Cray X1 Evaluation: Overview and Scalability Analysis

Patrick H. Worley

Oak Ridge National Laboratory
(ORNL)

February 26, 2004
SIAM PP04
Hyatt at Fisherman's Wharf
San Francisco, California

Acknowledgements

- Research sponsored by the Atmospheric and Climate Research Division and the Office of Mathematical, Information, and Computational Sciences, Office of Science, U.S. Department of Energy under Contract No. DE-AC05-00OR22725 with UT-Battelle, LLC.
- These slides have been authored by a contractor of the U.S. Government under contract No. DE-AC05-00OR22725. Accordingly, the U.S. Government retains a nonexclusive, royalty-free license to publish or reproduce the published form of this contribution, or allow others to do so, for U.S. Government purposes
- Oak Ridge National Laboratory is managed by UT-Battelle, LLC for the United States Department of Energy under Contract No. DE-AC05-00OR22725.

Collaborators

In addition to the explicit acknowledgements in the slides, this presentation includes contributions from the following individuals:

- Thomas H. Dunigan, Jr., ORNL
- Mark R. Fahey, ORNL
- John Levesque, Cray Inc.
- James B. White III, ORNL

We also thankfully acknowledge the assistance of the staff of the Center for Computational Sciences at ORNL.

Phoenix

Cray X1 with 64 SMP nodes

- 4 Multi-Streaming Processors (MSP) per node
- 4 Single Streaming Processors (SSP) per MSP
- Two 32-stage 64-bit wide vector units running at 800 MHz and one 2-way superscalar unit running at 400 MHz per SSP
- 2 MB Ecache per MSP
- 16 GB of memory per node

for a total of 256 processors (MSPs), 1024 GB of memory , and 3200 GF/s peak performance.

Other Platforms

- Earth Simulator: 640 8-way vector SMP nodes and a 640x640 single-stage crossbar interconnect. Each processor has 8 64-bit floating point vector units running at 500 MHz.
- HP/Compaq AlphaServer SC at Pittsburgh Supercomputing Center: 750 ES45 4-way SMP nodes (1GHz Alpha EV68) and a Quadrics QsNet interconnect with two network adapters per node.
- IBM p690 cluster at ORNL: 27 32-way p690 SMP nodes (1.3 GHz POWER4) and an SP Switch2 with two to eight network adapters per node.
- IBM SP at the National Energy Research Supercomputer Center (NERSC): 184 Nighthawk II 16-way SMP nodes (375MHz POWER3-II) and an SP Switch2 with two network adapters per node.
- SGI Altix 3700 at ORNL: 2 128-way SMP nodes and NUMAflex fat-tree interconnect. Each processor is a 1.5 GHz Itanium 2 with a 6 MB L3 cache

Background Material

Papers

Cray X1 Evaluation Status Report, ORNL/TM-2004/13

Cray X1 Evaluation, ORNL/TM-2003/67

Early Evaluation of the Cray X1, SC03

Talks

Early Evaluation of the Cray X1, SC03

Early Evaluation of the Cray X1 - Part 1.5, SC03, Cray Exhibit Booth

Latest Performance Results from ORNL: Cray X1 and SGI Altix, LACSI 2003

URLs

http://www.csm.ornl.gov/evaluation/PHOENIX

http://www.csm.ornl.gov/~dunigan/cray

Evaluation Process

Standard ORNL Early Evaluation Framework

Fast, Fair, Open, Hierarchical, Application-Oriented

Input and Participation from the Performance Community, including

- Argonne National Laboratory (ANL)
- Lawrence Berkeley National Laboratory (LBNL)
- Pacific Northwest National Laboratory (PNNL)
- University of Tennessee

Input and Participation from Application Communities

- Biology Workshop
- Climate Modeling Workshop
- Fusion Workshop
- Computational Materials Workshop
- X1 tutorials and hands-on workshops

Evaluation Goals

- Evaluate benchmark and application performance and compare with systems from other HPC vendors.
- Determine the most effective approaches to using the Cray X1.
- Evaluate system and system administration software reliability and performance.
- Predict scalability, both in terms of problem size and number of processors.
- Collaborate with Cray to incorporate this information and experience into their next generation designs.

Evaluation Methodology

Hierarchical evaluation

- Microbenchmarks
- Application-relevant kernels
- Compact or full parallel application codes

Open evaluation

- Rapid dissemination of evaluation results
- Systems available to external performance researchers

Fair evaluation

- Determining appropriate ways of using system, evaluating both traditional and alternative programming paradigms
- Collecting data with both standard and custom benchmarks

X1 Evaluation Complexities

Vectorization

- Timings from unoptimized/unanalyzed benchmarks are not illuminating.
- Porting and optimization efforts can be time-consuming.
- Successes (achieving good performance) are easy to identify. Failures are more difficult. When do you stop trying? Is it a mismatch with the architecture, a mismatch with the coding style, or simply a lack of understanding by the performance engineer? What code modifications are acceptable?

Large Optimization Search Space

- The X1 introduces a new level of parallelism (streams) that can be exploited by users (in addition to vectorization, OpenMP, and MPI).
- The X1 also supports globally addressable memory paradigms such as SHMEM, Co-Array Fortran, UPC, and MPI-2 one-sided.

Scalability Analysis

Latency and Bandwidth

- HALO benchmarks
- MPI SWAP benchmarks

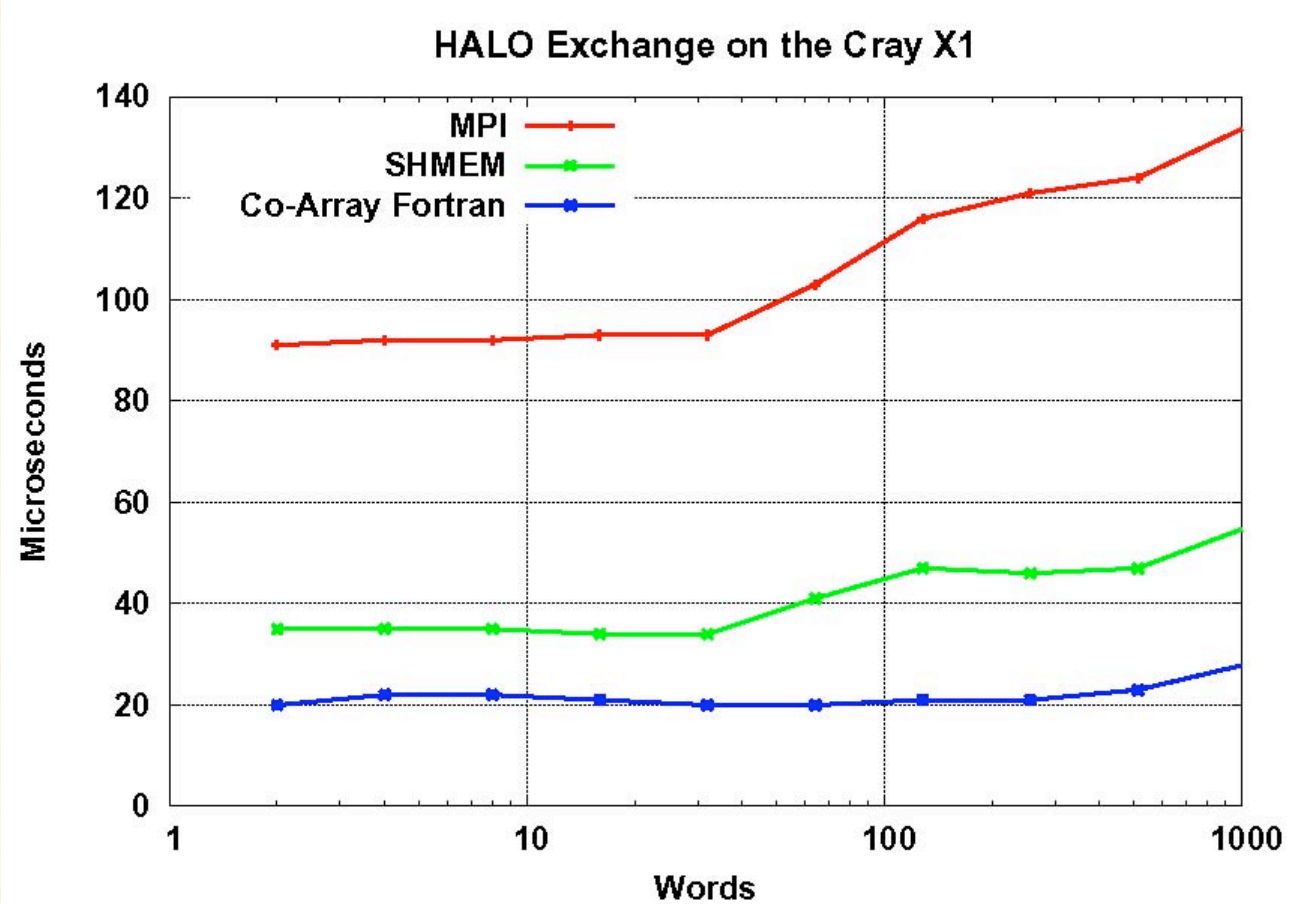
Application Benchmarks

- POP ocean code (latency sensitive)
- GYRO fusion code (bandwidth sensitive)

The application benchmark problems were chosen to investigate scaling issues. See the evaluation web site for performance results for production-sized problems on the current system.

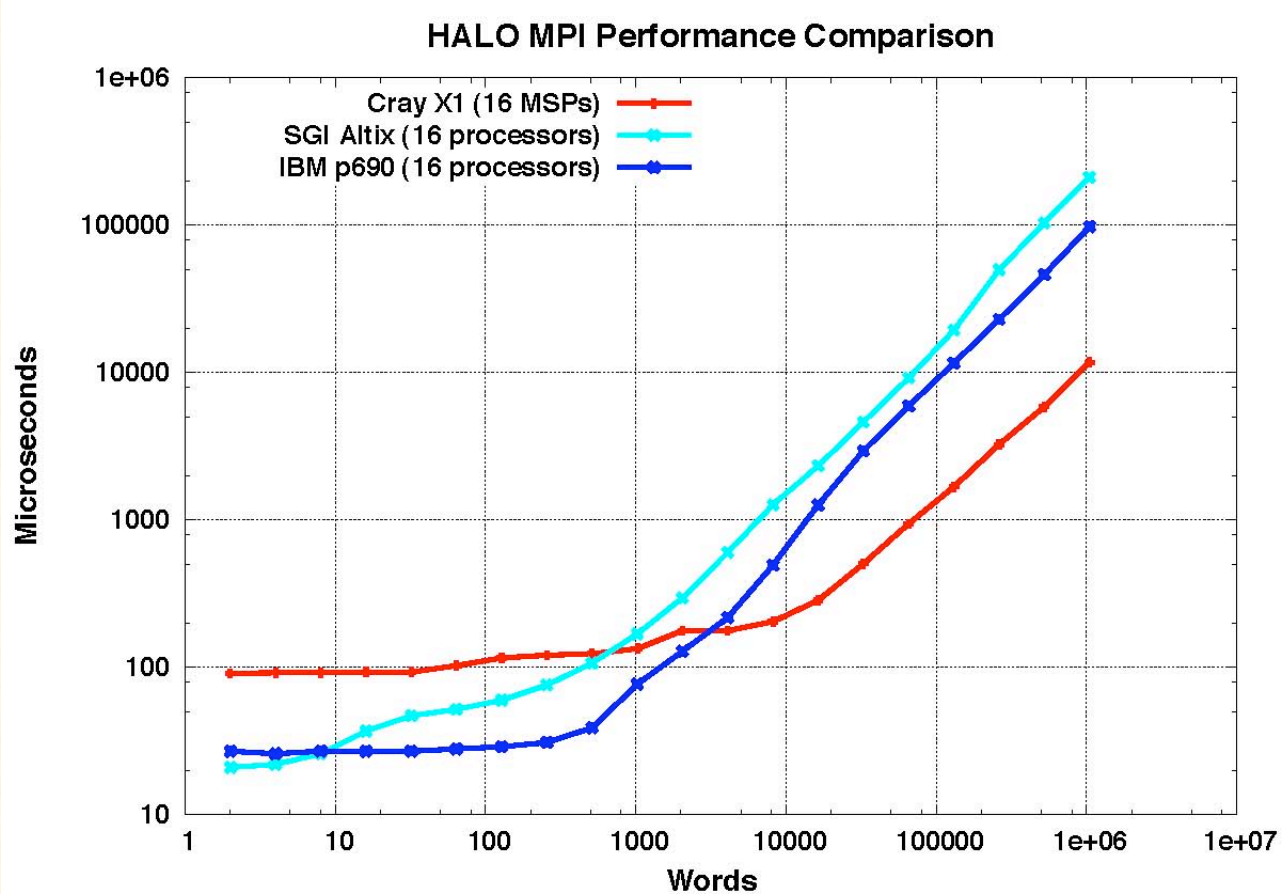
HALO Exchange Paradigm Comparison

Comparing performance of MPI, SHMEM, and Co-Array Fortran implementation of Allan Wallcraft's HALO benchmark on 16 MSPs. SHMEM and Co-Array Fortran are substantial performance enhancers for this benchmark.



