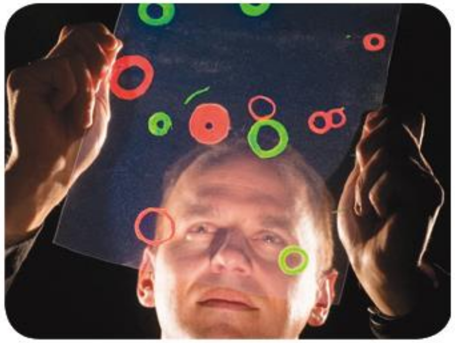
Exceptional service in the national interest

Sandia National Laboratories

Experiences with Sandia National Laboratories HPC applications and MPI performance



Mahesh Rajan, Doug Doerfler, Richard Barrett, Joel Stevenson, Anthony Agelastos, Ryan Shaw and Hal Meyer

MVAPICH User Group Meeting, Aug 25-27, 2014, Columbus OH

U.S. DEPARTMENT OF ENERGY

NNSA National Nuclear Security Administration

Sandia National Laboratories is a multi-program 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. SAND2011-0439P

Sandia's History

exceptional service in the national interest.

THE WHITE HOUSE
WASHINGTON

May 13, 1949

Dear Mr. Wilson:

I am informed that the Atomic Energy Commission intends to ask that the Bell Telephone Laboratories accept under contract ... at Albuquerque, New Mexico.

... atomic ... na- tional defense, and should ... technical direc- tion.

I hope that after you have heard more in detail from the Atomic Energy Commission, your organization will find it possible to undertake this task. In my opinion you have here an opportunity to render an **exceptional service in the national interest.**

I am writing a similar note direct to Dr. O. E. Buckley.

Very sincerely yours,

Harry Truman

Mr. Leroy A. Wilson,
President,
American Telephone and Telegraph Company,
195 Broadway,
New York 7, N. Y.

Sandia's Sites

Albuquerque, New Mexico

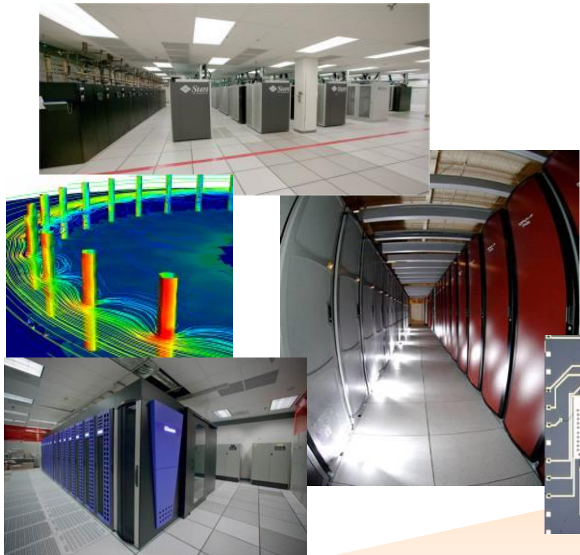
Livermore, California

Tonopah, Nevada

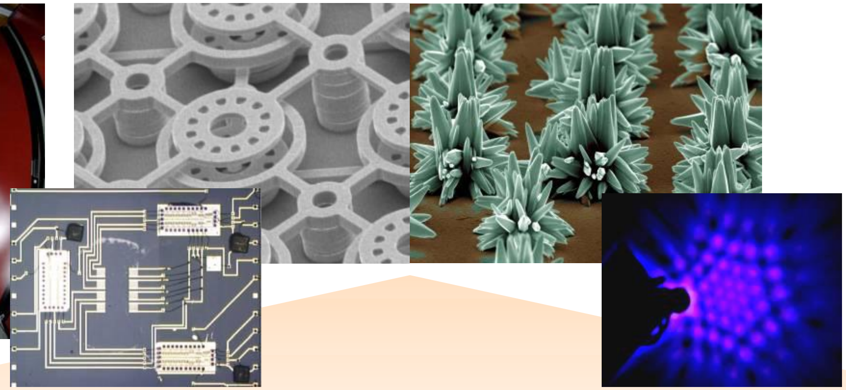
Waste Isolation Pilot Plant, Carlsbad, New Mexico

