**Michael Gittings¹, Robert Weaver², Michael Clover¹,
Thomas Betlach¹, Nelson Byrne¹, Robert Coker²,
Edward Dendy², Robert Hueckstaedt², Kim New²,
W Rob Oakes², Dale Ranta¹ and Ryan Stefan³**

¹ Science Applications International Corp. MS A-1, 10260 Campus Point Drive,

San Diego, CA 92121, USA

² Los Alamos National Laboratory, MS T087, PO Box 1663, Los Alamos, NM 87545, USA

³ TaylorMade-adidas Golf, 5545 Fermi Court, Carlsbad, CA 92008-7324, USA

E-mail: michael.r.clover@saic.com

Received 27 December 2007, in final form 29 January 2008

Published 20 November 2008

Computational Science & Discovery **1**

QsNet

- High speed interconnect designed by Quadrics
- Communication API is normally MPI
- Interconnect consist of a PCI card connected with cober cables
- Quadrics was formed in 1996