

Towards Petascale Computing for Science

Horst Simon

Lenny Oliker, David Skinner, and Erich Strohmaier

Lawrence Berkeley National Laboratory

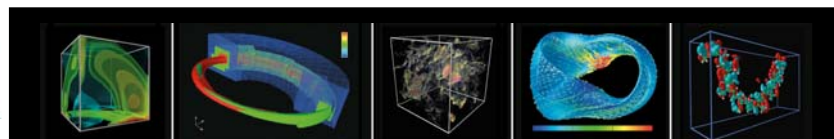
The Salishan Conference on High-Speed Computing

April 19, 2005



Outline

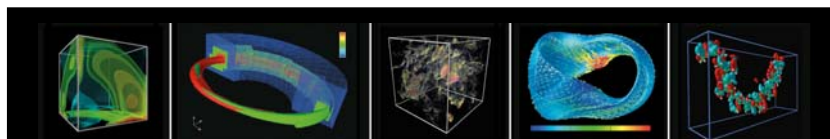
- Science Driven Architecture
- Performance on today's (2004 - 2005) platforms
- Challenges with scaling to the Petaflop/s level
- Two tools that can help: IPM and APEX/MAP



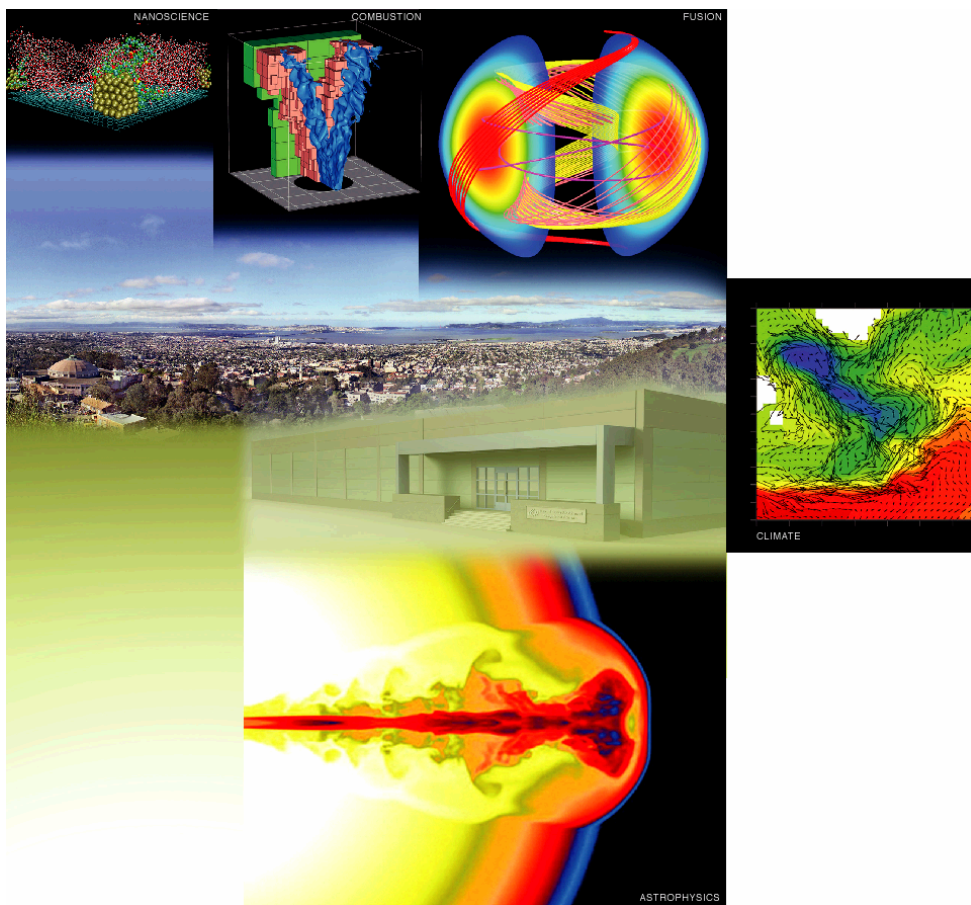
Scientific Applications and Underlying Algorithms Drive Architectural Design

- 50 Tflop/s - 100 Tflop/s **sustained** performance on applications of national importance
- Process:
 - identify applications
 - identify computational methods used in these applications
 - identify architectural features most important for performance of these computational methods

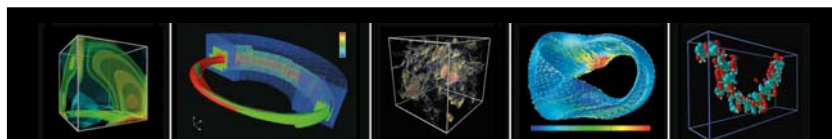
Reference: Creating Science-Driven Computer Architecture: A New Path to Scientific Leadership, (Horst D. Simon, C. William McCurdy, William T.C. Kramer, Rick Stevens, Mike McCoy, Mark Seager, Thomas Zacharia, Jeff Nichols, Ray Bair, Scott Studham, William Camp, Robert Leland, John Morrison, Bill Feiereisen), Report LBNL-52713, May 2003. (see www.nersc.gov/news/reports/HECRTF-V4-2003.pdf)



Capability Computing Applications in DOE/SC



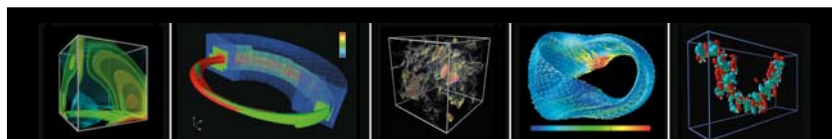
- Accelerator modeling
- Astrophysics
- Biology
- Chemistry
- Climate and Earth Science
- Combustion
- Materials and Nanoscience
- Plasma Science/Fusion
- QCD
- Subsurface Transport



Capability Computing Applications in DOE/SC (cont.)

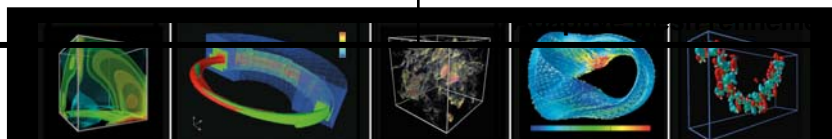
These applications and their computing needs have been well-studied in the past years:

- “A Science-Based Case for Large-scale Simulation”, David Keyes, Sept. 2004 (<http://www.pnl.gov/scales>).
- “Validating DOE’s Office of Science “Capability” Computing Needs”, E. Barsis, P. Mattern, W. Camp, R. Leland, SAND2004-3244, July 2004.