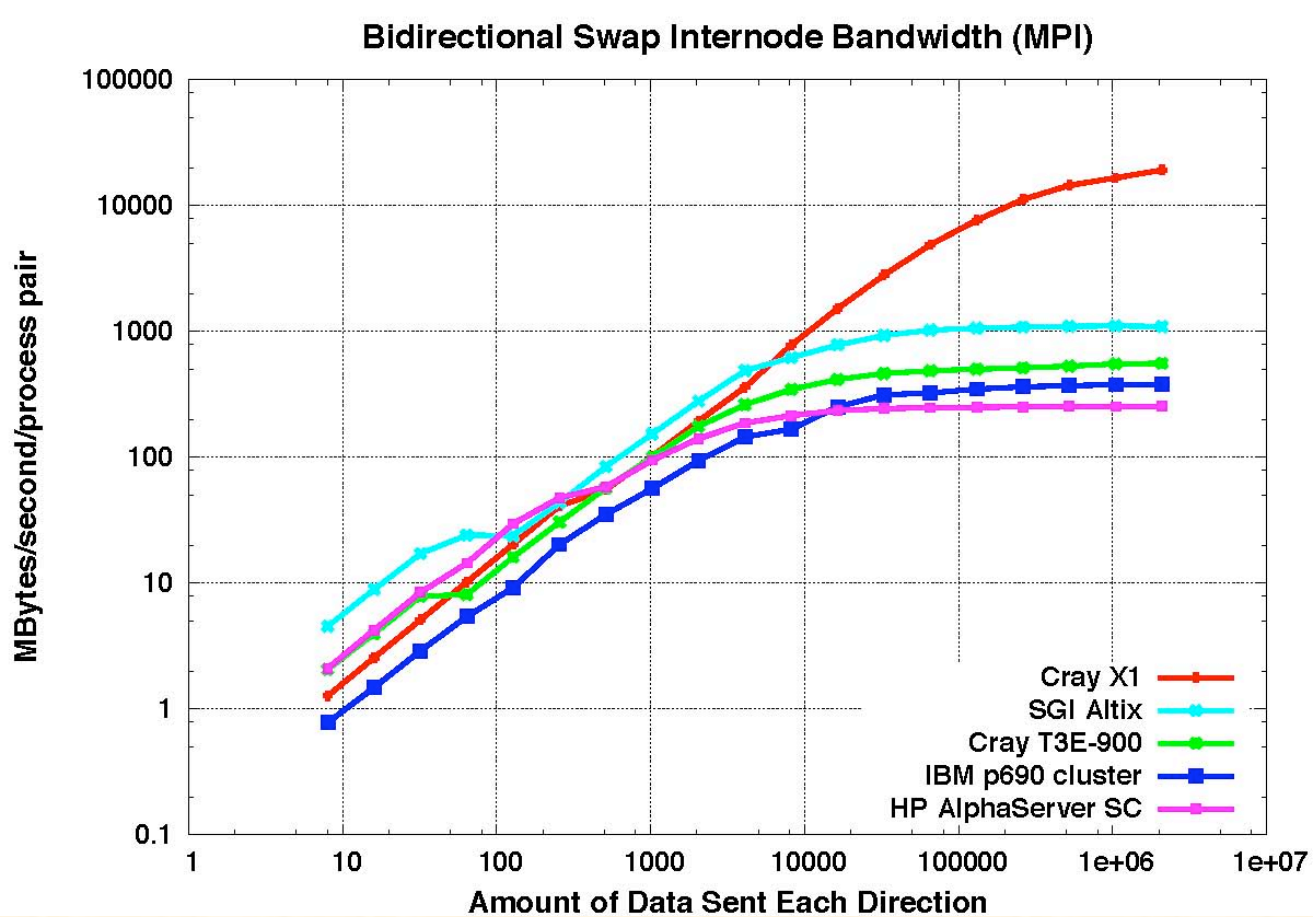
HALO Exchange Platform Comparisons

Comparing HALO performance using MPI on 16 MSPs of the Cray X1 and 16 processors of the IBM p690 (within a 32 processor SMP) and the SGI Altix (within a 128 processor SMP). Achievable bandwidth is much higher on the X1. For small halos, the p690 and Altix MPI HALO performance is between the X1 SHMEM and Co-Array Fortran HALO performance.



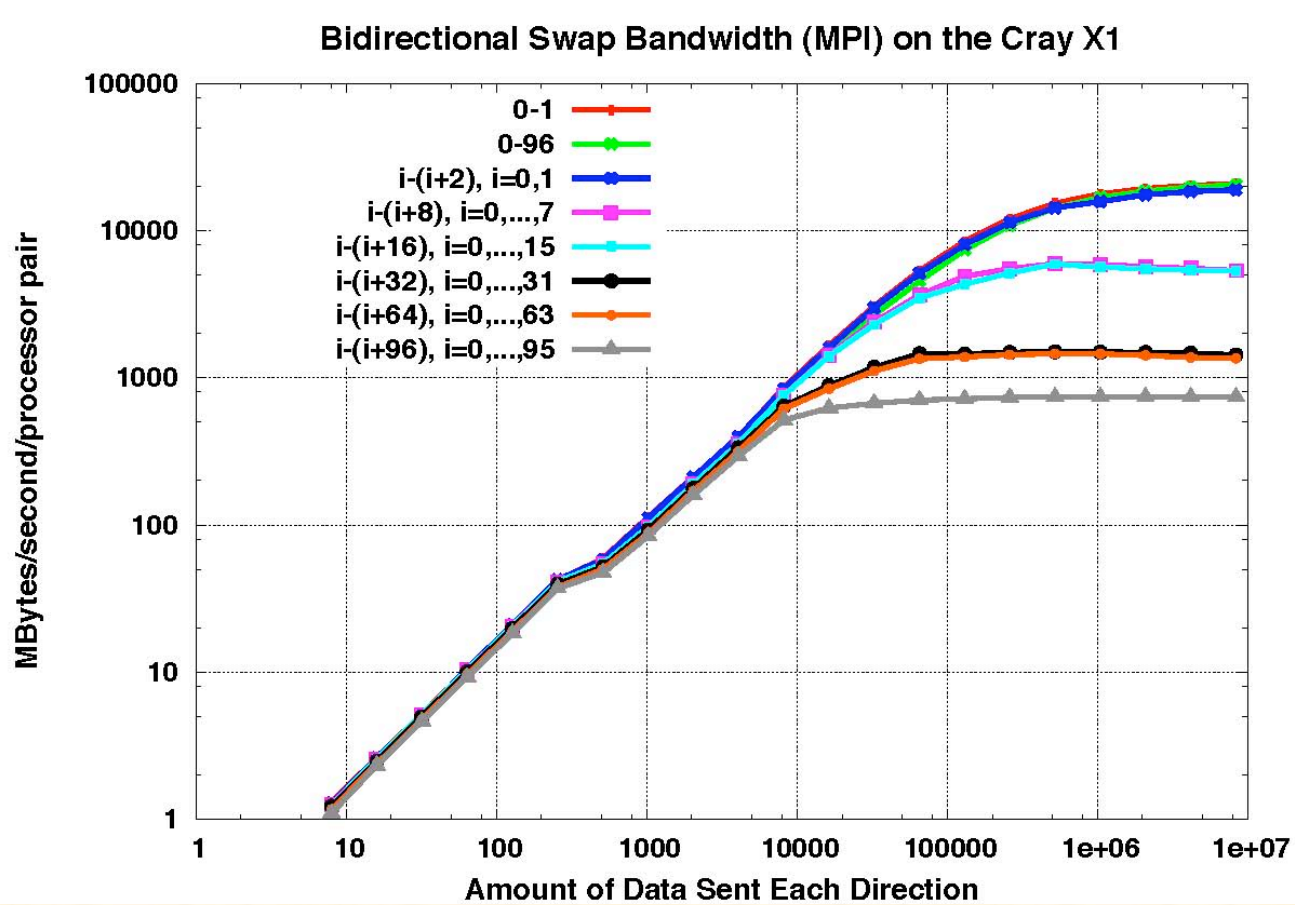
MPI SWAP Platform Comparisons

Comparing performance of SWAP for different platforms. Experiment measures bidirectional bandwidth between two processors in different SMP nodes. For the Altix we used processors 96 and 192. The significant advantage in bandwidth on the X1 (approx. 20X better than on the Altix) is important for a number of important applications.