Pantex, Texas

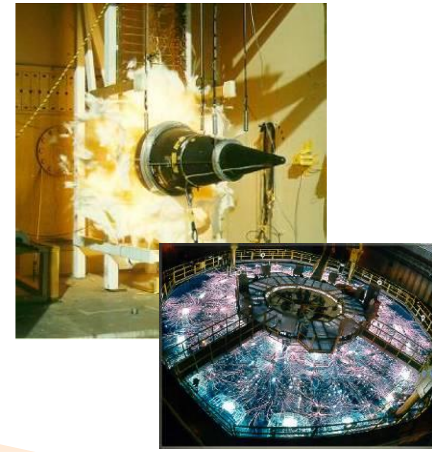
Research Disciplines Drive Capabilities



High Performance Computing

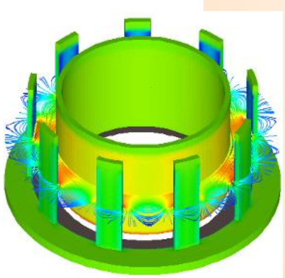


Nanotechnologies & Microsystems

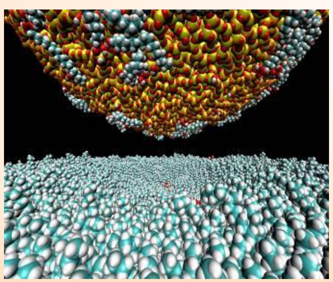


Extreme Environments

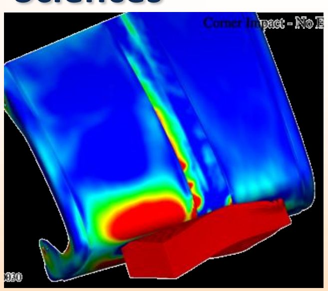
Computer Science



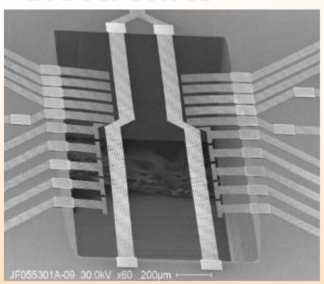
Materials



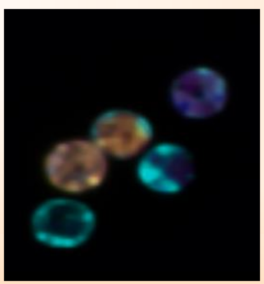
Engineering Sciences



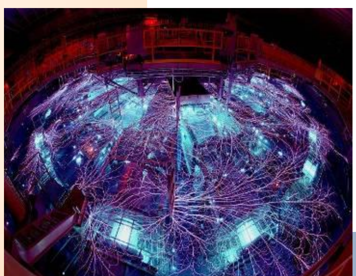
Micro Electronics



Bioscience



Pulsed Power



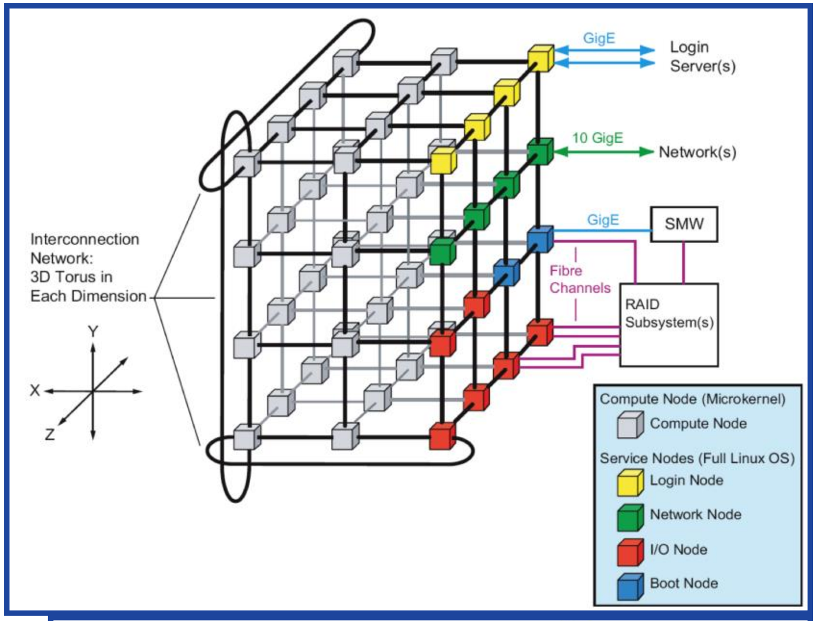
Research Disciplines

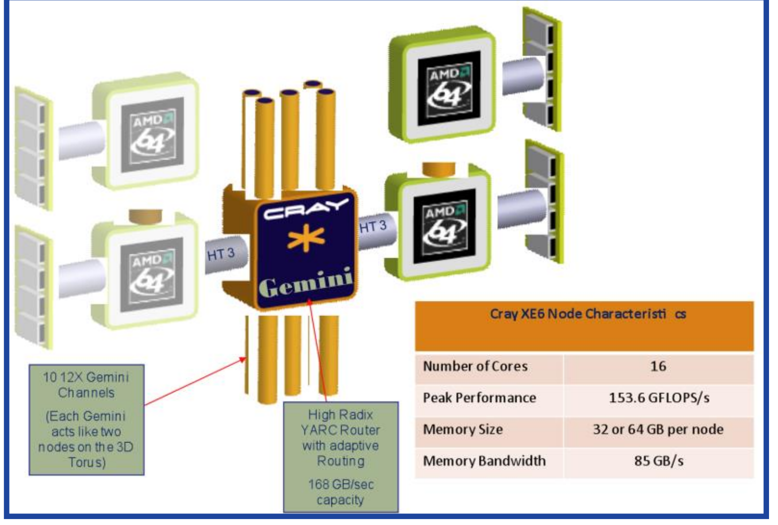
Capacity Computing Resources (6-2014)

System	Vendor	Nodes	Processor cores total	Processor:Sockets:Cores/Socket & Interconnect	Memory/ Node	Memory/ Core	TFLOPS	GA Date
Red Sky	**Sun**	**2,823**	**22,584**	**2.93 GHz Intel Nehalem:2S:4C; Mellanox ConnectX IB QDR**	**12**	**1.5**	**264**	**Apr-10**
Glory	Appro	272	4,352	2.2 GHz AMD:4S:4C	32	2	38	Jan-09
Chama	**Appro**	**1,232**	**19,712**	**2.6 GHz Intel Sandy Bridge:2S:8C; Qlogic 4X IB QDR**	**32**	**2**	**392**	**Sep-12**
Uno	Dell	251	3,344	2.7 GHz Intel Sandy Bridge:2S:8C/4S:8C	64/128	4/8	71	TBD
Black Total:		**4,578**	**49,992**				**765**	
Red Sky	Sun	519	4,152	2.93 GHz Intel Nehalem:2S:4C	12	1.5	48	Nov-10
Jemez	**HP**	**288**	**4608**	**2.6 GHz Intel Sandy Bridge:2S:8C; Mellanox ConnectX-3 IB FDR**	**32**	**2**	**95**	**Jun-14**
Unity	Appro	272	4,352	2.2 GHz AMD:4S:4C	32	2	38	Mar-09
Whitney	Appro	272	4,352	2.2 GHz AMD:4S:4C	32	2	38	Mar-09
Pecos	Appro	1,232	19,712	2.6 GHz Intel Sandy Bridge:2S:8C	32	2	392	Sep-12
Red Total:		**2,583**	**37,176**				**611**	
Red Mesa	Sun	1,920	15,360	2.93 GHz Intel Nehalem:2S:4C	12	1.5	180	May-10
Green Total:		**1,920**	**15,360**				**180**	
Gila	**HP**	**48**	**1,152**	**2.3 GHz AMD 6176:2S:12C**	**24**	**2**	**11**	**Apr -13**
Dark Bridge	**Appro**	**924**	**14,784**	**2.6 GHz Intel Sandy Bridge:2S:8C**	**64**	**4**	**294**	**Nov-12**
Dark Sand	**Appro**	**924**	**14,784**	**2.6 GHz Intel Sandy Bridge:2S:8C**	**64**	**4**	**294**	**Dec-13**
Orange Total:		**1,896**	**30,720**				**599**	
TOTAL:		**10,977**	**133,248**				**2,155**	

Sandia/LANL ACES Capability System: Cray XE6

- **Topology**
  - **Gemini High-Speed Interconnect**
  - **Phase 2: 24x16(8)x24 3D Torus**
- **Node: dual-socket, AMD Magny-Cours**
  - **16 total cores (8 per socket)**
  - **2.4 GHz core clock rate**
  - **8 channels (4 per socket) 1333 MHz DDR3 memory**
  - **4 FLOPS per clock per core**
  - **32 GB total memory**
  - **153.6 GF peak FP**
  - **85.3 GB/s peak memory BW**





Total # of racks	**96**
Total # of compute blades	**2,235**
# of compute nodes	**8,940**
# of cores	**143,040**

Sandia's Advanced Architecture Systems
Focus: Portability of SNL applications, Performance, Programming Models

Hostname /Integrator	Num Nodes	Compute Node Processor	Memory/Node	Accelerator	Interconnect
Volta / Cray XC30m	56	Intel Xeon E5-2695 V2 dual socket (Ivy Bridge) with 24 cores total 2.4 GHz.	64 GB DDR3-1866MHz	None	Aires
Teller /Penguin	104	AMD A10-5800K (Piledriver) 3.8GHz Quad-core	100 nodes have 16GB DDR3-1600MHz	400-core Radeon HD 6550D @ 600MHz	Qlogic QSFP QDR Infiniband
Shannon /Appro	32	two 8-core Sandy Bridge Xeon E5-2670 @ 2.6GHz	128GB DDR3	24 2x NVIDIA K20x 3 4x NVIDIA K20x 4 4x NVIDIA K40m	Mellanox QDR Infiniband
Compton / appro	42	two 8-core Sandy Bridge Xeon E5-2670 @ 2.6GHz (HT activated)	24GB (3*8Gb)	42 nodes with Pre-production KNC 2 per node	Mellanox Infiniscale IV QDR Infiniband
Curie /Cray XK7	52	16-core 2.1GHz 64bit AMD Opteron 6200 CPU's (Interlagos)	32 GB - 4 channels of DDR3 memory per compute node	NVIDIA K20x	Gemini 1.2