Science Breakthroughs Enabled by Leadership Computing Capability

Science Areas	Goals	Computational Methods	Breakthrough Target (50-100 Tflop/s)
Nanoscience	Simulate the synthesis and predict the properties of multi-component nanosystems	Quantum molecular dynamics Quantum Monte Carlo Iterative eigensolvers Dense linear algebra Parallel 3D FFTs	Simulate nanostructures with hundreds to thousands of atoms as well as transport and optical properties and other parameters
Combustion	Predict combustion processes to provide efficient, clean and sustainable energy	Explicit finite difference Implicit finite difference Zero-dimensional physics Adaptive mesh refinement Lagrangian particle methods	Simulate laboratory scale flames with high fidelity representations of governing physical processes
Fusion	Understand high-energy density plasmas and develop an integrated simulation of a fusion reactor	Multi-physics, multi-scale Particle methods Regular and irregular access Nonlinear solvers Adaptive mesh refinement	Simulate the ITER reactor
Climate	Accurately detect and attribute climate change, predict future climate and engineer mitigation strategies	Finite difference methods FFTs Regular and irregular access Simulation ensembles	Perform a full ocean/ atmosphere climate model with 0.125 degree spacing, with an ensemble of 8-10 runs
Astrophysics	Determine through simulations and analysis of observational data the origin, evolution and fate of the universe, the nature of matter and energy, galaxy and stellar evolutions	Multi-physics, multi-scale Dense linear algebra Parallel 3D FFTs Spherical transforms Particle methods	Simulate the explosion of a supernova with a full 3D model



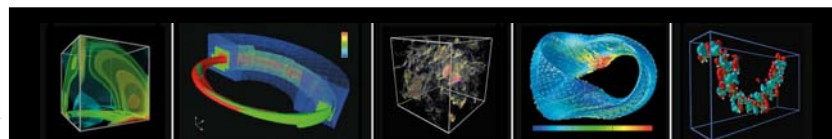
Opinion Slide

One reason why we have failed so far to make a good case for increased funding in supercomputing is that we have not yet made a compelling science case.

A better example: “The Quantum Universe”

“It describes a revolution in particle physics and a quantum leap in our understanding of the mystery and beauty of the universe.”

<http://interactions.org/quantumuniverse/>



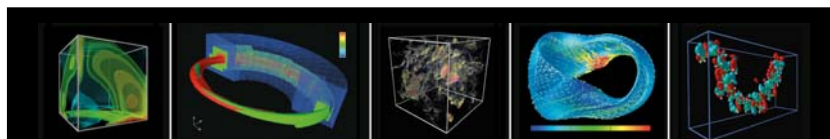
How Science Drives Architecture

State-of-the-art computational science requires increasingly diverse and complex algorithms

Science Areas	Multi-Physics and Multi-Scale	Dense Linear Algebra	FFTs	Particle Methods	AMR	Data Parallelism	Irregular Control Flow
Nanoscience	X	X	X	X		X	X
Combustion	X			X	X	X	X
Fusion	X	X		X	X	X	X
Climate	X		X		X	X	X
Astrophysics	X	X	X	X	X	X	X

Only balanced systems that can perform well on a variety of problems will meet future scientists' needs!

Data-parallel and scalar performance are both important

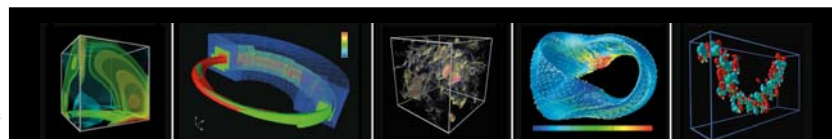


Phil Colella's “Seven Dwarfs”

Algorithms that consume the bulk of the cycles of current high-end systems in DOE:

- Structured Grids
- Unstructured Grids
- Fast Fourier Transform
- Dense Linear Algebra
- Sparse Linear Algebra
- Particles
- Monte Carlo

(Should also include optimization / solution of nonlinear systems, which at the high end is something one uses mainly in conjunction with the other seven)

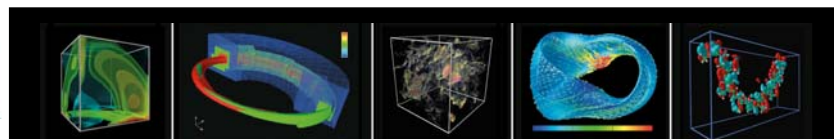


“Evaluation of Leading Superscalar and Vector Architectures for Scientific Computations”

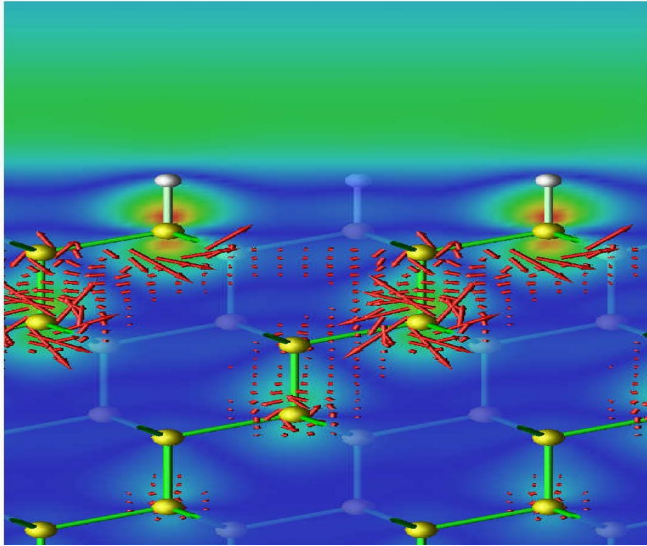
Leonid Oliker, Andrew Canning, Jonathan Carter
LBNL

Stephane Ethier
PPPL

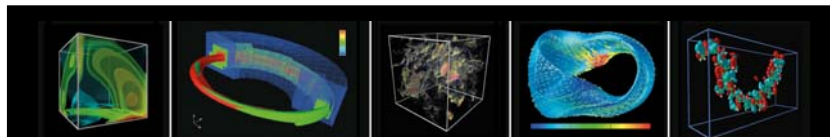
(see SC04 paper at <http://crd.lbl.gov/~oliker/>)



Material Science: PARATEC

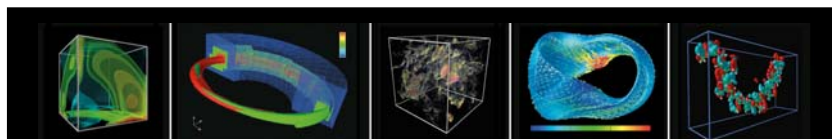


- PARATEC performs first-principles quantum mechanical total energy calculation using pseudopotentials & plane wave basis set
 - Density Functional Theory to calc structure & electronic properties of new materials
 - DFT calc are one of the largest consumers of supercomputer cycles in the world
-
- PARATEC uses all-band CG approach to obtain wavefunction of electrons
 - Part of calc. in real space other in Fourier space using specialized 3D FFT to transform wavefunction
 - Generally obtains high percentage of peak on different platforms
 - Developed with Louie and Cohen's groups (UCB, LBNL), Raczkowski