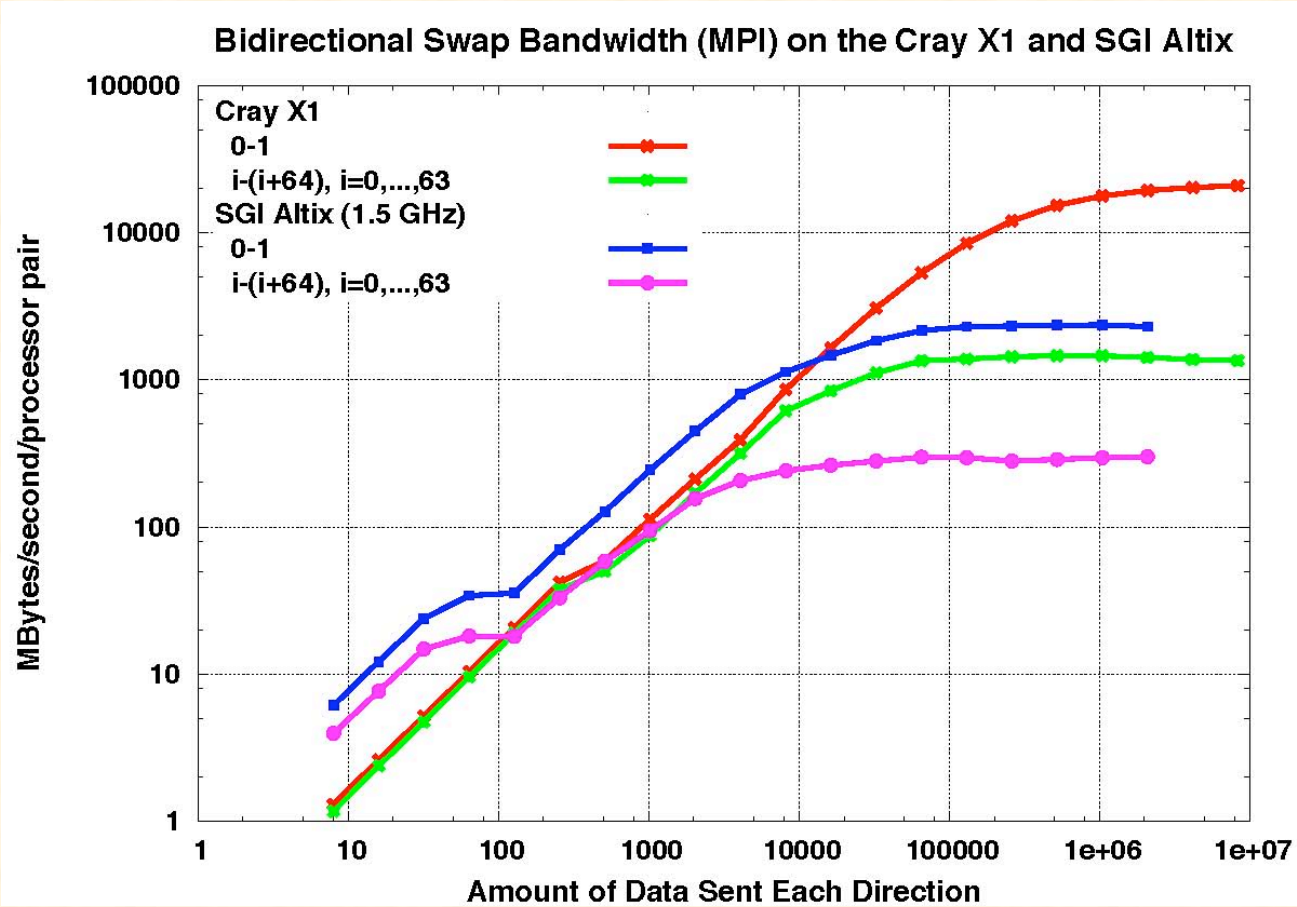
MPI SWAP Contention Evaluation

Comparing performance of SWAP for different communication patterns. Each experiment measures the per processor pair bandwidth when some number of pairs are swapping data simultaneously. For example, i-(i+8),i=0,..,7 means that processor 0 is swapping data with processor 8, processor 1 with processor 9, etc.



MPI SWAP Contention Evaluation

Comparing performance of SWAP for different communication patterns on the X1 and the Altix. Contention also degrades performance on the Altix, and the X1 retains its advantage.



Latency and Bandwidth Summary

- While MPI latency is mediocre at the current time, latency for Co-Array Fortran or SHMEM is excellent, and we expect that MPI performance will improve.
- MPI Bandwidth is excellent for individual processor pairs. Performance degradation due to contention is well-behaved and is significant for the examined communication pattern only for messages > 8KBytes on the current system.
- MPI latency and bandwidth are relatively insensitive to distance between communicating processors on the current system.

Parallel Ocean Program (POP)

- Developed at Los Alamos National Laboratory. Used for high resolution studies and as the ocean component in the Community Climate System Model (CCSM)
- Ported to the Earth Simulator by Dr. Yoshikatsu Yoshida of the Central Research Institute of Electric Power Industry (CRIEPI).
- Initial port to the Cray X1 by John Levesque of Cray, using Co-Array Fortran for conjugate gradient solver.
- X1 and Earth Simulator ports merged and modified by Pat Worley and Trey White of Oak Ridge National Laboratory.
- Optimization on the X1 ongoing.

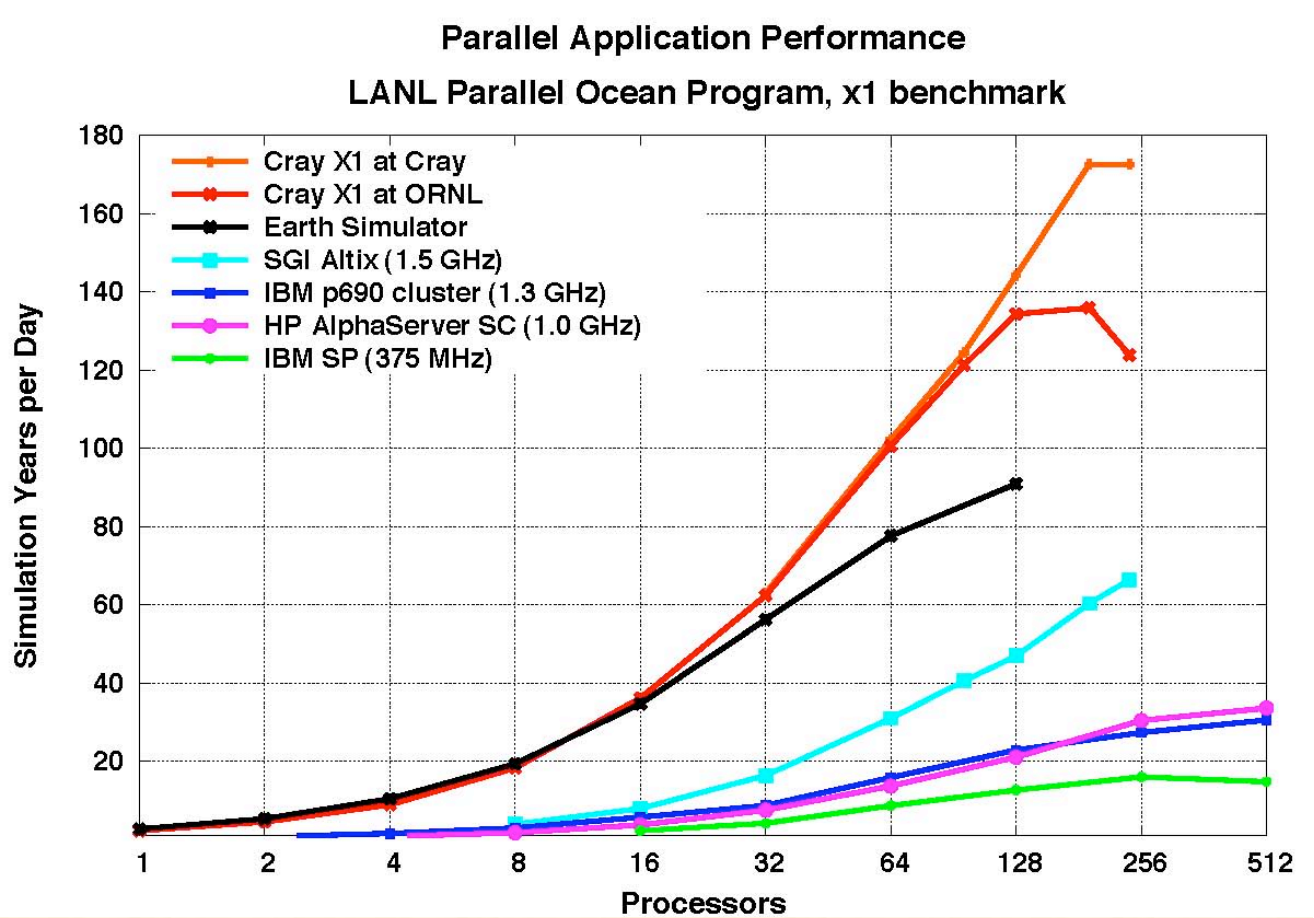
POP Experiment Particulars

- Two primary computational phases
  - Baroclinic: 3D with limited nearest-neighbor communication; scales well.
  - Barotropic: dominated by solution of 2D implicit system using conjugate gradient solves; scales poorly.
- One fixed size benchmark problem
  - One degree horizontal grid (“by one” or “x1”) of size 320x384x40.
- Domain decomposition determined by grid size and 2D virtual processor grid. Results for a given processor count are the best observed over all applicable processor grids.

POP Simulation Rate

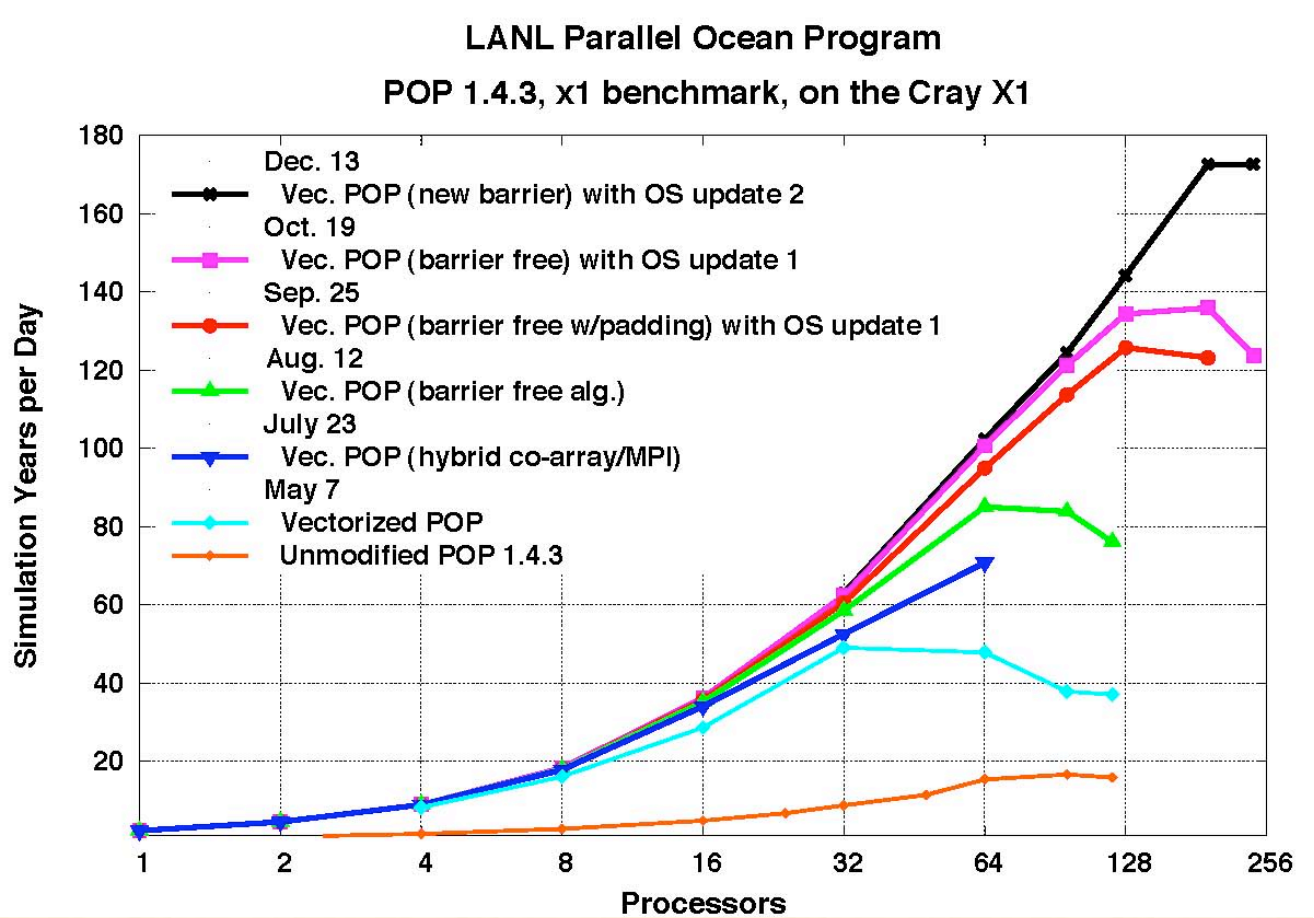
Comparing performance and scaling across platforms.

- Earth Simulator results courtesy of Dr. Y. Yoshida of the Central Research Institute of Electric Power Industry
- IBM SP results courtesy of Dr. T. Mohan of Lawrence Berkeley National Laboratory
- X1 at Cray results courtesy of J. Levesque of Cray, Inc. Cray system using beta versions of system software.