Sandia's Advanced Architecture Systems and Mantevo mini-apps

- Project Aim: Study Future Architectures, Power and Programming Models
  - Study low power cores based nodes
  - AMD APU, Intel Xeon Phi, NVIDIA GPU, …
  - Lighter-weight hosts, e.g. ARM?; hostless?
  - Multi-level memory hierarchies
  - Interconnects
  - Programming models:
  - Memory model abstractions, e.g. Kokkos
  - Task parallel, over-decomposition of smaller granularity data parallel computation.

Miniapps : Tools enabling exploration

Focus	Proxy for a key app performance issue
Intent	Tool for codesign: output is information
Scope of change	Any and all
Size	A few thousand lines of code
Availability	Open source (LGPL)
Developer/owner	Application team
Life span	Until its no longer useful

Related:

Benchmark	Output: metric to be ranked.
Compact app	Application relevant answer.
Skeleton app	Inter-process comm, application “fake”
Proxy app	Über notion

Mantevo 2.0+ status; http://mantevo.org

miniApp or miniDriver	
Cleverleaf 1.0	Eulerian on structured grid with AMR
CloverLeaf 1.0.1	Compressible Euler eqns, explicit 2nd order accurate
CoMD 1.1	Molecular dynamics (SPaSM)
EpetraBenchmarkTest 1.0	Exercises Epetra sparse and dense kernels.
HPCCG 1.0	Unstructured implicit finite element
miniFE 2.0	Implicit finite element solver
miniGhost 1.0.1	FDM/FVM explicit (halo exchange focus)
miniMD 1.2	Molecular dynamics (Lennard-Jones)
miniXyce 1.0	SPICE-style circuit simulator
miniAMR	Adaptive mesh refinement of an Eulerian mesh
miniSMAC	FD 2D incompressible N/S on a structured grid.
PathFinder	Signature search
miniAero	3D unstructured FV R-K 4th order time marching, inviscid Roe Flux
miniContact	Solid mechanics
miniExDyn-FE	Explicit Dynamics (Kokkos-based)
miniITC-FE	Implicit Thermal Conduction (Kokkos-based)
phdMesh	Explicit FEM: contact detection

2.0 release

+

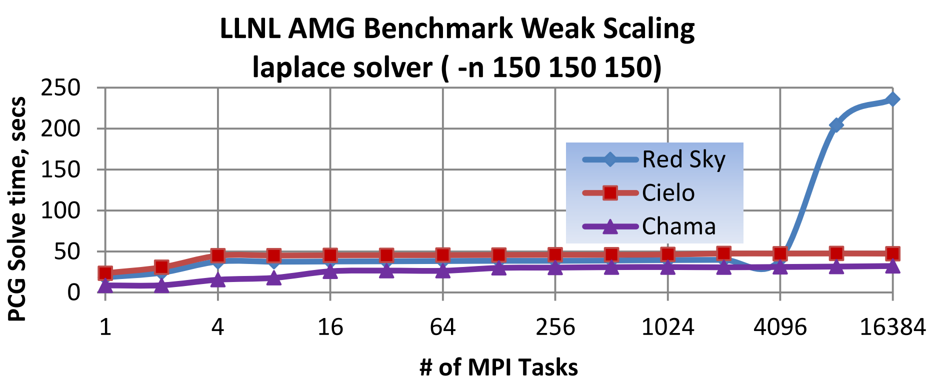
Coming soon:
- miniSAT
- miniTri

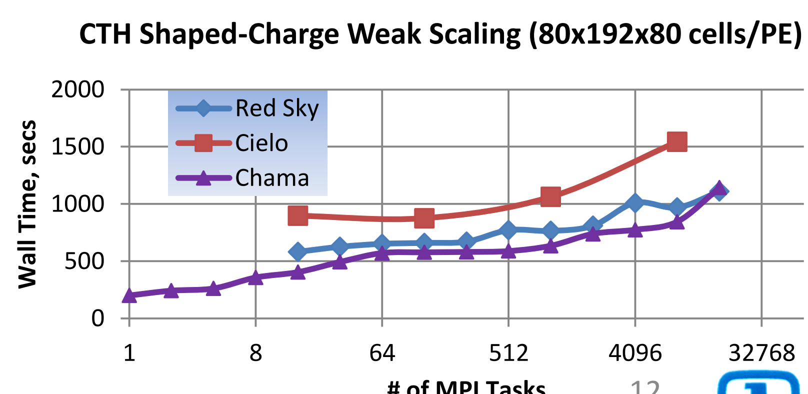
MPI Challenges with SNL HPC Systems and applications

- Limited staff supporting a variety of systems with different application requirements
  - Example: Chama, 1232 Node Capacity Cluster (software managed through modules)
    - MPI: OpenMPI(1.4,1.5,1.6), MVAPICH(1.2,1.7)
    - Compilers: GNU, Intel, PGI
    - Tools: mpiP, Vtune, HPCToolkit, OpenSpeedShop, TAU, MAP
- Makes the job of optimal resource utilization a challenge with so many possible combinations of compilers, libraries, tools and applications running on our many systems