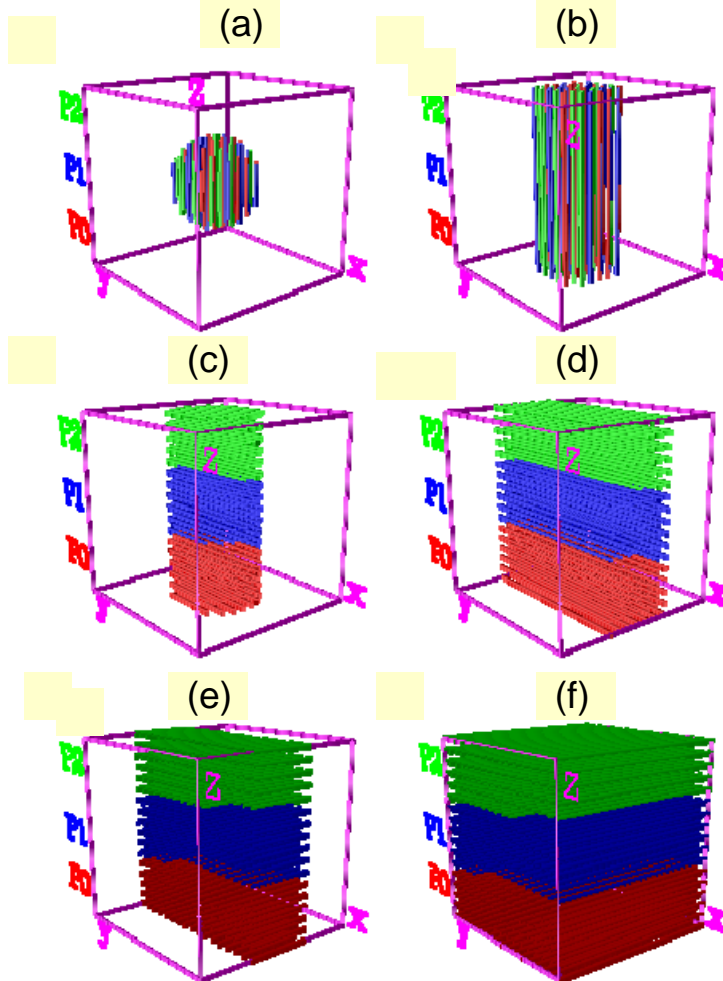
PARATEC: Code Details

- Code written in F90 and MPI (~50,000 lines)
- 33% 3D FFT, 33% BLAS3, 33% Hand coded F90
- Global Communications in 3D FFT (Transpose)
- 3D FFT handwritten, minimize comms. reduce latency (written on top of vendor supplied 1D complex FFT)
- Code has setup phase then performs many (~50) CG steps to converge the charge density of the system (data on speed is for 5CG steps, does not include setup)



PARATEC: 3D FFT

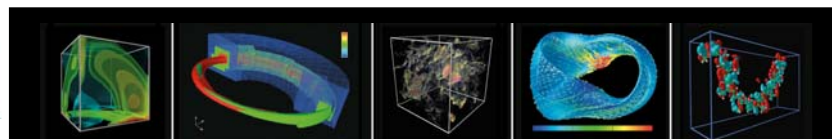
FIGURE



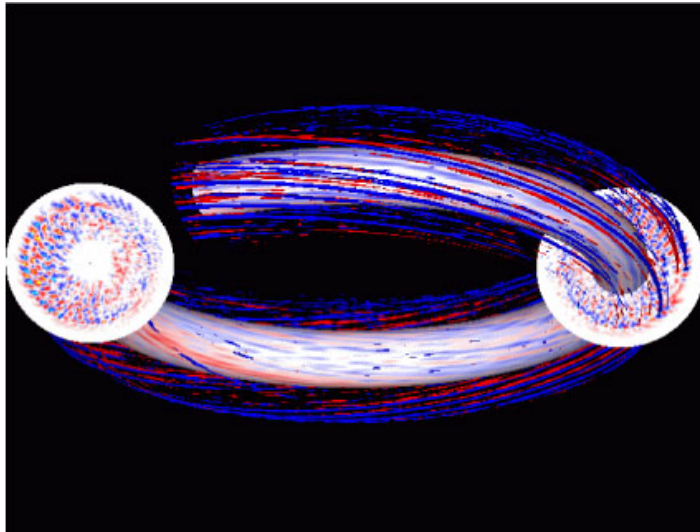
- 3D FFT done via 3 sets of 1D FFTs and 2 transposes
- Most communication in global transpose (b) to (c) little communication (d) to (e)
- Many FFTs done at the same time to avoid latency issues
- Only non-zero elements communicated/calculated
- Much faster than vendor supplied 3D-FFT

PARATEC: Performance

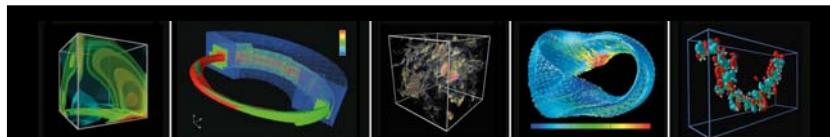
Data Size	P	Power 3		Power4		Altix		ES		X1	
		Gflops/ P	% peak	Gflops/ P	% peak	Gflops/ P	% peak	Gflops/ P	% peak	Gflops/ P	% peak
432 Atom	32	0.95	63%	2.0	39%	3.7	62%	4.7	60%	3.0	24%
	64	0.85	57%	1.7	33%	3.2	54%	4.7	59%	2.6	20%
	128	0.74	49%	1.5	29%	---	---	4.7	59%	1.9	15%
	256	0.57	38%	1.1	21%	---	---	4.2	52%	---	---
	512	0.41	28%	---	---	---	---	3.4	42%	---	---
686 Atom	128							4.9	62%	3.0	24%
	256							4.6	57%	1.3	10%



Magnetic Fusion: GTC



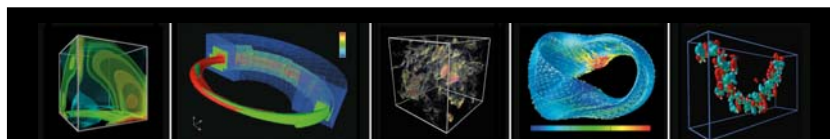
- Gyrokinetic Toroidal Code: transport of thermal energy (plasma microturbulence)
- Goal magnetic fusion is burning plasma power plant producing cleaner energy
- GTC solves gyroaveraged gyrokinetic system w/ particle-in-cell approach (PIC)
- PIC scales N instead of N^2 – particles interact w/ electromag field on grid
- Allows solving equation of particle motion with ODEs (instead of nonlinear PDEs)
- Main computational tasks:
 - Scatter: deposit particle charge to nearest grid points
 - Solve the Poisson eqn to get potential at each grid point
 - Gather: Calc force on each particle based on neighbors potential
 - Move particles by solving eqn of motion along the characteristics
 - Find particles moved outside local domain and update
- Developed at Princeton Plasma Physics Laboratory, vectorized by Stephane Ethier