POP Performance Evolution on the X1

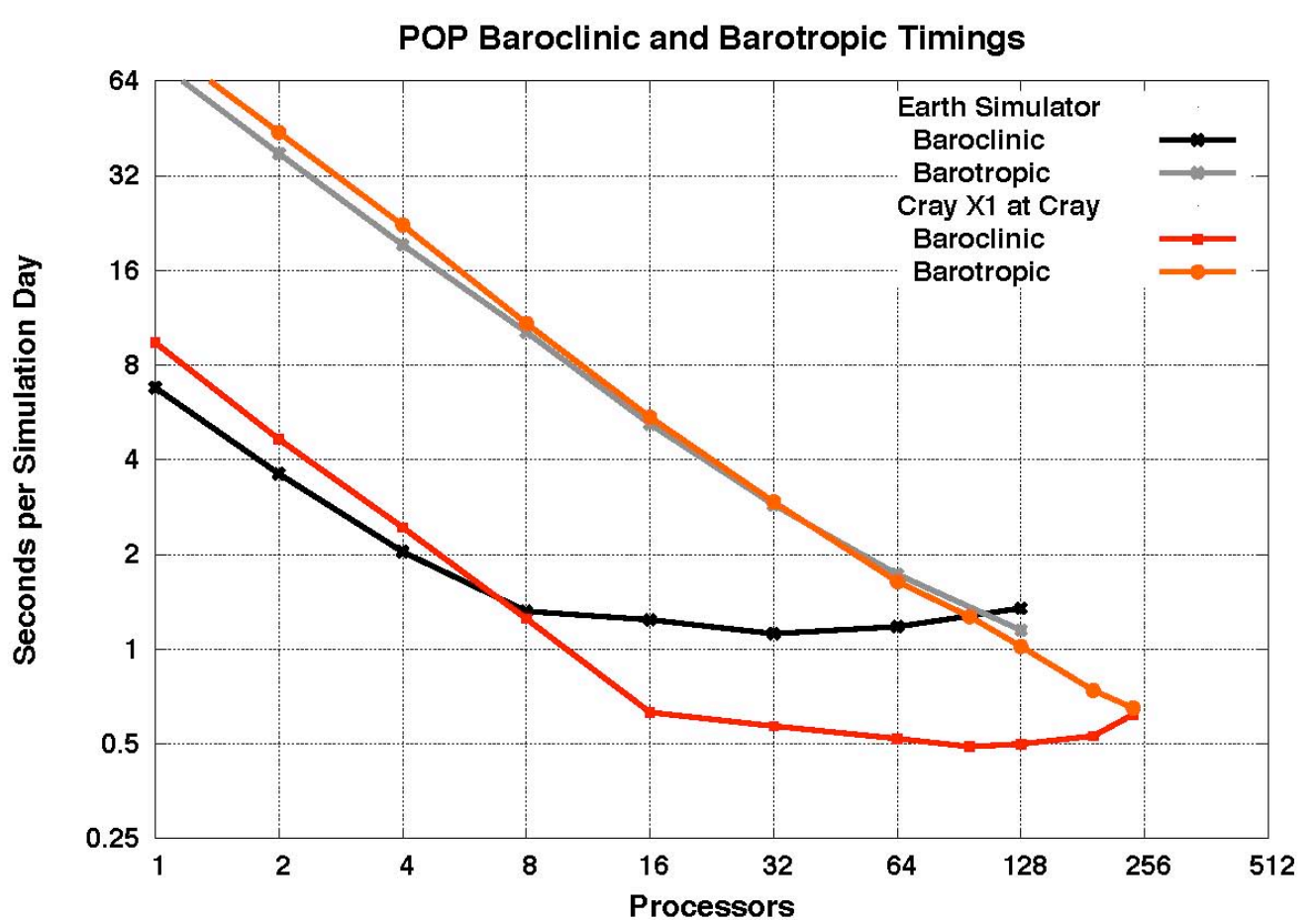
Comparing performance and scaling on the X1 over time. While the majority of recent improvements are due to performance improvements in the OS and other system software, many of these were motivated by the parallel algorithm analysis and optimizations.



POP Performance Diagnosis vs. ES40

Cray X1 at Cray
Communication-bound for more than 240 processors, with communication costs just starting to increase.

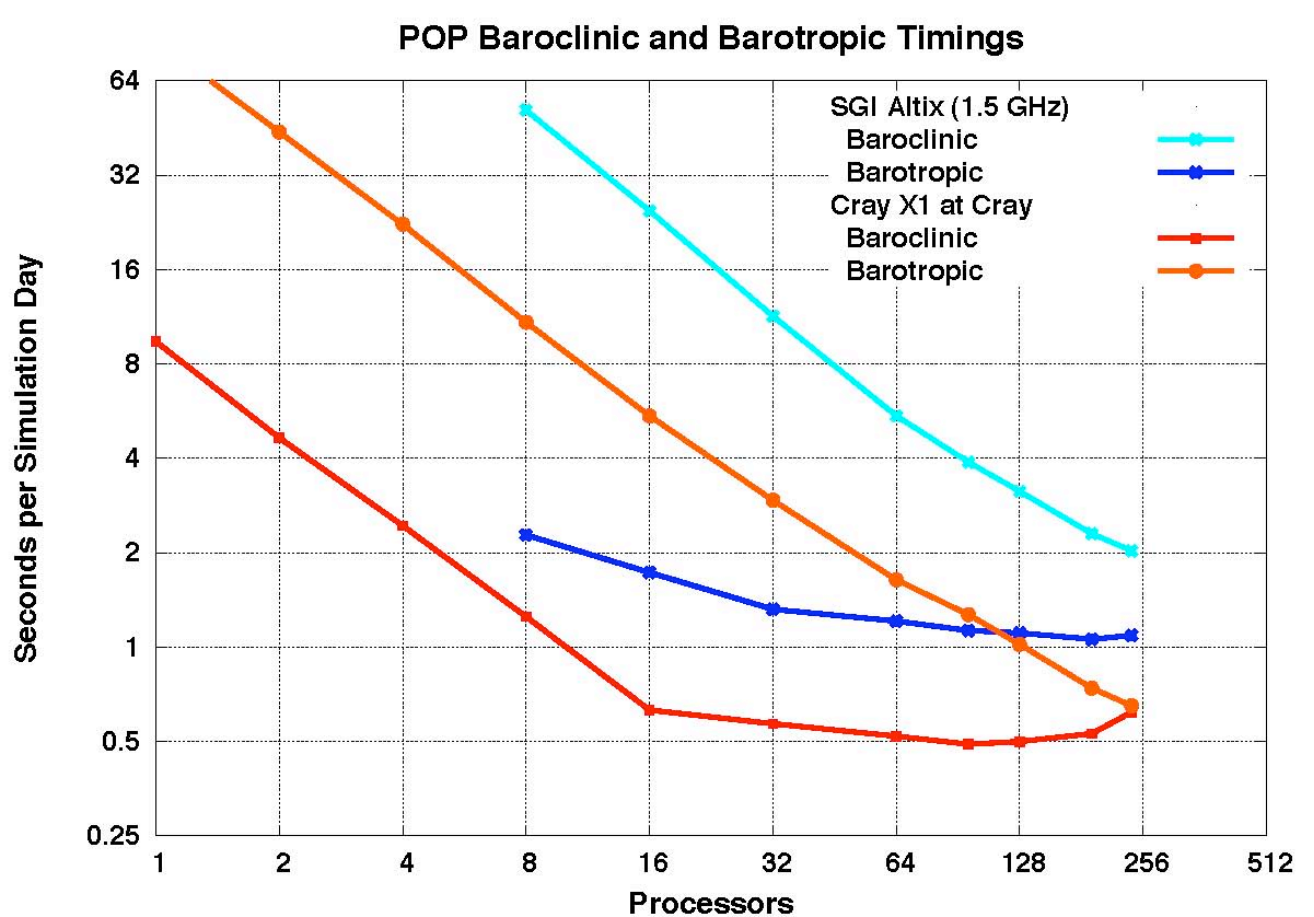
Earth Simulator
Communication-bound for 128 processors. Better vector performance for large granularity, but worse performance compared to X1 for small granularity (shorter vectors).



POP Performance Diagnosis vs. Altix

Cray X1 at Cray
- Communication-bound for more than 240 processors, with communication costs just starting to increase.

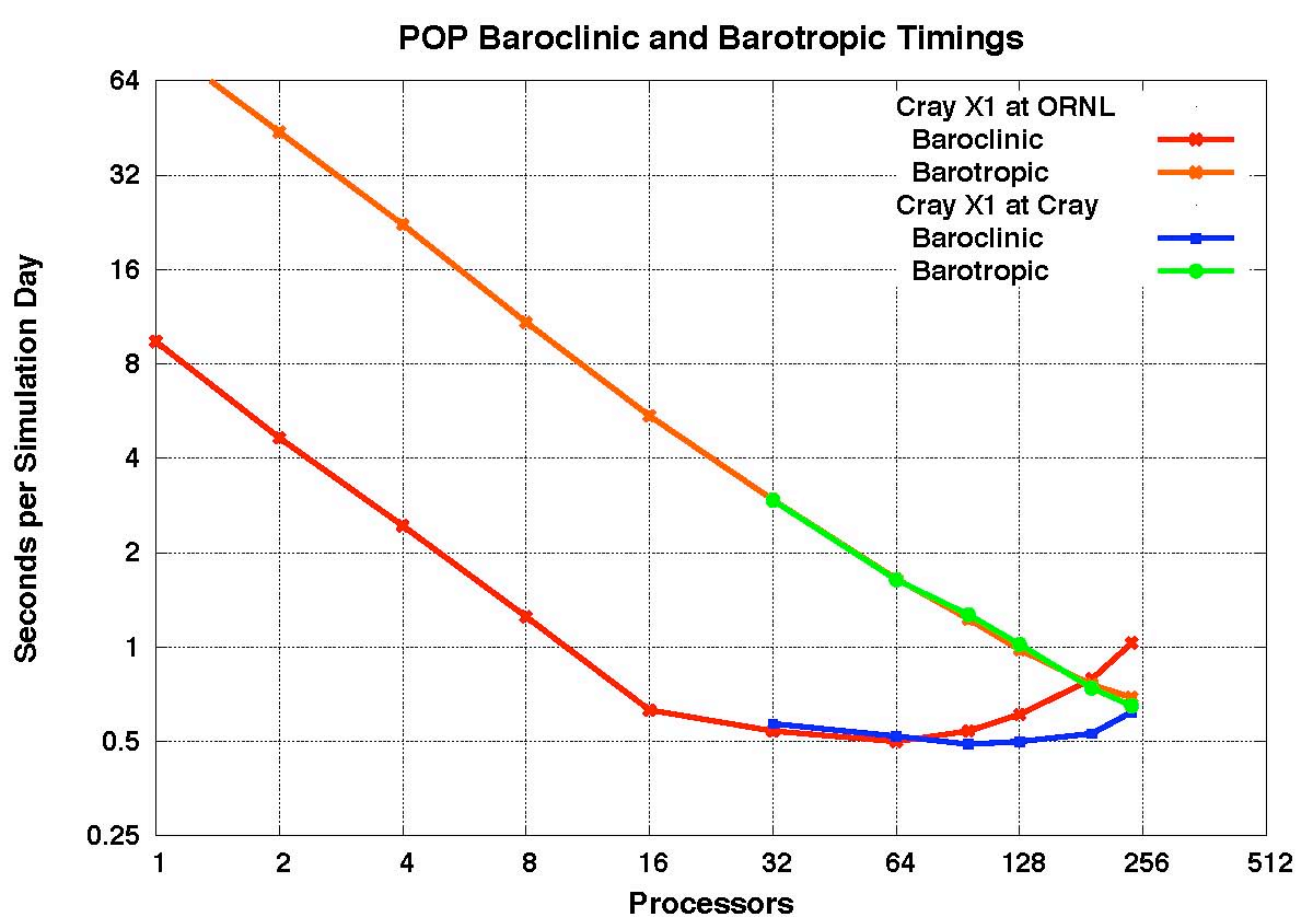
SGI Altix
- Not yet communication bound. Using MPI point-to-point and collectives for barotropic. Initial experiments with SHMEM do not show significant improvement.