Use of MVAPICH at SNL on Chama

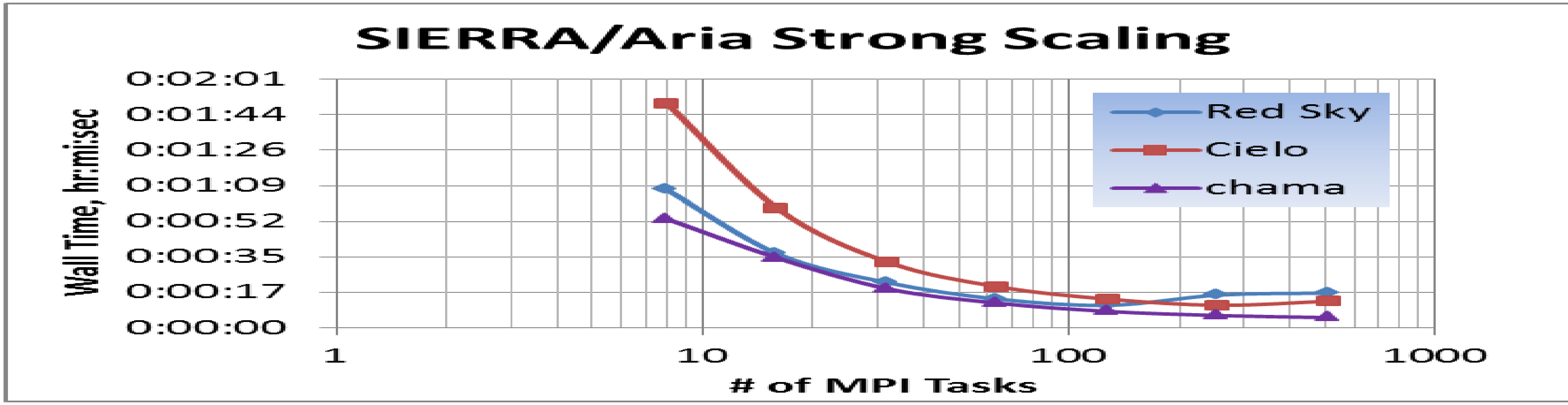
- Used MVAPICH psm/1.2 when we acquired and installed the TLCC2 systems in 2012/13
    - For all initial evaluation of the new system
    - For Top500 Linpack measurements: Feb 24; 332 TFLOPS; 81.11 % Efficiency on 1230 Nodes; Chama #31 on the top500 (Feb 2012)
    - For early investigations of throughput improvement with doubling the memory per core
- FOR SNL's Neutron Generator code; Aleph
    - Memory issues when tried with OpenMPI
    - Also later discovered an internal bug with an OpenMPI sub communicator
- **Early scaling studies on Chama with MVAPICH using *Cielo acceptance applications* showed excellent scaling. First time such good scaling seen with commodity clusters. Below scaling plots for a few Cielo acceptance apps (See SC2013 PMBS paper by Rajan, et.al.)**

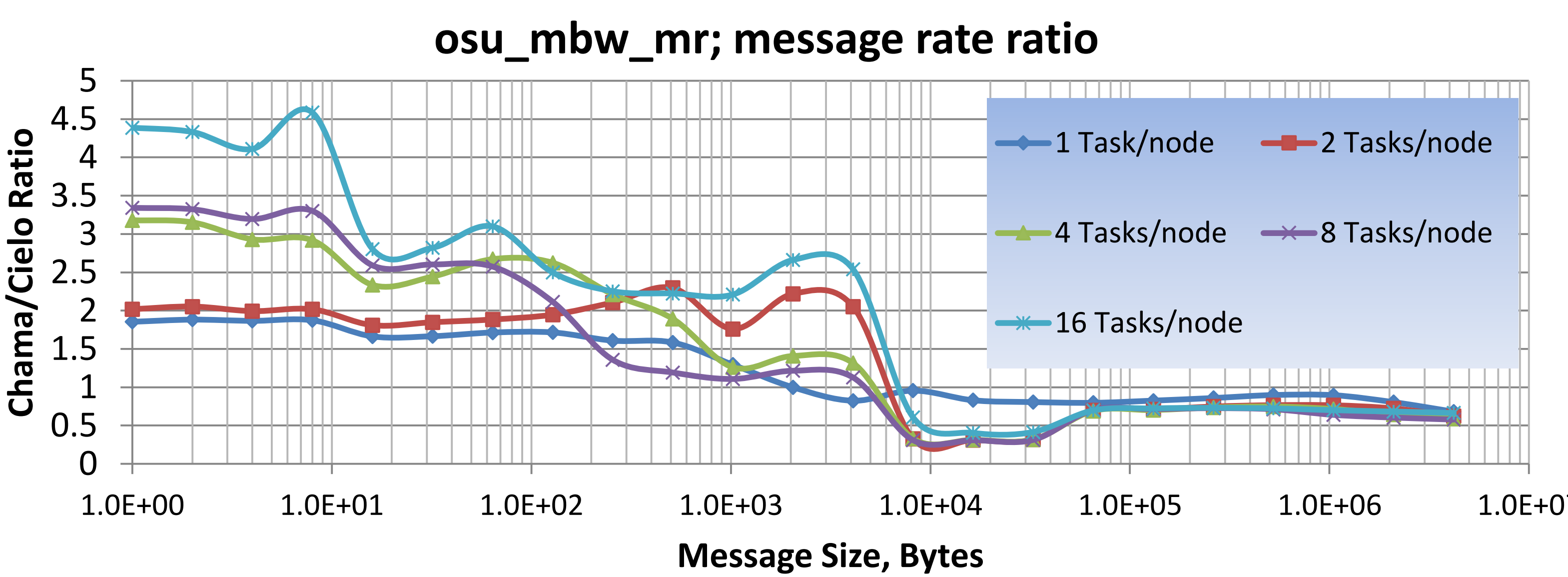




A FEW APPLICATIONS TO ILLUSTRATE MPI CHALLENGES SEEN

SIERRA/Aria- Implicit CFD; Key to performance -1k size MPI message rate; At max scale Chama is 2.6X better than Cielo; MPI message rate all important for solver based codes





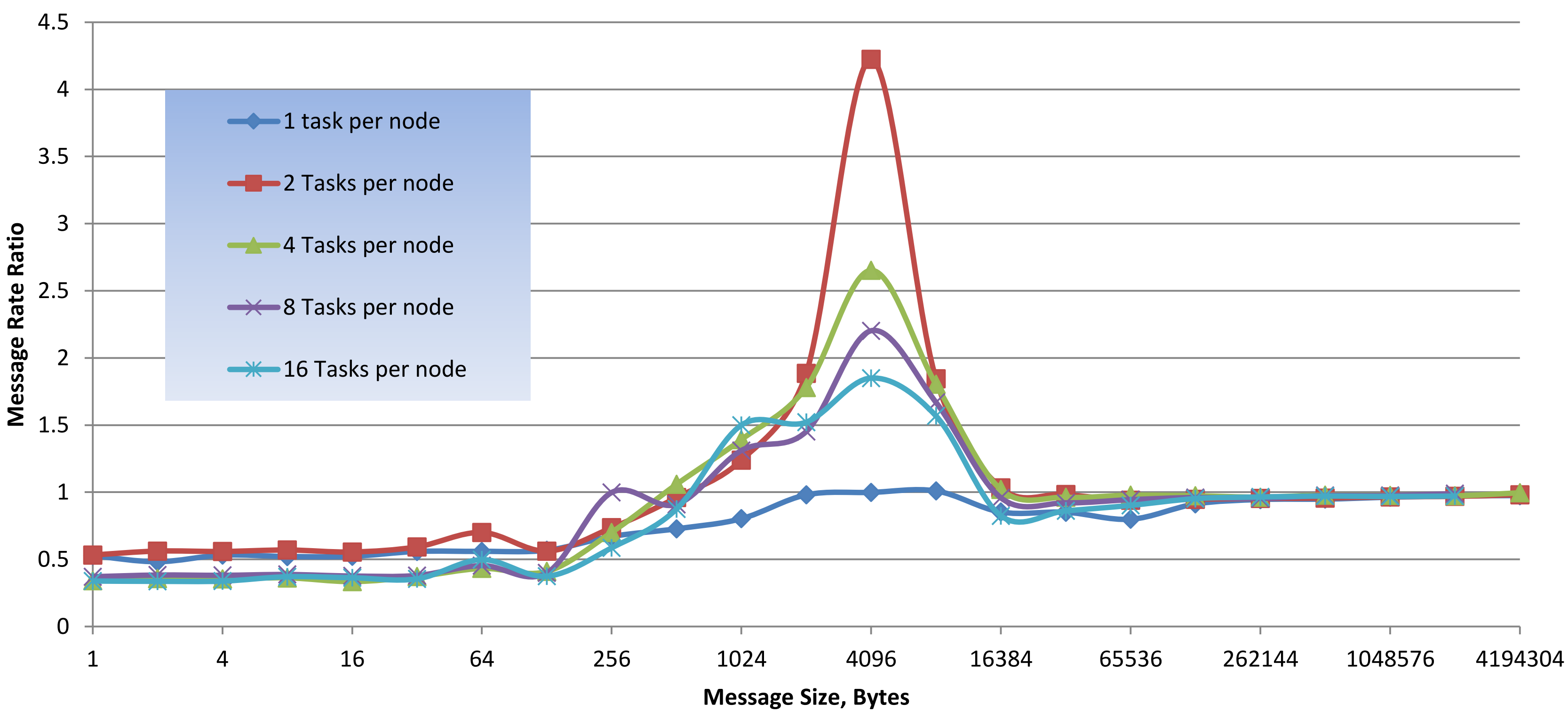
Sandia implicit codes benefit from MVAPICH