GTC: Performance

Number r Particles	P	Power 3		Power4		Altix		ES		X1	
		Gflops /P	%pea k	Gflops/ P	%pea k	Gflops/ P	%pea k	Gflops /P	%pea k	Gflops/ P	%pea k
10/cell 20M	32	0.13	9%	0.29	5%	0.29	5%	1.15	14%	1.00	8%
	64	0.13	9%	0.32	5%	0.26	4%	1.00	13%	0.80	6%
100/cell 200M	32	0.13	9%	0.29	5%	0.33	6%	1.62	20%	1.50	12%
	64	0.13	9%	0.29	5%	0.31	5%	1.56	20%	1.36	11%
	1024	0.06	4%								

GTC is now scaling to 2048 processors on the ES for a total of 3.7 TFlops/s



Issues in Applications Scaling

Applications Status in 2005

- A few Teraflop/s sustained performance
- Scaled to 512 - 1024 processors

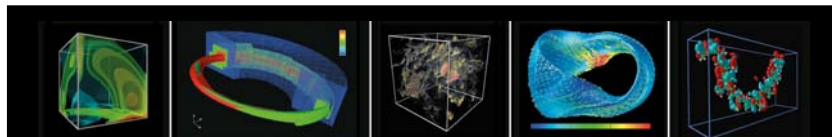
Applications on Petascale Systems need to deal with

- 100,000 processors (assume nominal Petaflop/s system with 100,000 processors of 10 Gflop/s each)
- Multi-core processors
- Topology sensitive interconnection network



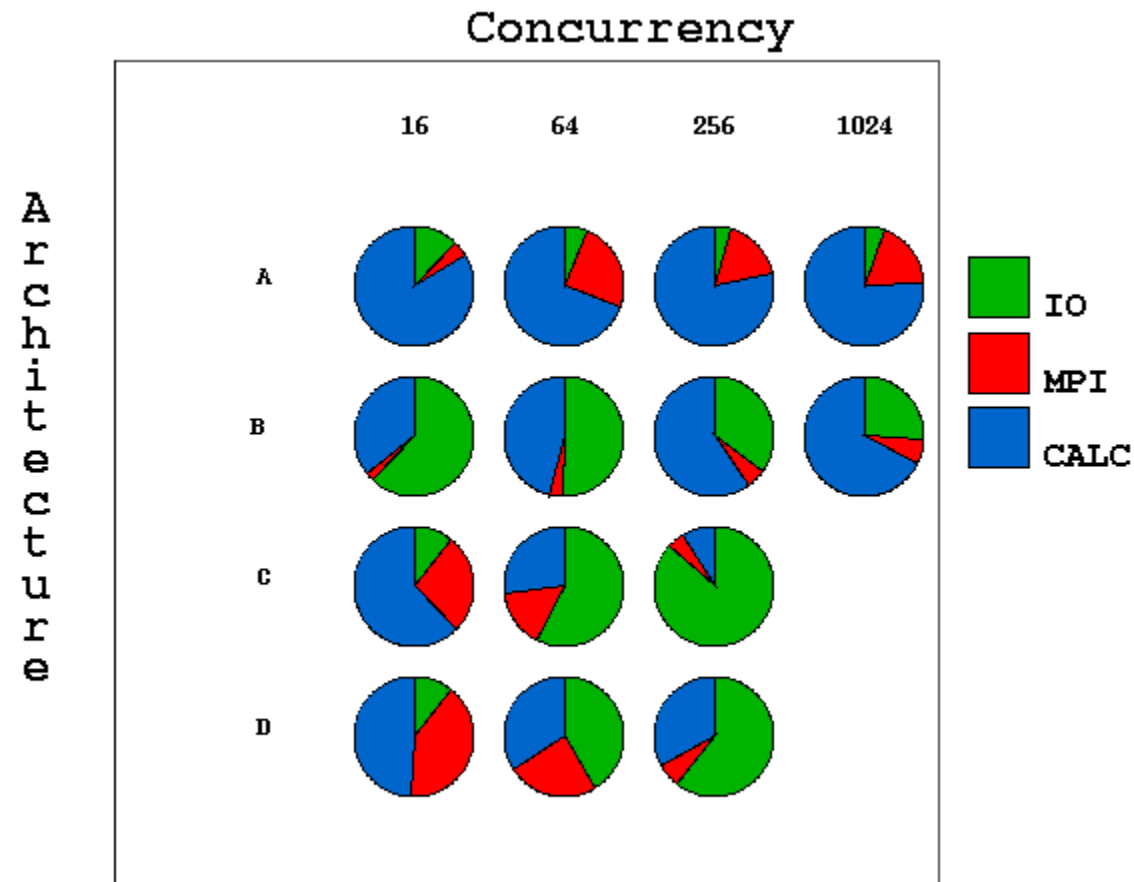
Integrated Performance Monitoring (IPM)

- brings together multiple sources of performance metrics into a single profile that characterizes the overall performance and resource usage of the application
- maintains low overhead by using a unique hashing approach which allows a fixed memory footprint and minimal CPU usage
- open source, relies on portable software technologies and is scalable to thousands of tasks
- developed by David Skinner at NERSC (see <http://www.nersc.gov/projects/ipm/>)