POP Performance Diagnosis vs. X1 at Cray

Cray X1 at ORNL
- Communication-bound for more than 192 processors, with communication costs increasing.

Cray X1 at Cray
- Communication-bound for more than 240 processors. Communication costs increasing, but more slowly than on the system at ORNL



POP Summary and Futures

Using CRIEPI vectorization …

- X1 long vector performance is not as good as on the Earth Simulator (ES40), but X1 short vector performance is superior .
- X1 computation rate is 3-4 times better than that of the best nonvector system for the same processor count up to 128 processors.
- X1-specific vectorization of POP is currently being investigated.

Scalability of POP is determined by communication latency

- Co-Array Fortran performance is excellent. Algorithms currently used for global reduction are not scalable, and alternatives are being investigated.
- System limitations to scalability have been identified and are being addressed in future releases of the system software, as evidenced by results from the X1 at Cray.
- MPI short message and collective performance is mediocre. MPI collectives should perform as well as the Co-Array Fortran implementation, and we expect the performance need for Co-Array Fortran to diminish in the near future.

GYRO

- GYRO is an Eulerian gyrokinetic-Maxwell solver developed by R.E. Waltz and J. Candy at General Atomics. It is used in the DOE SciDAC Fusion Energy project studying plasma microturbulence.
- GYRO comes with ports to a number of different platforms. The port and optimization on the Cray X1 is primarily due to Mark Fahey of ORNL.
- Optimization on the X1 ongoing.

GYRO Experiment Particulars

One fixed size time dependent benchmark:

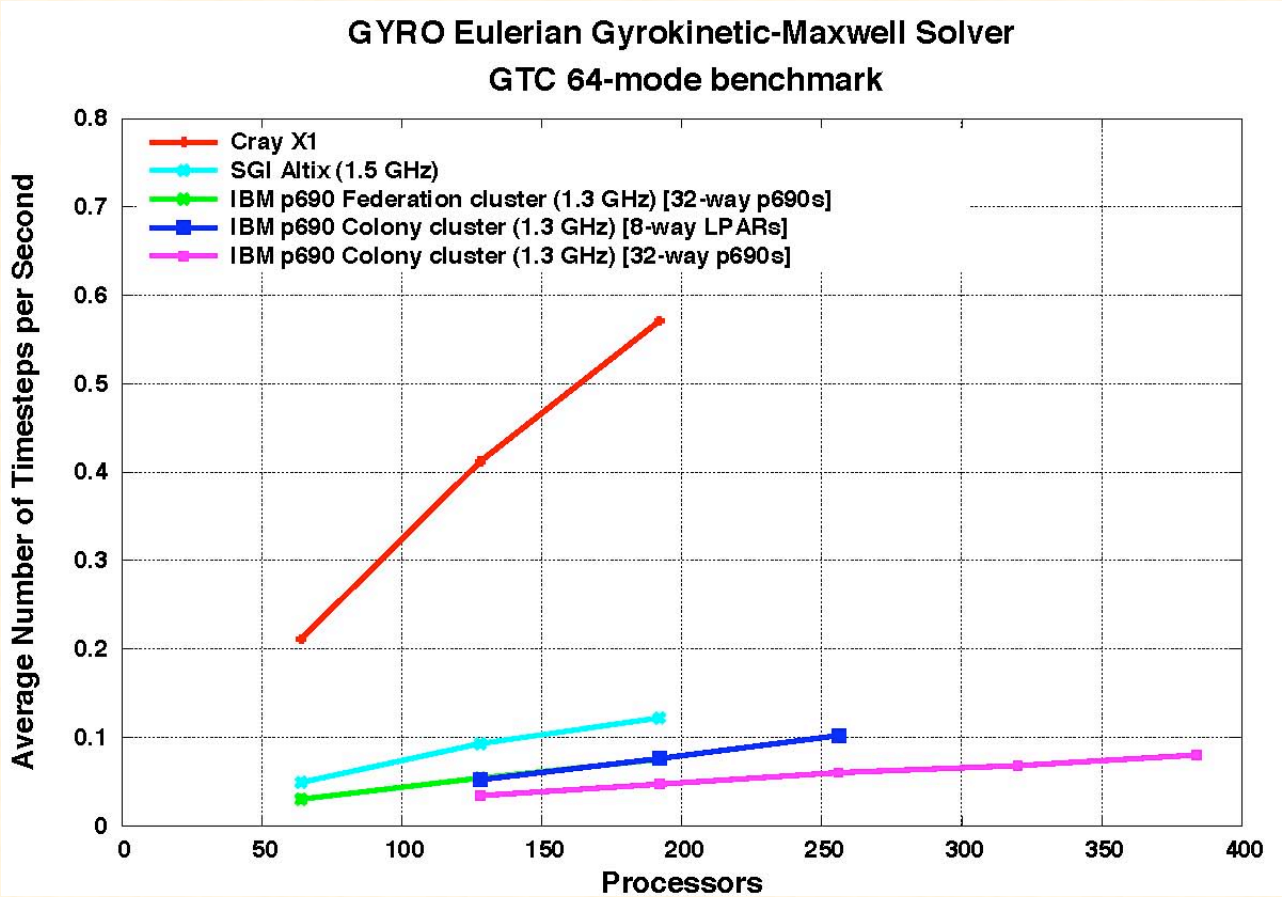
GTC.n64.500

- 64-mode adiabatic electron case. It is run on multiples of 64 processors. Duration is 3 simulation seconds, representing 100 timesteps.
- Primary communication cost in this benchmark is calls to MPI_ALLTOALL to transpose distributed matrices.

Current production runs use 32 modes, so benchmark is large.

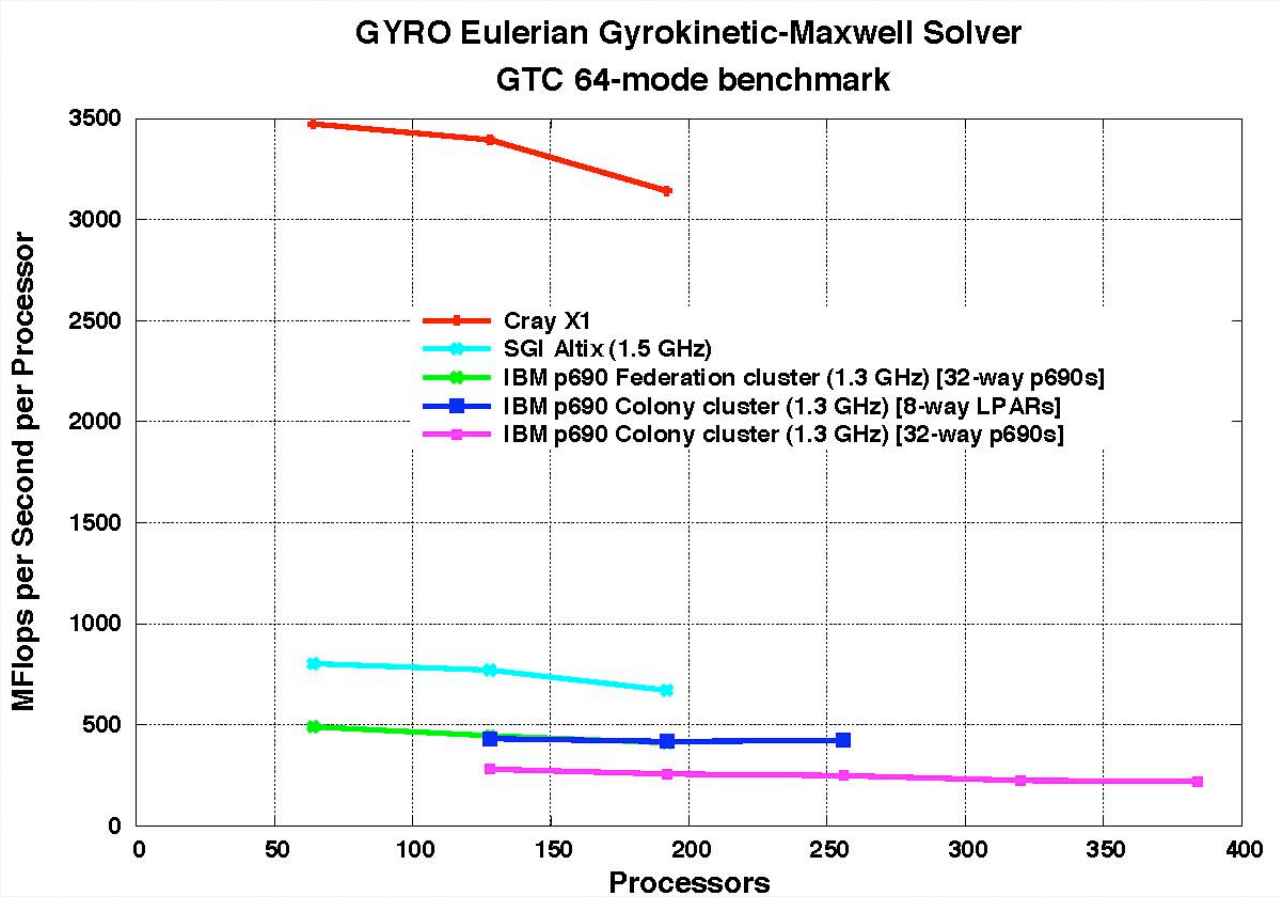
GYRO Simulation Rate

Comparing performance and scaling across platforms.
- X1 performance is significantly better than performance on other platforms, and the advantage grows with processor count.



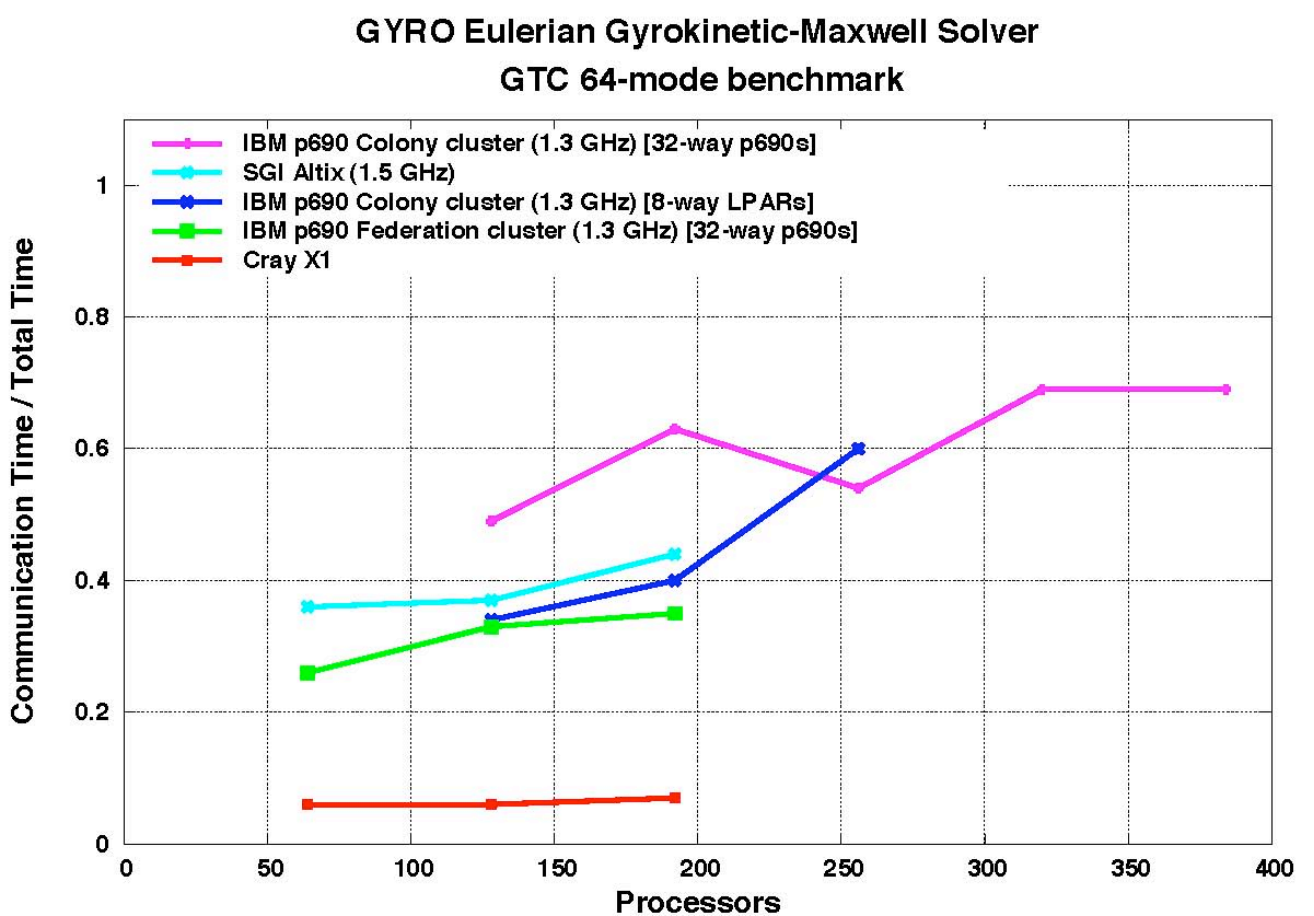
GYRO Computation Rate

All platforms show reasonable scaling, but IBM and SGI performance limited by communication overhead.