potential 1.5X gain in message rate at the 1k to 4k message size of interest

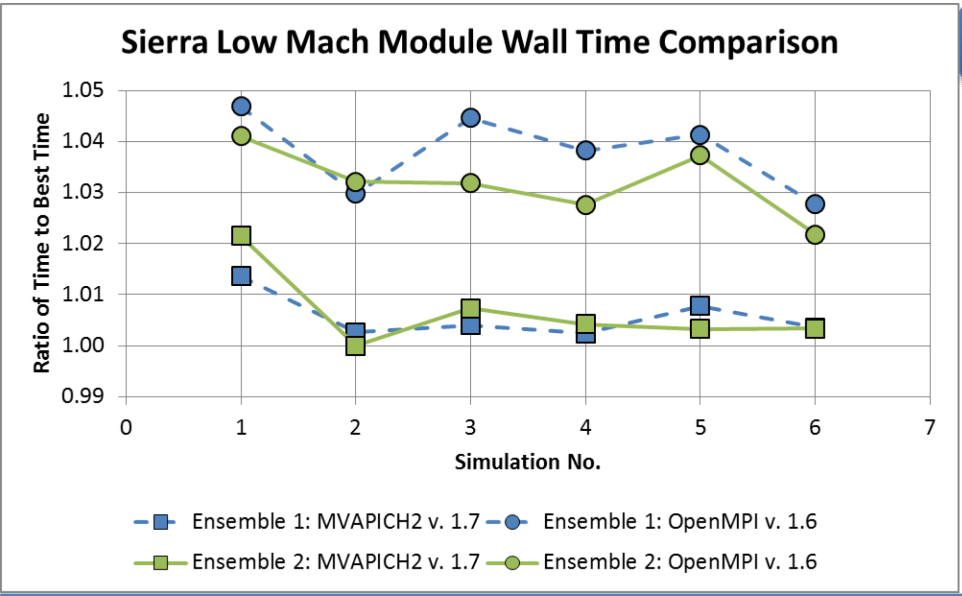
Intent is not to compare MPI libraries but find 'tunables' that can benefit users

used: mvapich2-intel-ofa/1.7 and openmpi-intel/1.6

osu_mbw_mr MVAPICH/OpenMPI ratio; Jemez



Another implicit CFD code showing benefit of MVAPICH Sierra NALU: Low Mach CFD



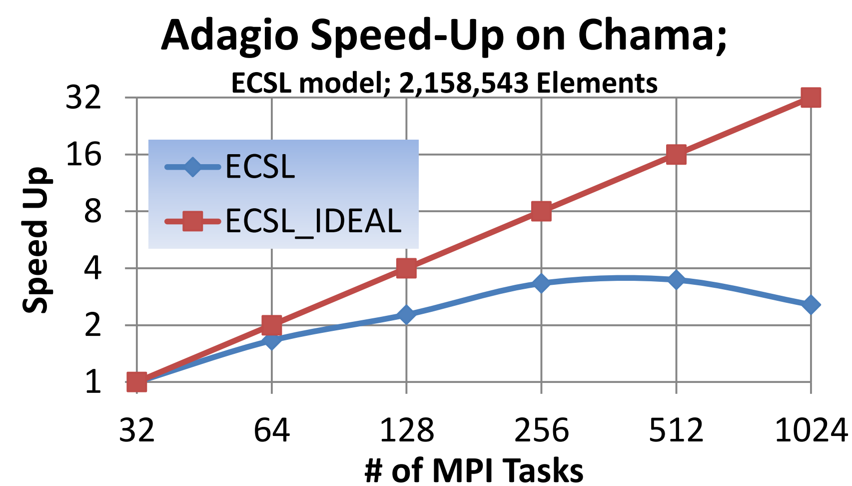
Simulation description & parameters:

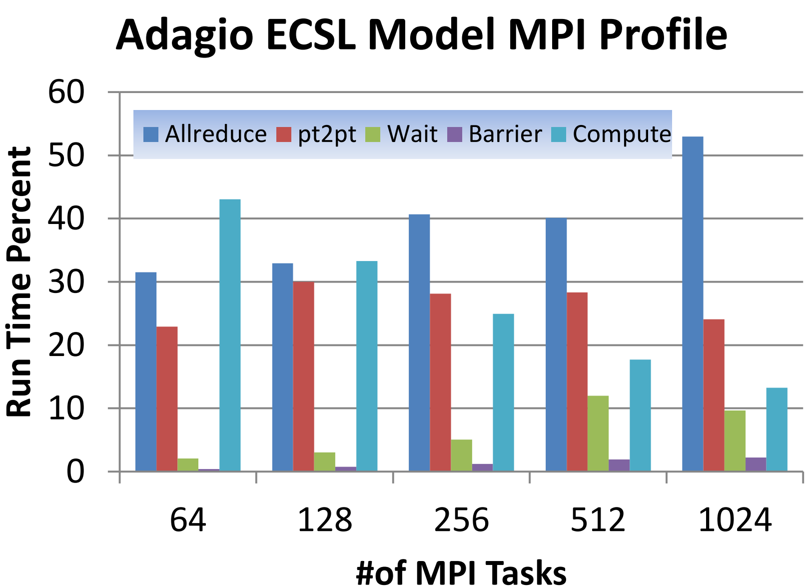
- Jet-in-crossflow simulation counterpart to Su & Mungal experimental results
- GMRES w/ML, Gram-Schmidt preconditioners
- Mesh: 152,463,520 hex-8 unstructured elem.
- 1,536 MPI ranks (1 rank per core) on Chama
- Each ensemble was run alternating MPI within the same node allocation, e.g., MVAPICH, OpenMPI, MVAPICH, OpenMPI,…