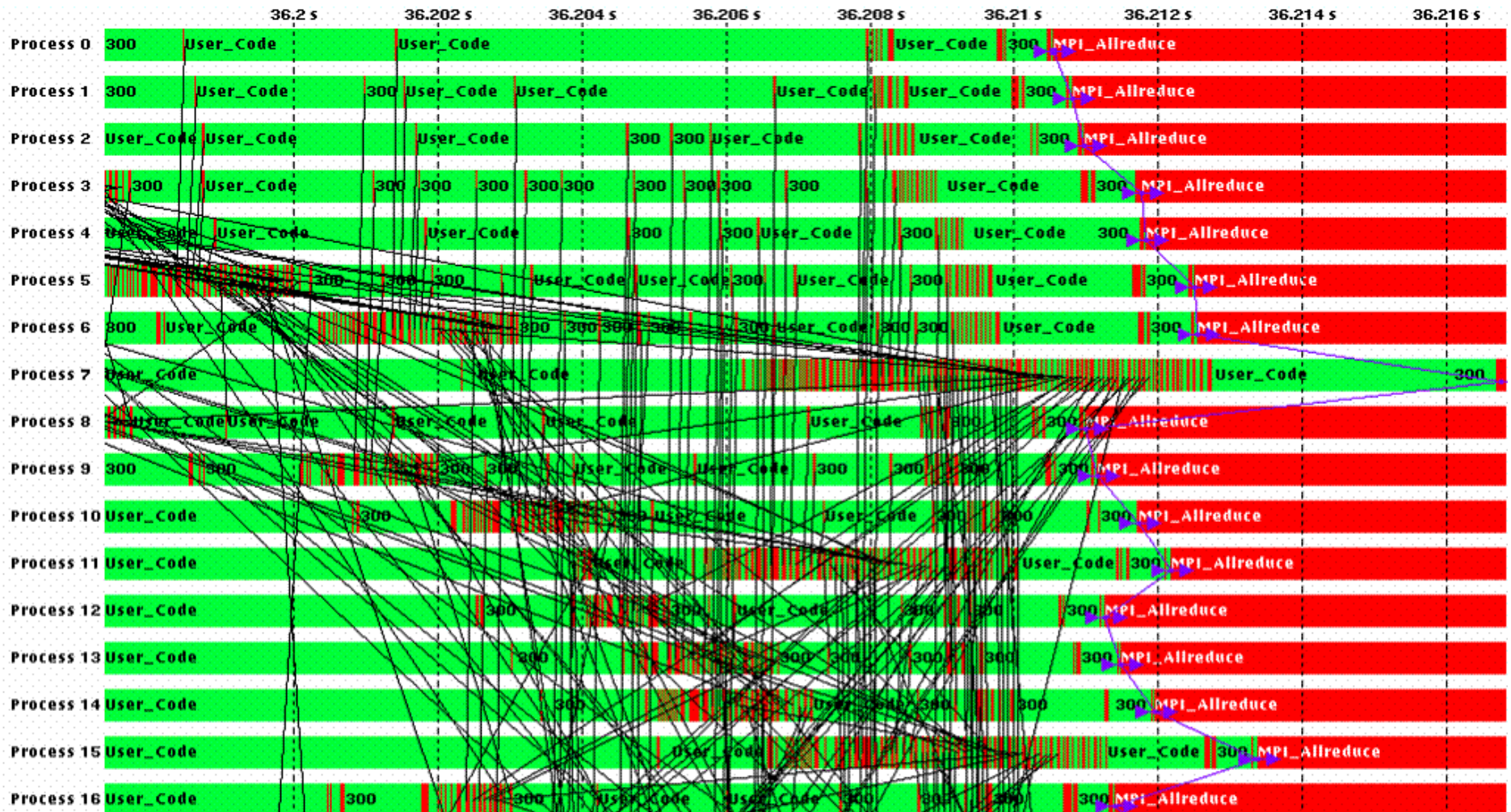


Scaling Portability: Profoundly Interesting

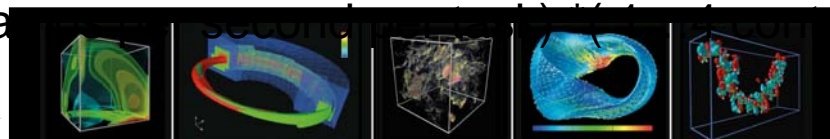
A high level description of the performance of a well known cosmology code on four well known architectures.