GYRO Communication Overhead

The X1 advantage in communication overhead is significant, and grows with processor count.



GYRO Summary

- Performance on nonvector systems is constrained by communication bandwidth.
  - **This is not true on the Cray X1.**
- Vectorization efforts are not complete. There are known (correctable) losses in vector performance. There is also a phase of computation that does not yet scale beyond 64-way parallelism. This has not limited performance on the nonvector systems, but it does limit performance on the X1.

Scalability Analysis Summary

X1 advantages

- Low latency
- High bandwidth
- Short vector performance

X1 issues

- OS support for scalability
- MPI latency
- MPI collectives
- Scalar performance

Questions ? Comments ?

For further information on these and other evaluation studies, visit

http://www.csm.ornl.gov/evaluation

or send e-mail to

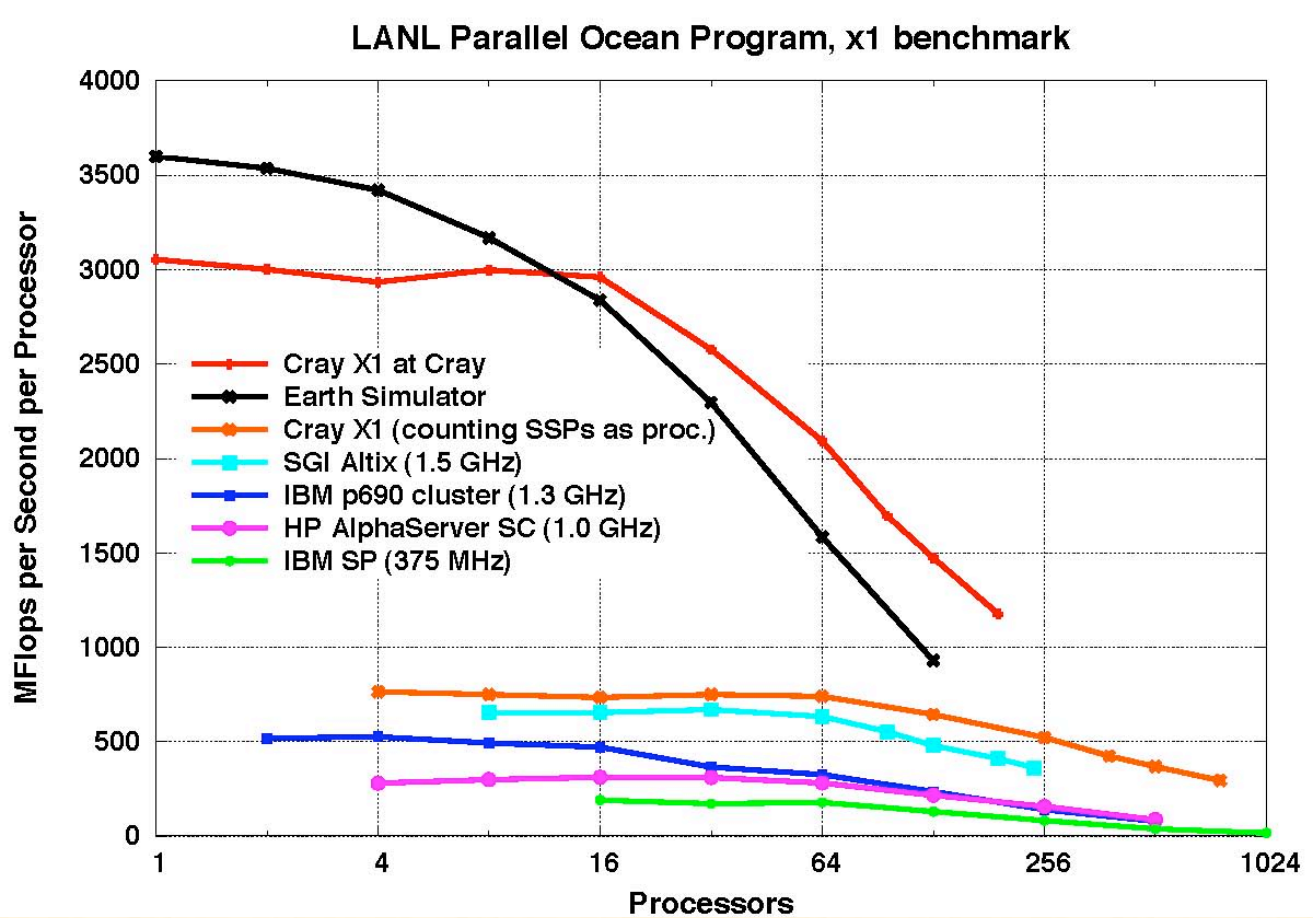
worleyph@ornl.gov

Additional Slides

- **Application Performance**
  - POP Benchmark
    - computation rate
    - performance analysis wrt X1 at ORNL
  - GYRO Benchmarks
    - small benchmark
    - large benchmark results replotted by SSP count
- **"What is a Processor"**

POP Computational Rate

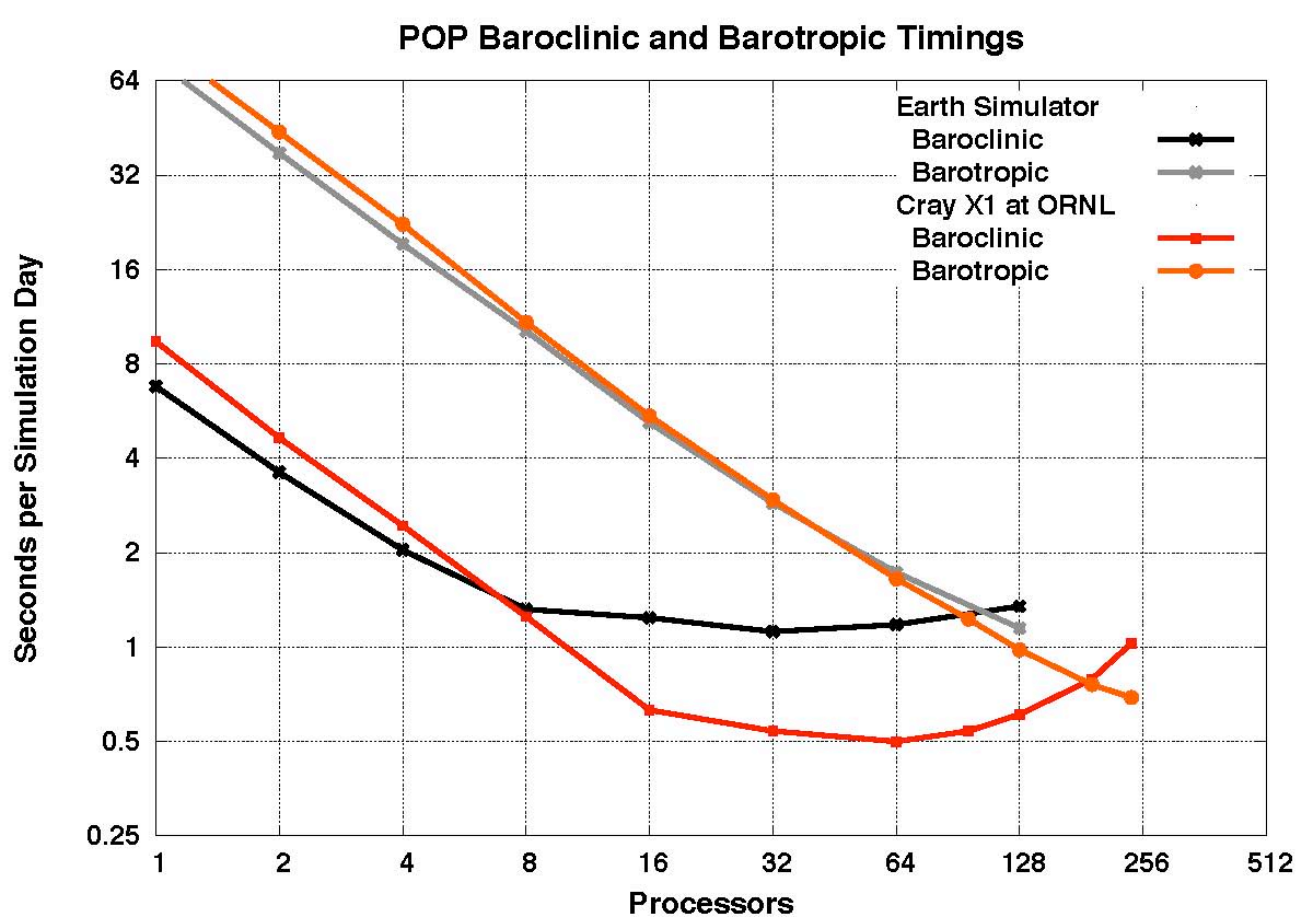
- X1 is not as fast as the Earth Simulator for small processor counts. X1 maintains performance better as granularity (and vector length) decrease.
- POP views the MSP as the processor. However, replotting data with SSPs as the processor still shows a significant performance advantage compared to the nonvector processors.



POP Performance Diagnosis vs. ES40

Cray X1 at ORNL
- Communication-bound for more than 192 processors, with communication costs increasing.