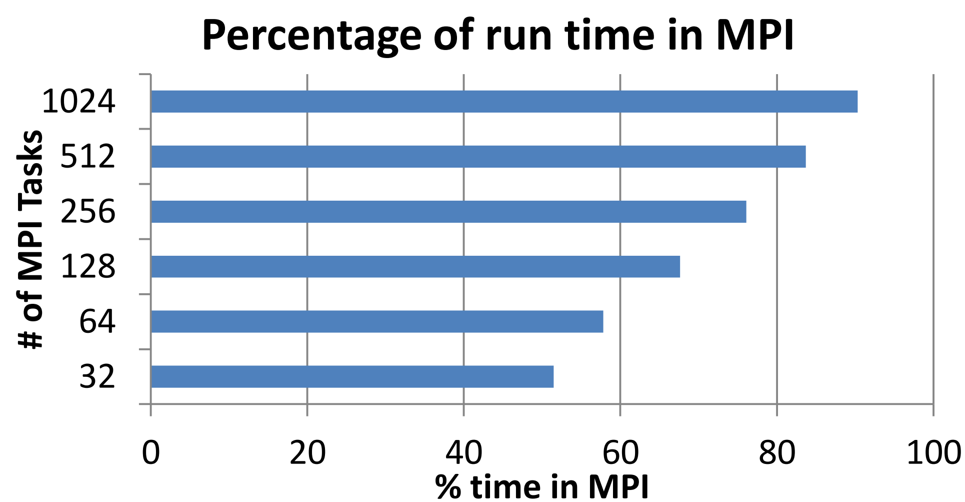
- Simulation observations:
  - **On average, MVAPICH is 2-4% faster than OpenMPI**
  - **MVAPICH shows lower run-time variability**
    - Variability with simulation runtimes **impedes** many things including performance regression testing
    - When the outliers in the run-time data are omitted MVAPICH's standard deviation is 2x-3x less than OpenMPI,
    - mpiP profile showed : ~60% of run time in MPI; 82% of MPI time in global operations (mostly MPI_Allreduce)
  - Asynchronous MPI global operations in MPI 3.0 could have a big impact
    - Potential reduction in variability due to OS noise/jitter
    - Asynchronous global operations may help reduce total time in Allreduce through better overlap of computations & communications

A most demanding application from an MPI perspective: Sierra/Adagio; Contact Algorithm

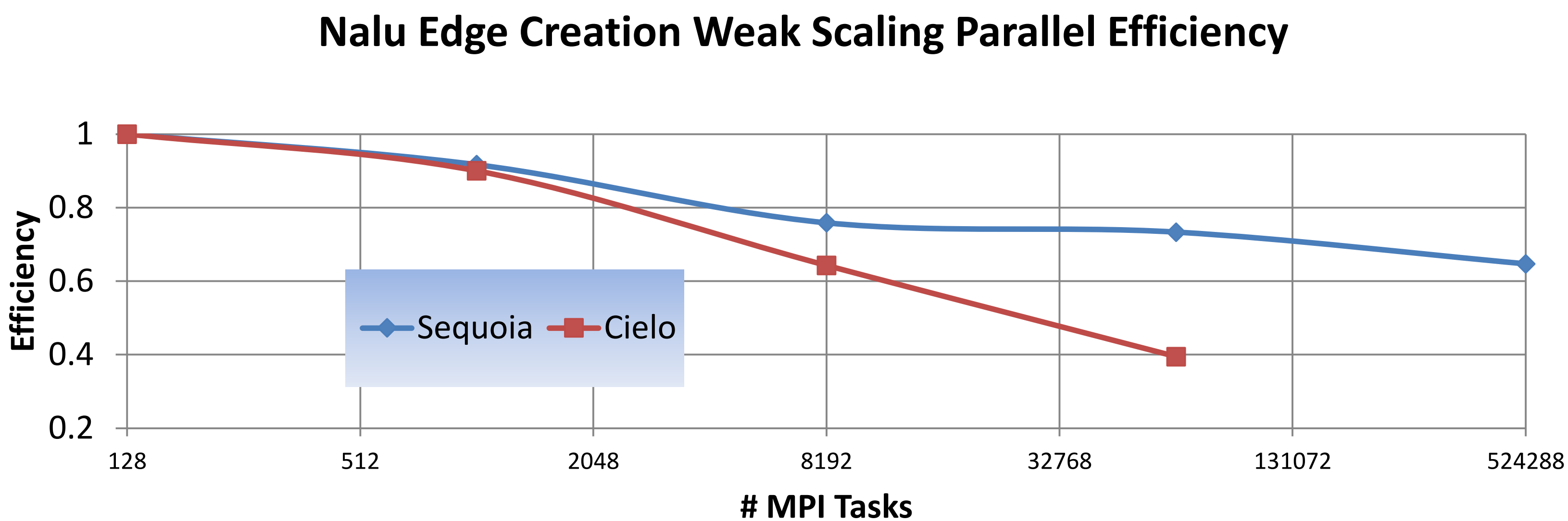
- ECSL Model
  - 2,158,543 elements Hex elements,
  - bolts, springs Contact,
  - multiple materials,
  - many parts, preload
- Sierra::Newton::Apst_Contact::contact_search takes 93% of run time out of which 76% is MPI;
- Contact search MPI calls are the dominant overhead.
- **MPI_Allreduce** time including ‘sync time’ due to load imbalance is the main scaling inhibitor. See chart below
- **Efficient Asynchronous MPI collectives are needed!!**