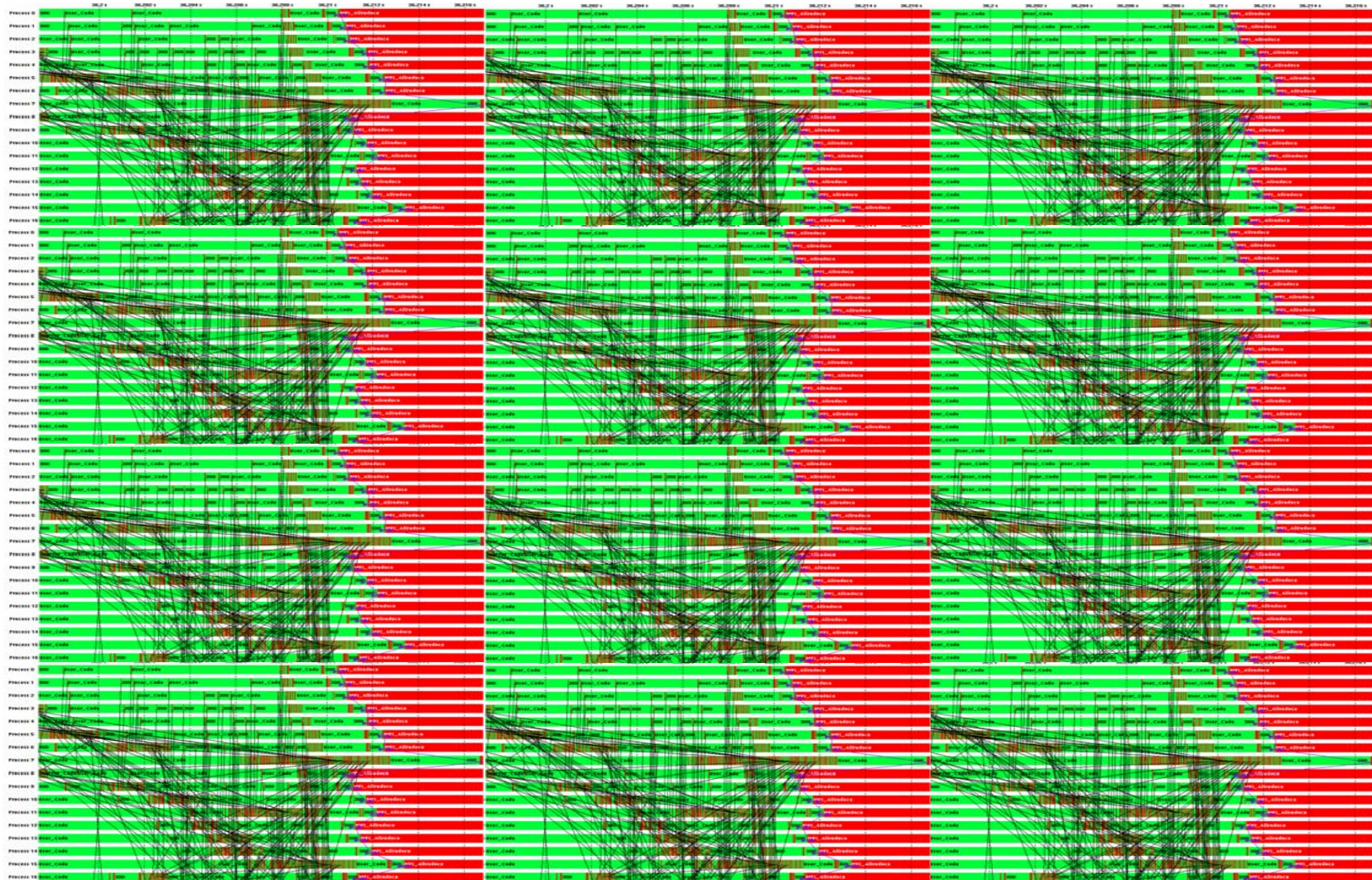
16 Way for 4 seconds



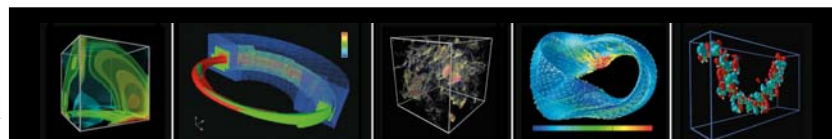
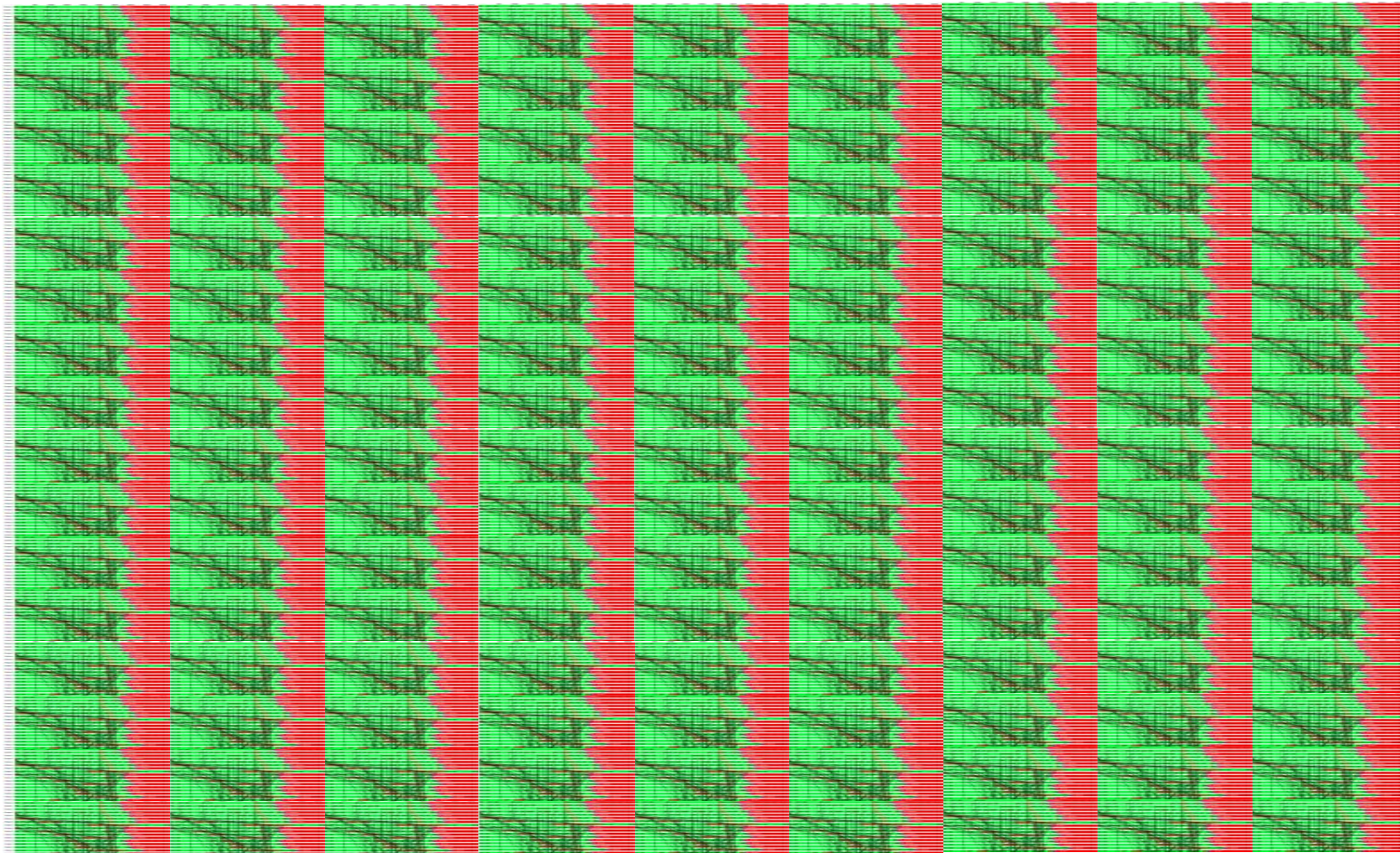
(About 20 timesteps per second, 16 processes, 16 context variables)