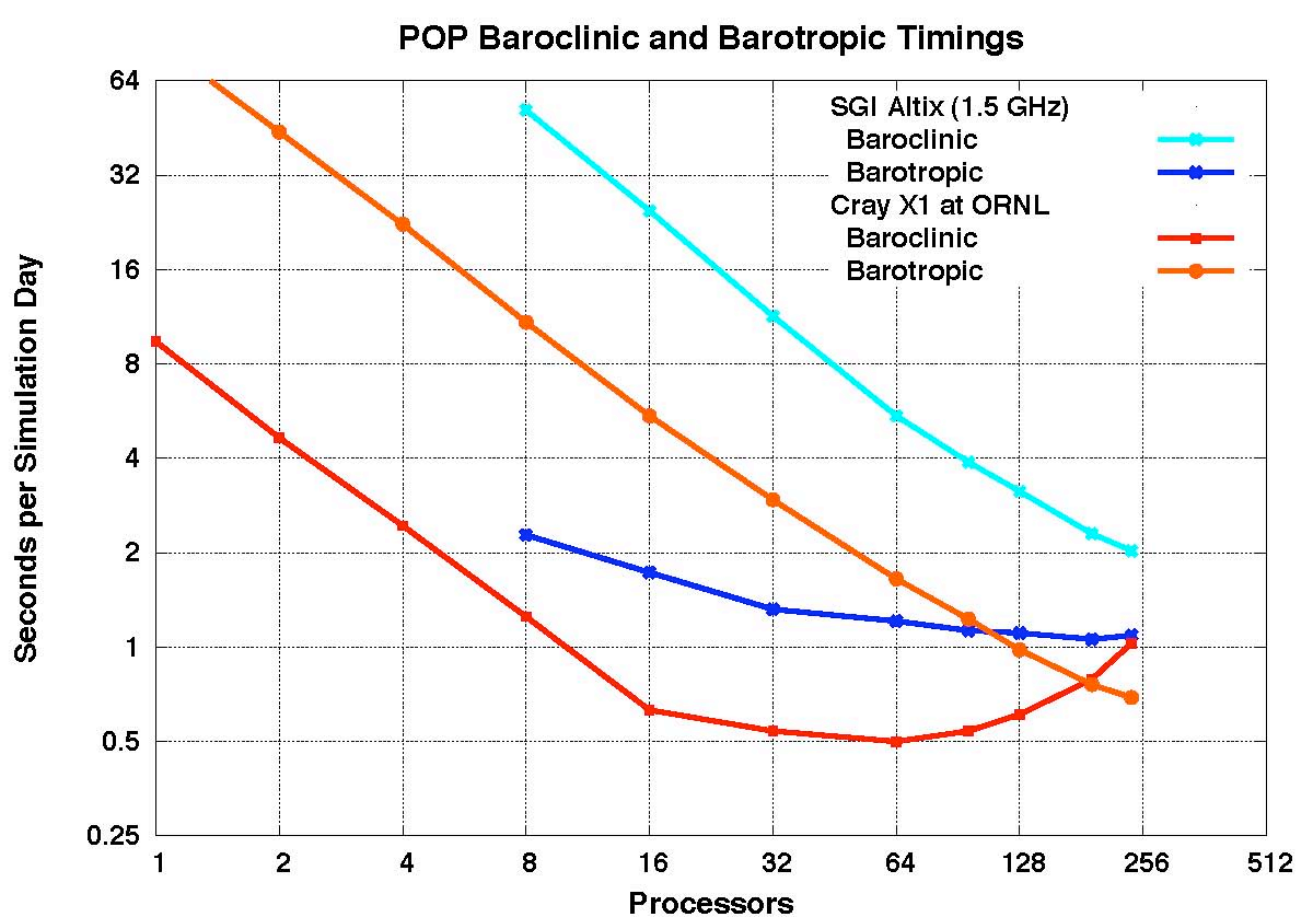
Earth Simulator
- Communication-bound for 128 processors. Better vector performance for large granularity, but worse performance compared to X1 for small granularity (shorter vectors).



POP Performance Diagnosis vs. Altix

Cray X1 at ORNL
- Communication-bound for more than 192 processors, with communication costs increasing.

SGI Altix
- Not yet communication bound. Using MPI point-to-point and collectives for barotropic. Initial experiments with SHMEM do not show significant improvement.



GYRO Experiment Particulars

Two benchmark problems, both time dependent:

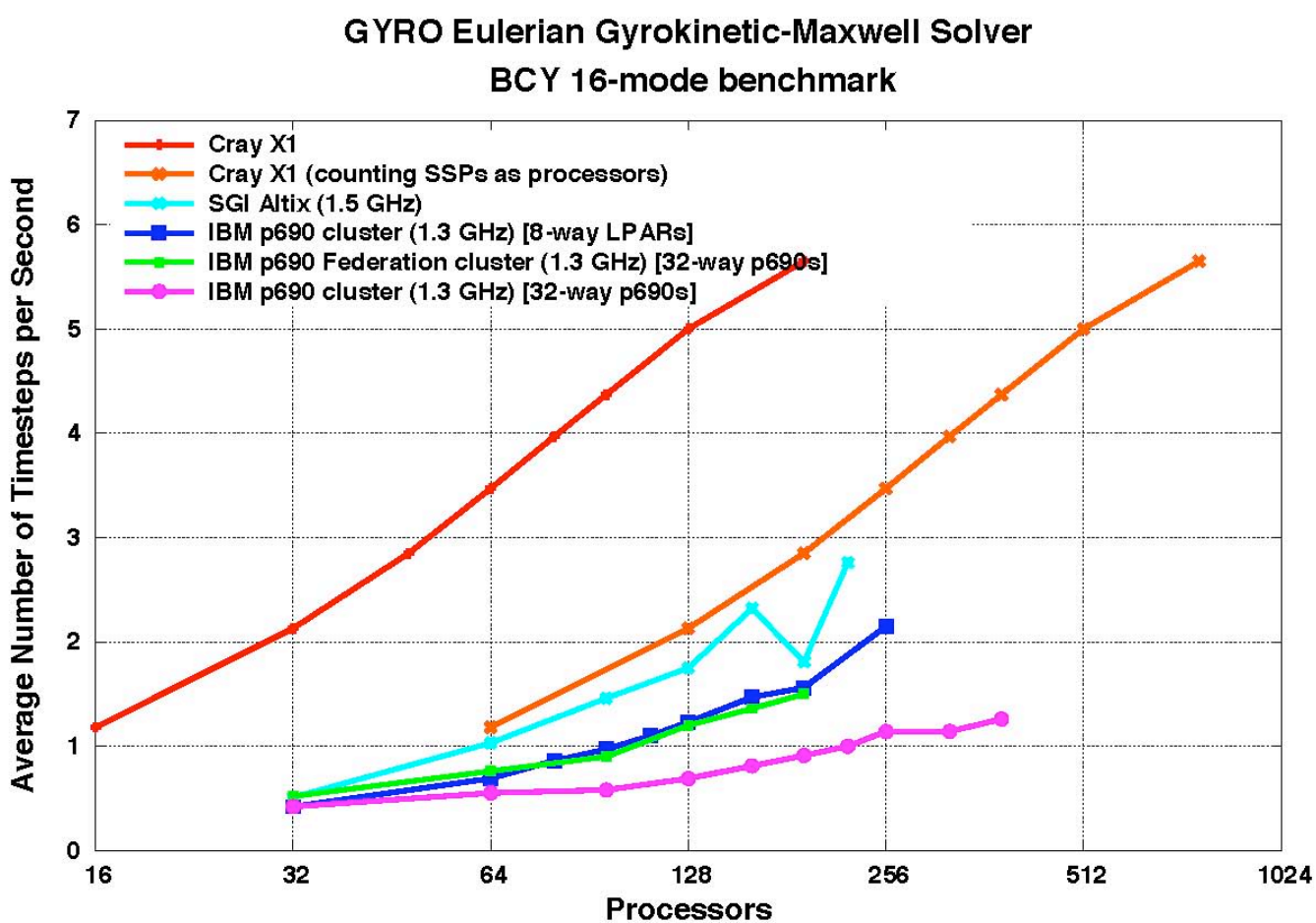
1. BCY.n16.b.25
   - 16-mode electromagnetic case. It is run on multiples of 16 processors. Duration is 8 simulation seconds, representing 1000 timesteps.

2. GTC.n64.500
   - 64-mode adiabatic electron case. It is run on multiples of 64 processors. Duration is 3 simulation seconds, representing 100 timesteps.

Current production runs use 32 modes, so benchmark #1 is somewhat small, while benchmark #2 is very large. (J. Candy is in the process of reformulating these benchmarks to also cover production-size problems.)

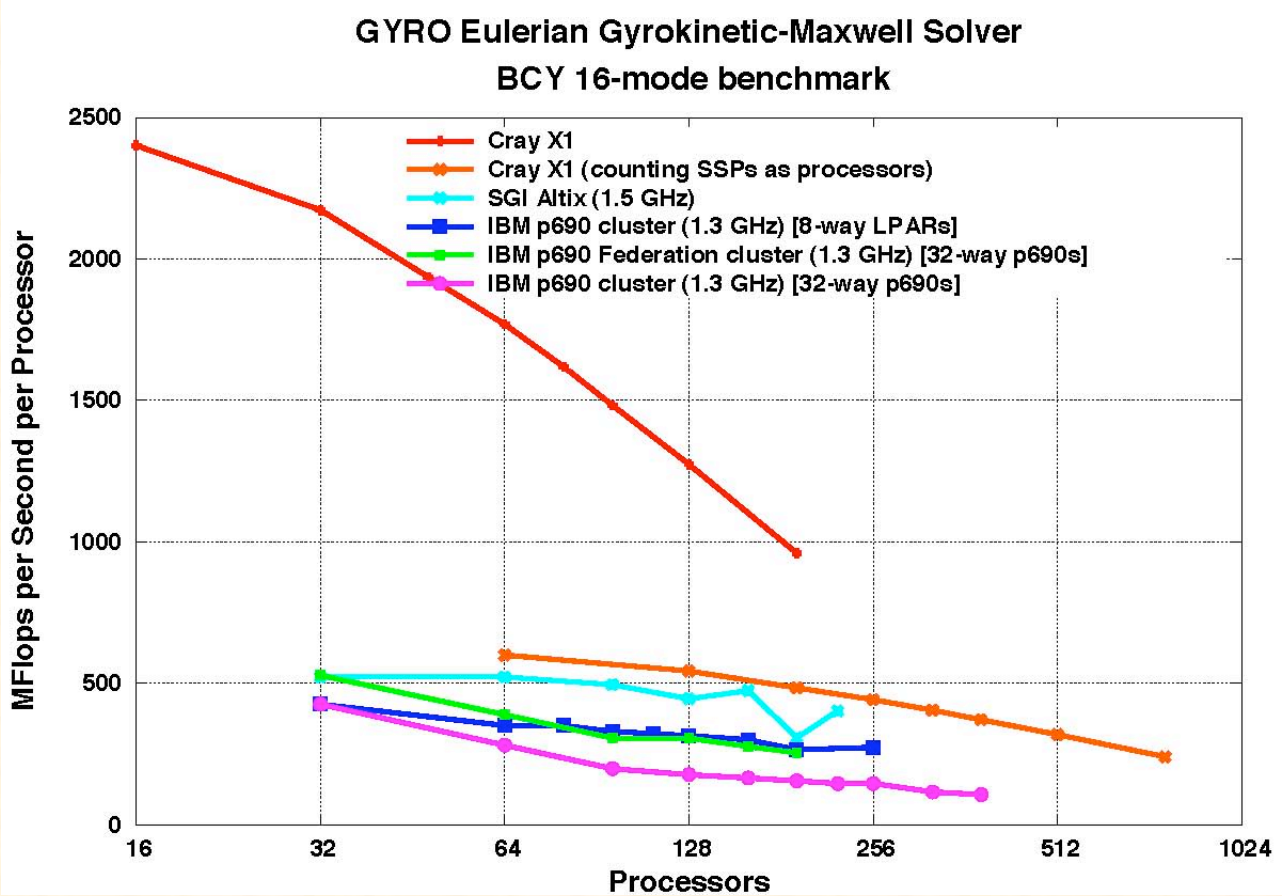
GYRO Simulation Rate

Comparing performance and scaling across platforms.
- X1 performance is significantly better than that on other platforms even for this modest size problem, and advantage grows with processor count. Even replotting data with SSPs as processors indicates that the X1 is the faster platform.