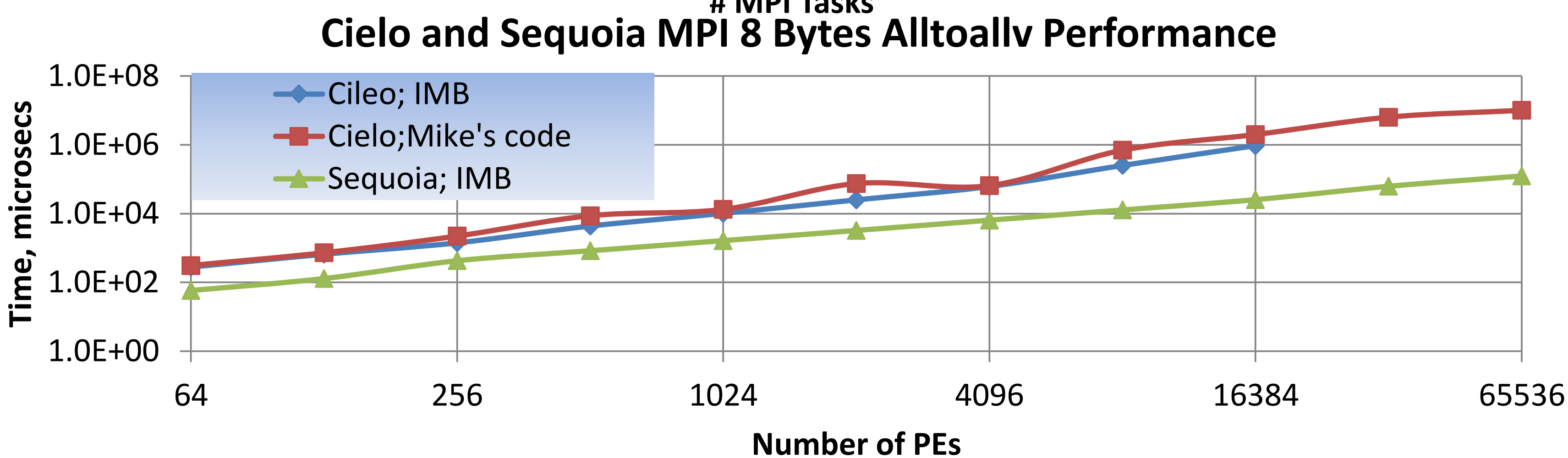






SNL Open Source CFD code NALU (openJet Problem) on Sequoia and Cielo; weak scaling. Edge Creation uses stk mesh; usage of MPI_Alltoallv Scaling inhibitor on Cielo Alltoallv; Sequoia is 5X @ 64PEs to 40X @16kPEs faster

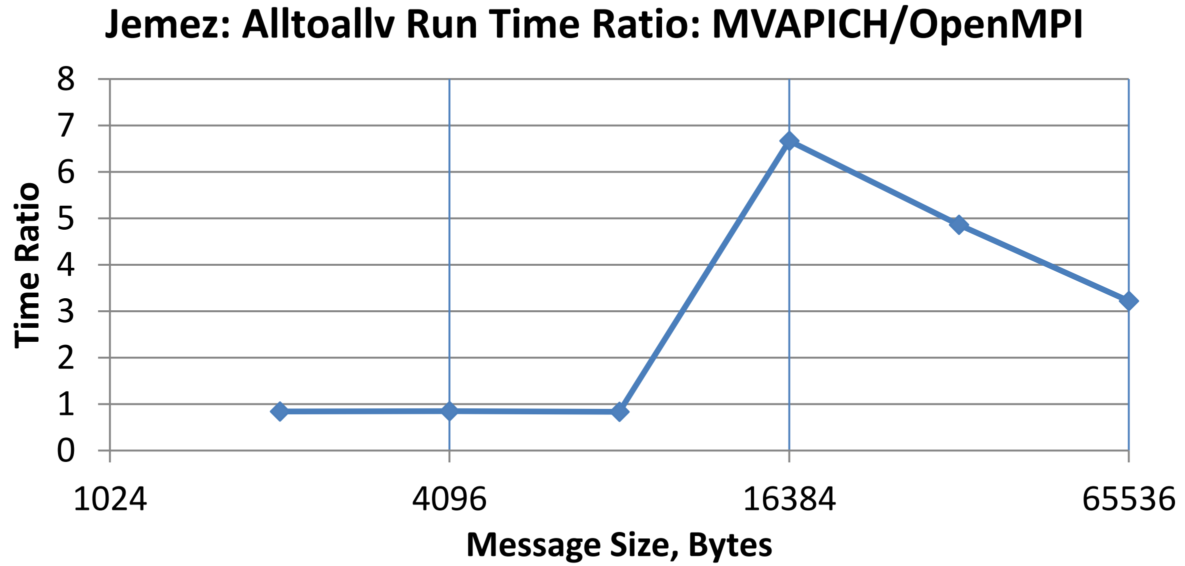




3DFFT with 2d (pencils) decomposition Key MPI compute kernel: Alltoallv

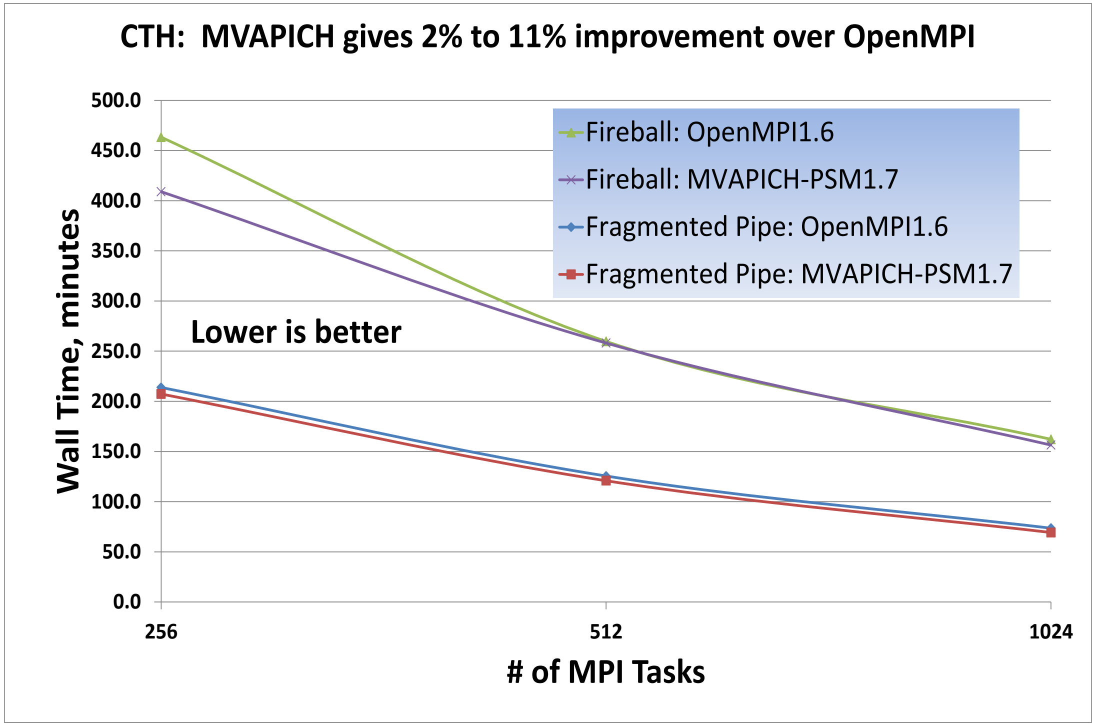
A typical case of interest with a grid of dimension of 1024x1024x1024; Forward and backward transform on 512 CORES (16x32 processor grid) on Jemez; **MVAPICH was about 20% slower then OpenMPI.** Alltoallv message exchange sizes are 32K Bytes. Would like to understand sudden poorer performance with MVAPICH at 16K + message size.

Are there ENV settings that would alleviate this problem?



CTH: MVAPICH gives 2% to 11% improvement over OpenMPI

- CTH is a multi-material, large deformation, strong shock wave, solid mechanics code developed at Sandia National Laboratories.
- The code is explicit and uses finite difference (volume) for the numerical simulation of the high-rate response of materials to impulsive loads.
- CTH modeling capabilities include shock waves, low mach flows, multi-phase materials, mixed-phase materials, elastic-plastic solids, visco-plastic solids, visco-elastic solids, fracture, failure, high explosive detonation and initiation in 1D, 2D, and 3D geometries.
- The code is widely used in the DOE and DOD communities in the development of explosives, blast and fragmenting warheads, kinetic energy penetrators, vehicle armor systems, and protective structures.
- Communication patterns are problem dependent, although in general, processors exchange information with up to six other processors in the domain and messaging is dominated by large messages (several MB in size), with some small message Allreduce at scale.