64 way for 12 seconds

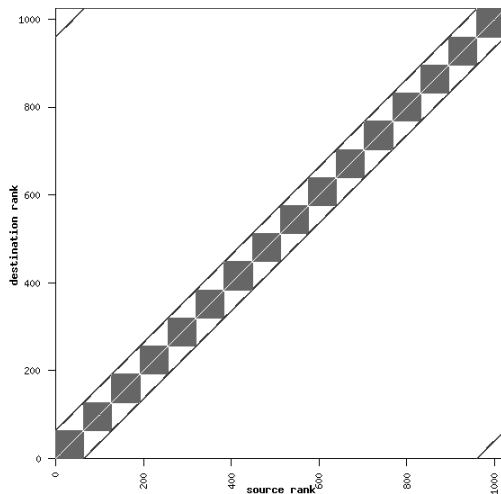


256 Way for 36 Seconds

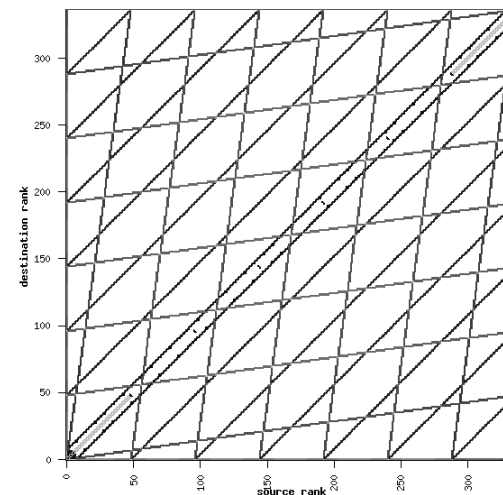


Application Topology

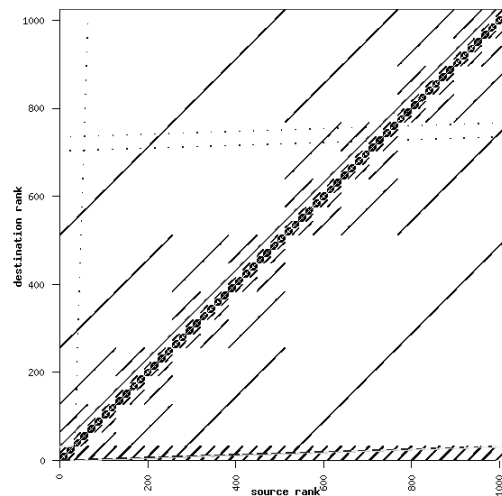
1024 way MILC



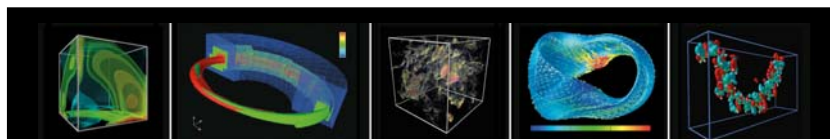
336 way FVCAM



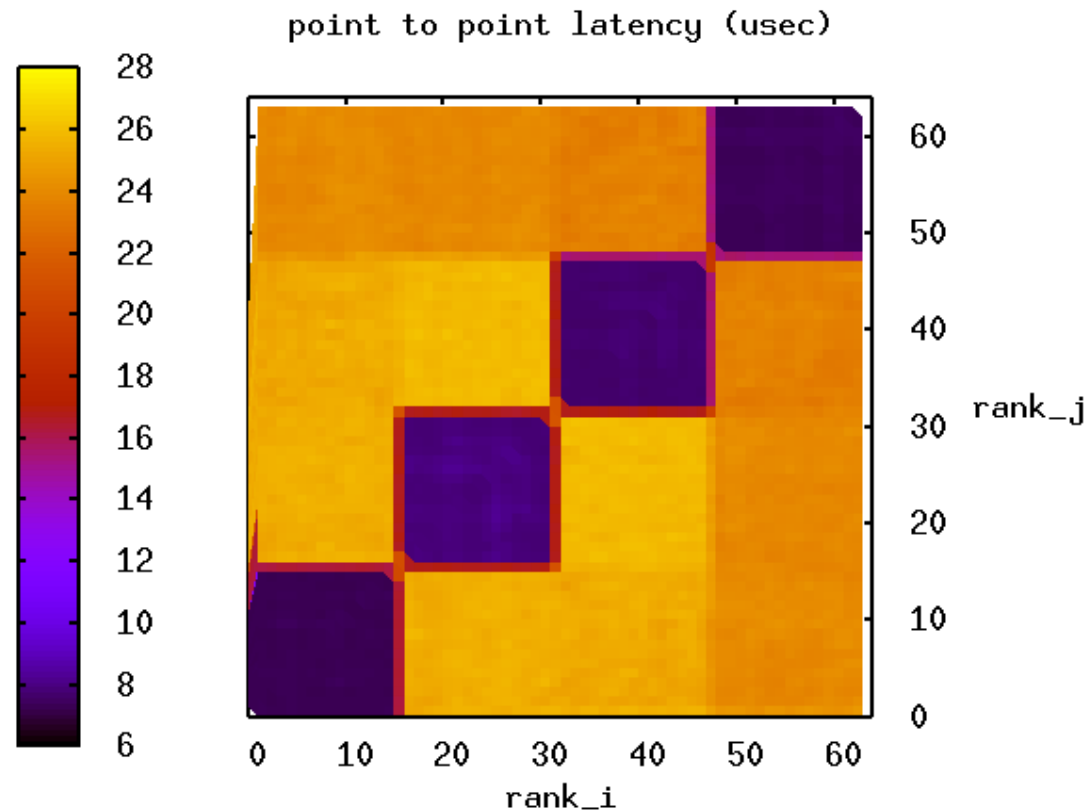
1024 way MADCAP



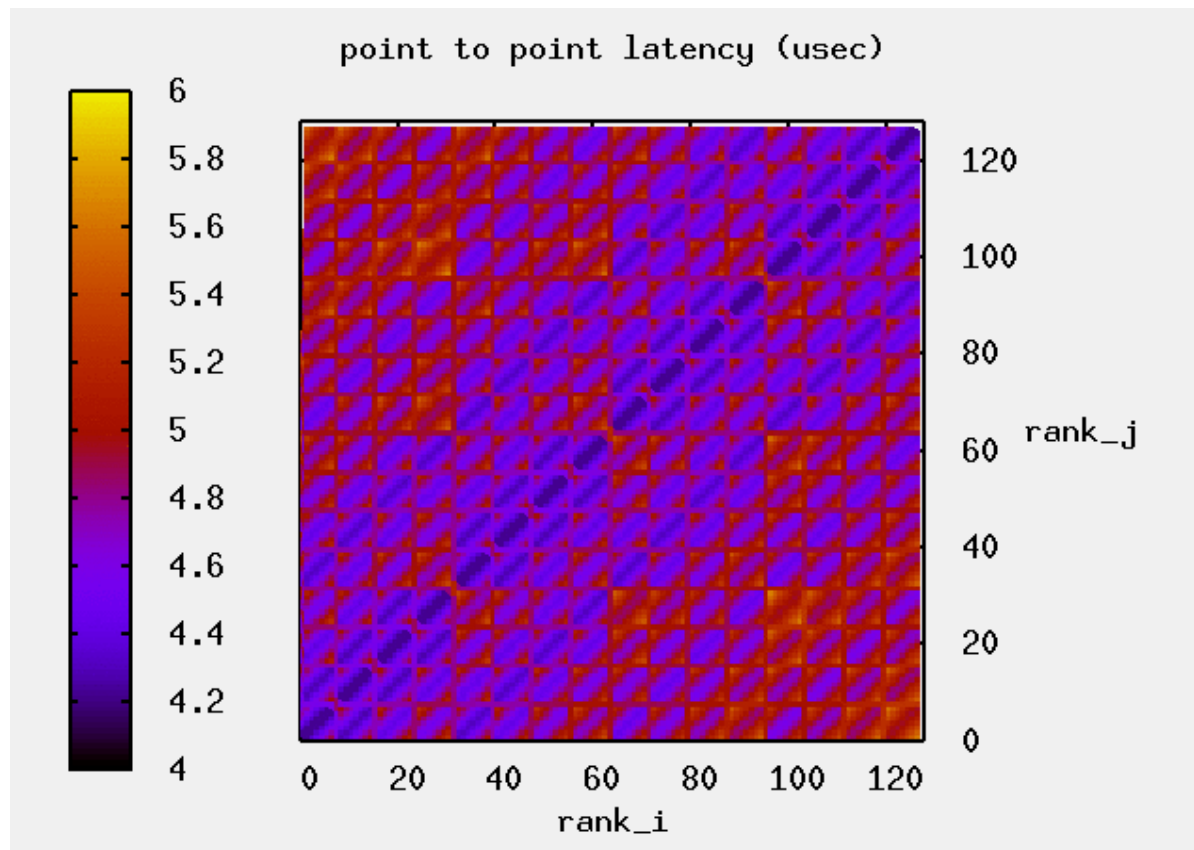
If the interconnect is topology sensitive, mapping will become an issue (again)