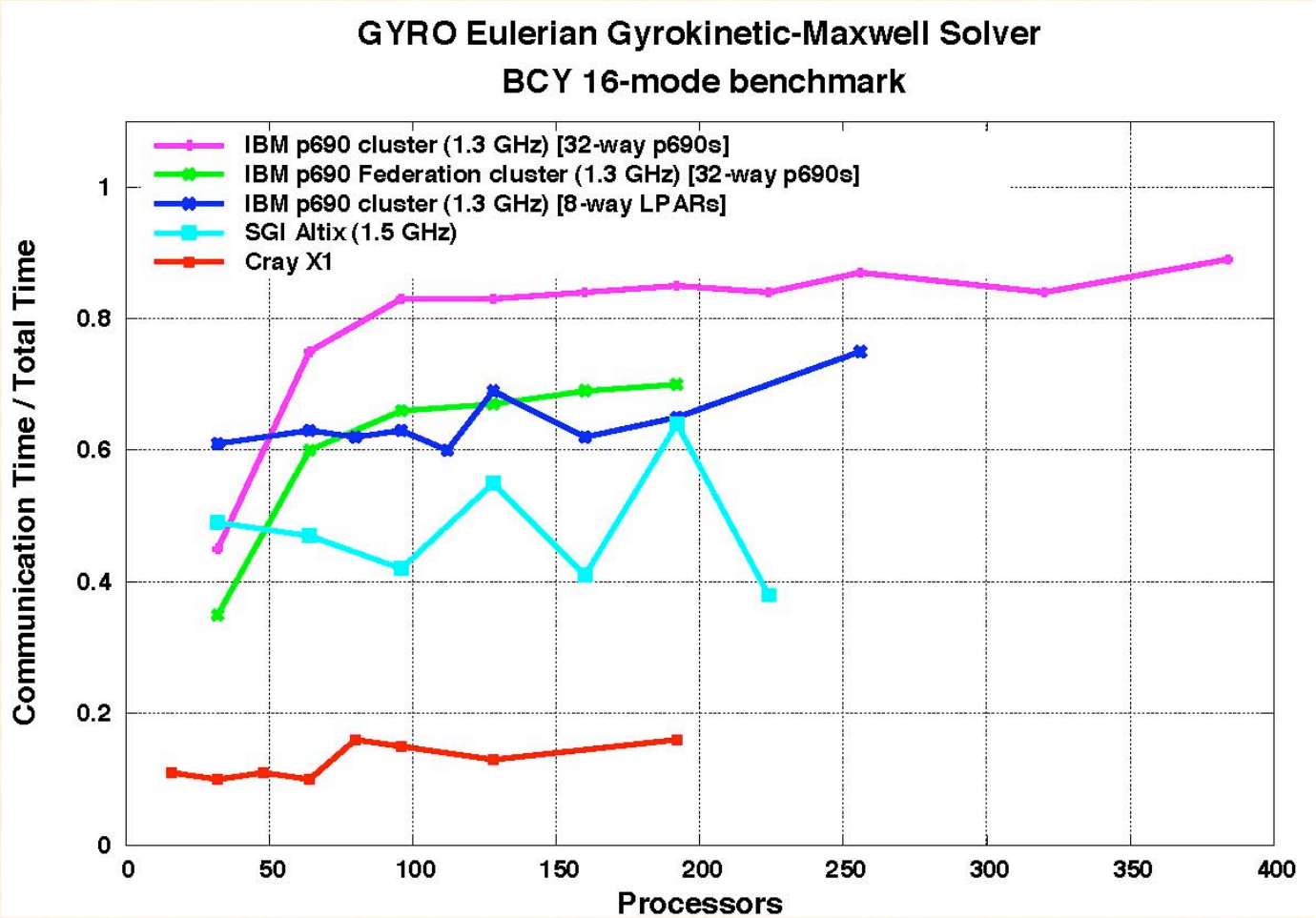
GYRO Computational Rate

- IBM performance is limited by communication overhead.
- X1 performance is limited by nonscaling part of parallel algorithm. At 192 processors, 25% of the time is spent in a phase that does not scale beyond 16-way parallelism.



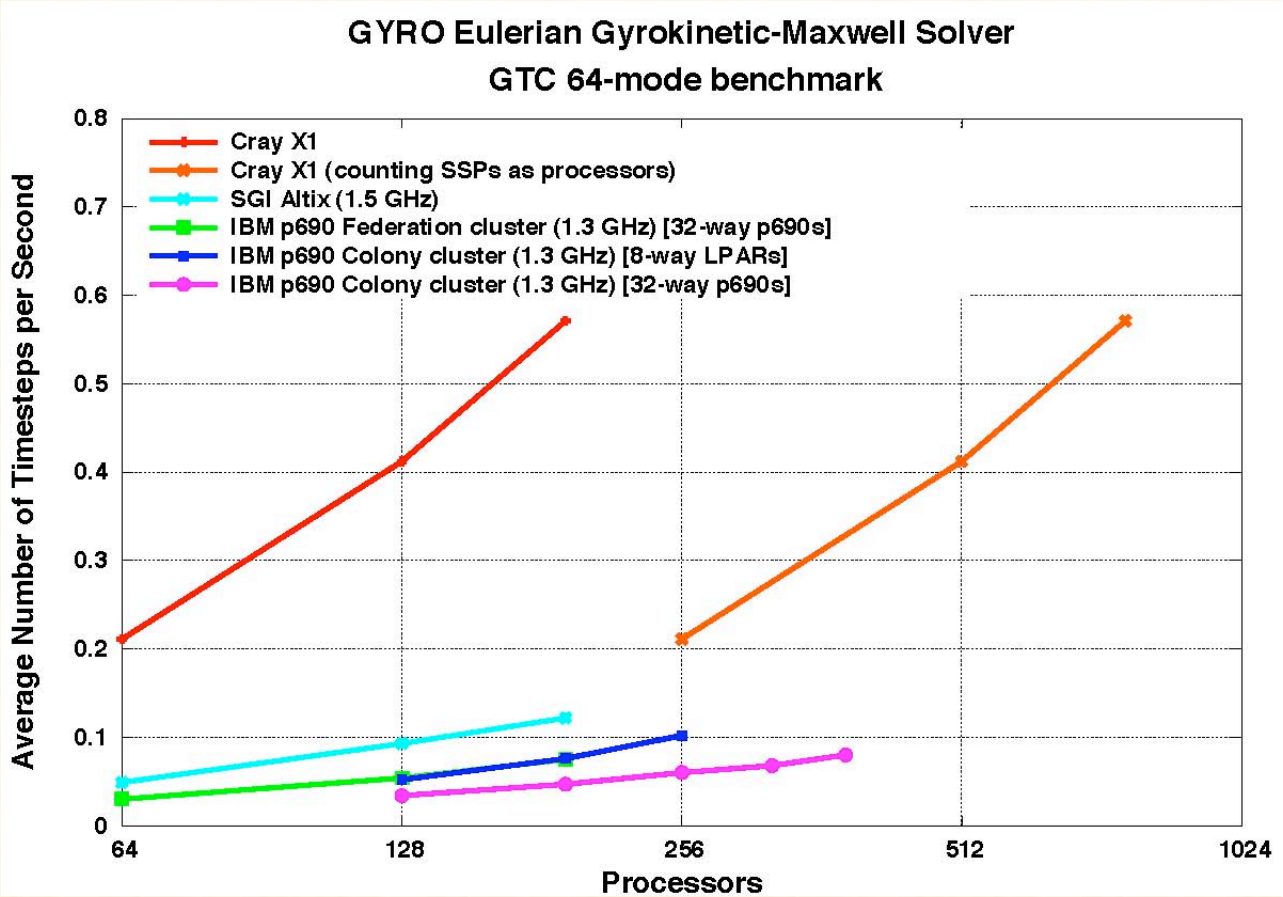
GYRO Communication Overhead

The X1 advantage in communication overhead is significant, and grows with processor count.



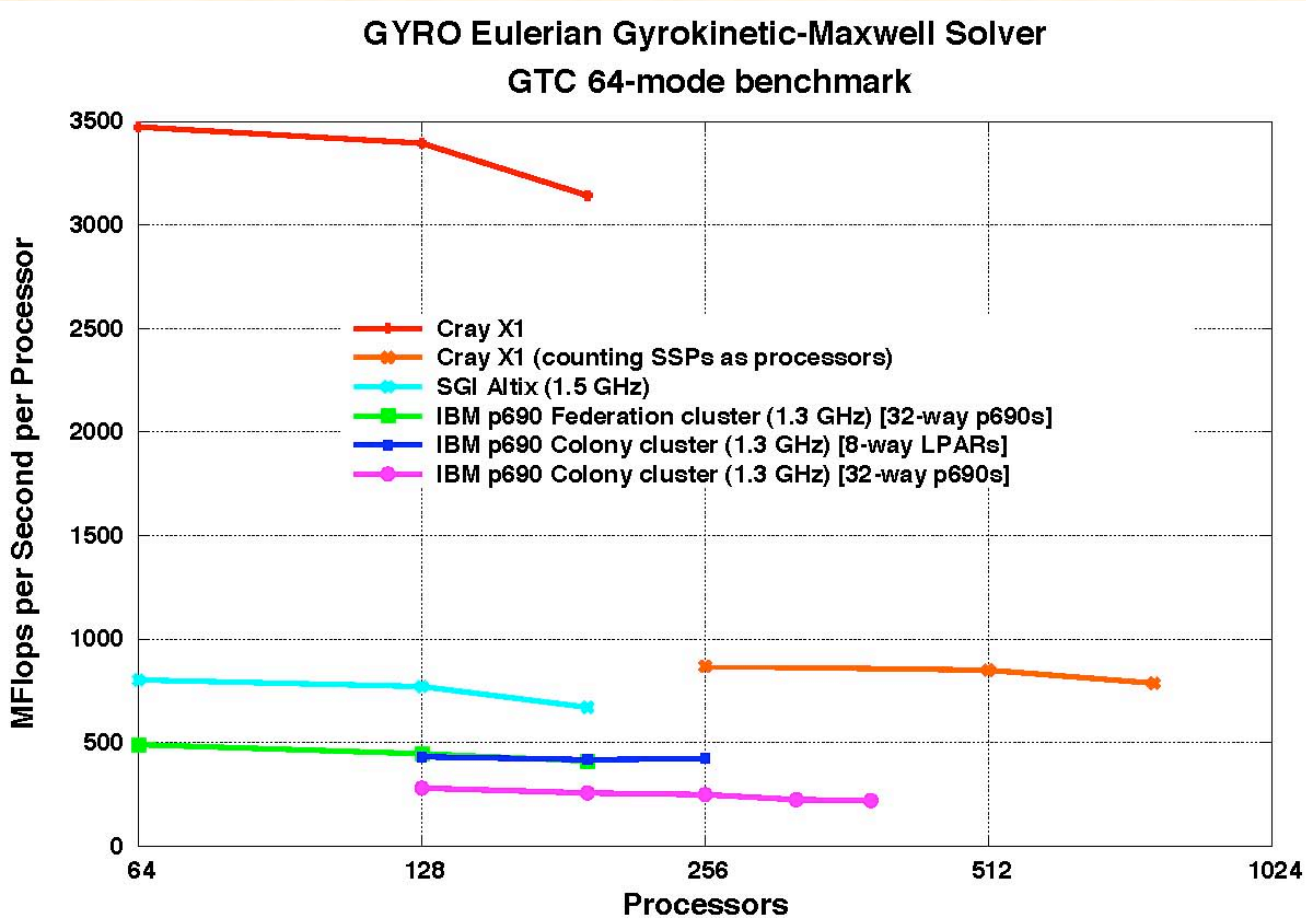
GYRO Simulation Rate

Comparing performance and scaling across platforms.
- X1 performance is significantly better than performance on other platforms, and the advantage grows with processor count.



GYRO Computation Rate

All platforms show reasonable scaling, but IBM and SGI performance limited by communication overhead.



Aside: What is a processor on the X1?

It depends on why you are asking the question …

- As a user, I want fewer, more powerful, processors, in order to minimize parallel overheads. So I *want* an MSP to be "the processor".
- As a user, a processor is what my code views as a processor. If it is written explicitly to exploit SSPs, than the SSP is the processor. If the compiler assigns work to SSPs without any intervention other than possibly simple compiler directives, then an MSP is the processor.
- The Cray X1 compiler can often partition inner loops and assign work to functional units in different SSPs (within the same MSP) and achieve good performance. This level of integration is typically seen only within a processor, so it is reasonable to call an MSP a processor in these cases.