Two to three runs each of fireball and fragmented pipe were performed at three core counts (256, 512, 1024) for both OpenMPI and MVAPICH. The scenarios were designed so that run times ranged between 1 hour and 8 hours. The minimum run times for OpenMPI and MVAPICH were then compared.

The "fireball" CTH simulation performed in this investigation is a 3D shock physics problem with AMR. A computational model was used to calculate the three-dimensional evolution of a nuclear fireball (shown in red, bottom left). CTH version 10.3p (Feb. 2013) was used for this calculation. This particular CTH problem was chosen for the investigation because it is representative of a large class of shock physics problems that run regularly on Sandia production platforms. This problem is a production simulation capable of running at a continuum of core counts. We used 256, 512, and 1024 cores for this investigation.

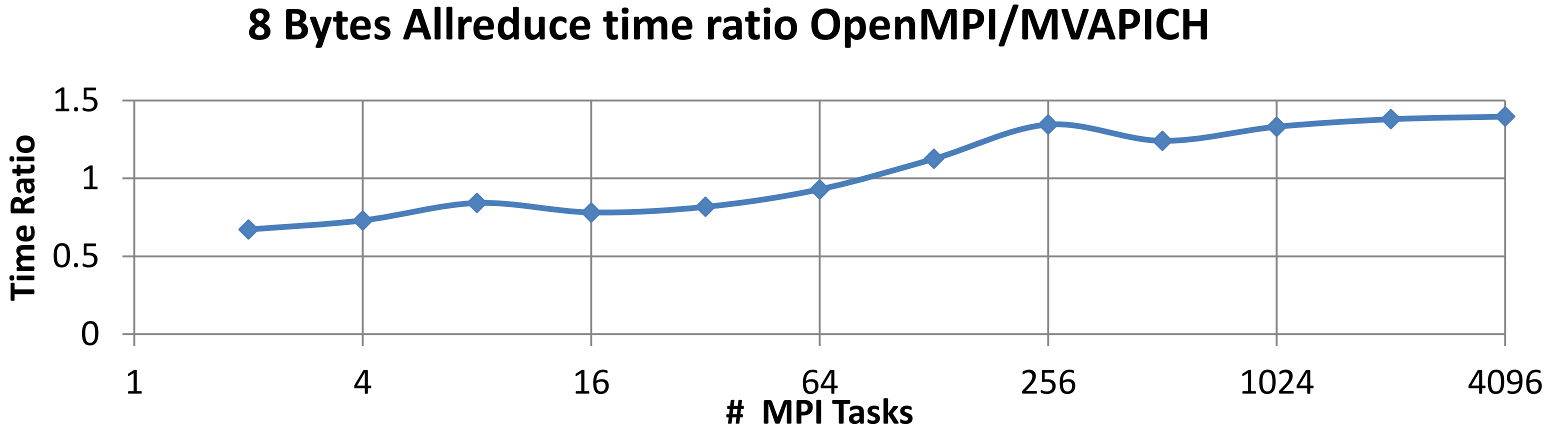
The "fragmented pipe" CTH simulation performed in this investigation is a 3D shock physics problem with AMR. A computational model was used to calculate the three-dimensional evolution of a fragmented pipe blowing apart in firecracker-like fashion (shown bottom right). CTH version 10.3p (Feb. 2013) was used for this calculation. This particular CTH problem was chosen for the investigation because it is representative of a large class of shock physics problems that run regularly on Sandia production platforms. We used 256, 512, and 1024 cores for this investigation.

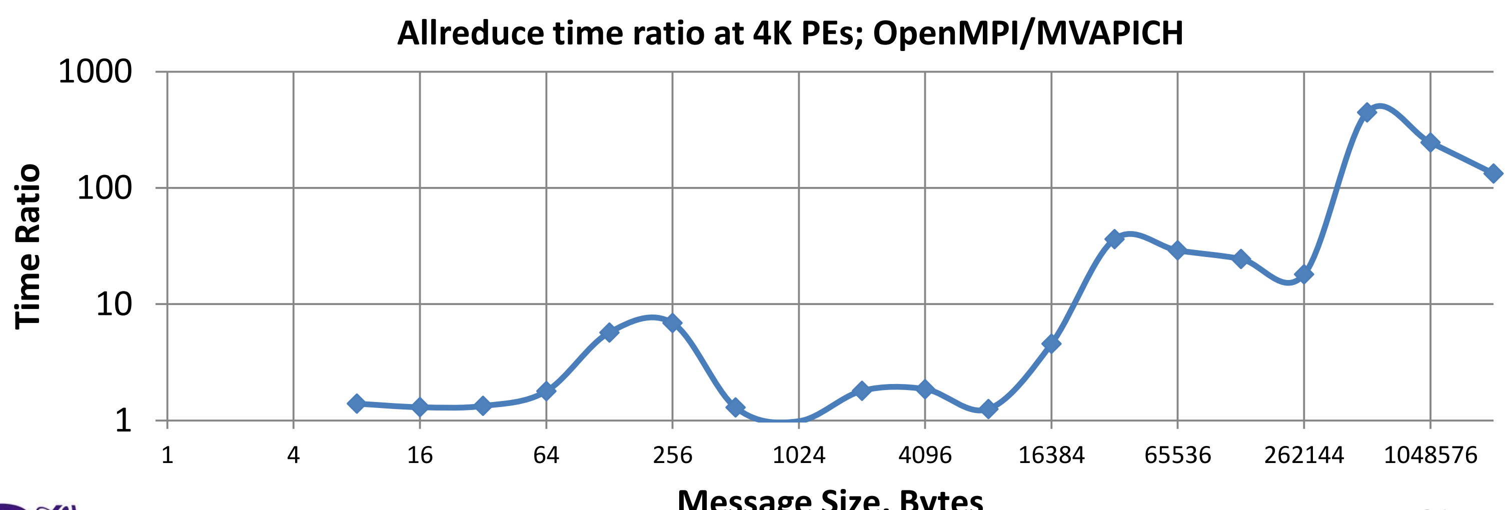
Allreduce used quite heavily with Sandia apps;

potential Allreduce time reduction for 128 to 4k PEs.

Intent is not to compare MPI libraries but find 'tunables' that can benefit users

used: mvapich2-intel-ofa/1.7 and openmpi-intel/1.6





Conclusions

- Our Goal: Use of MPI library that is robust and performs well; help users with optimizing performance for production runs on all our HPC systems
- Shared a few application use cases that stress MPI performance; e.g. collectives and 1k-4k messages
- Some benchmark data collected point to potential benefit for application performance with MVAPICH
- Would like from MVAPICH group & from MUG meeting
  - Suggestions on how users can extract best performance
  - Optimal environment variable settings
  - MPI 3.0 release and potential benefits
  - Collaboration with Sandia on Advanced architectures and use of Mantevo mini-apps