Interconnect Topology

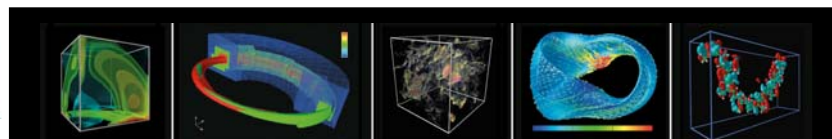
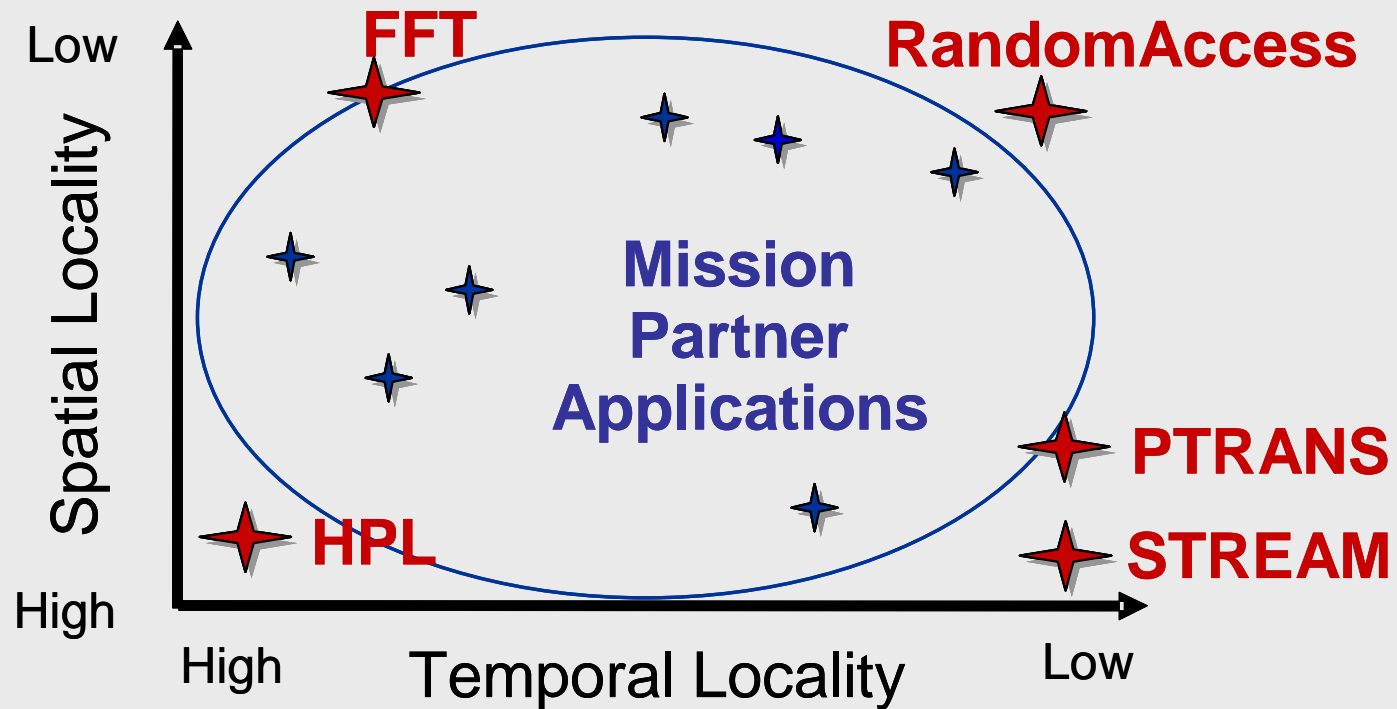


Interconnect Topology



DARPA HPCS will characterize applications

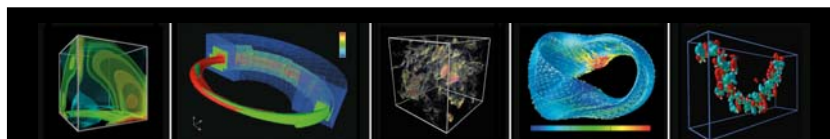
HPCS Program Goals & The HPCchallenge Benchmarks



APEX-Map: A Synthetic Benchmark to Explore the Space of Application Performances

Erich Strohmaier, Hongzhang Shan
Future Technology Group, LBNL
EStrohmaier@lbl.gov

Co-sponsored by DOE/SC and NSA



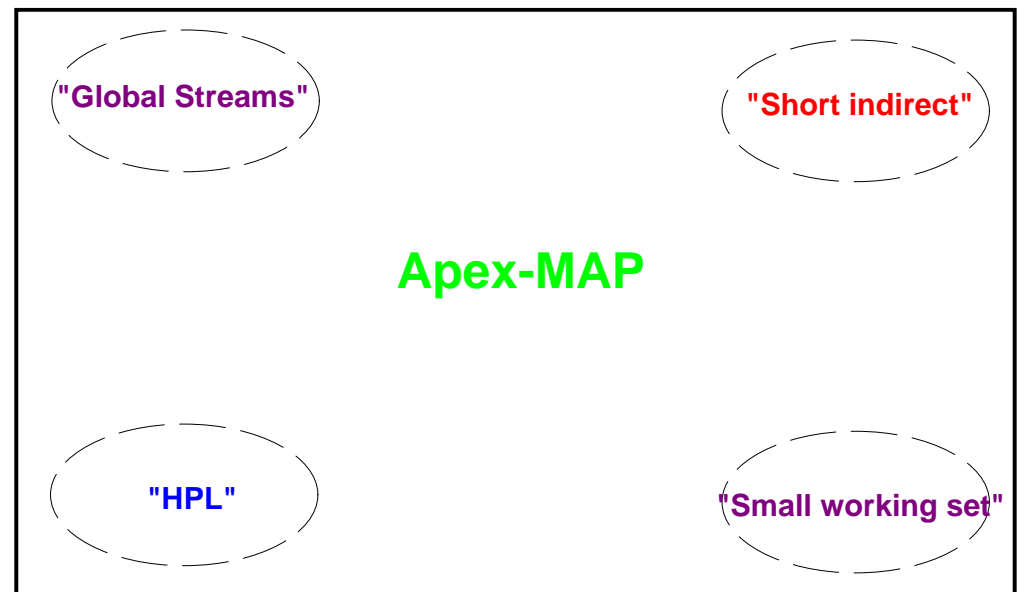
Apex-MAP characterizes architectures through a synthetic benchmark

Temporal Locality

1=Low

1/Re-use

0 = High

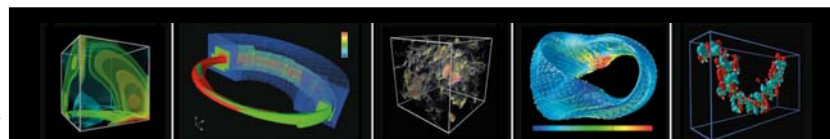


0 = High

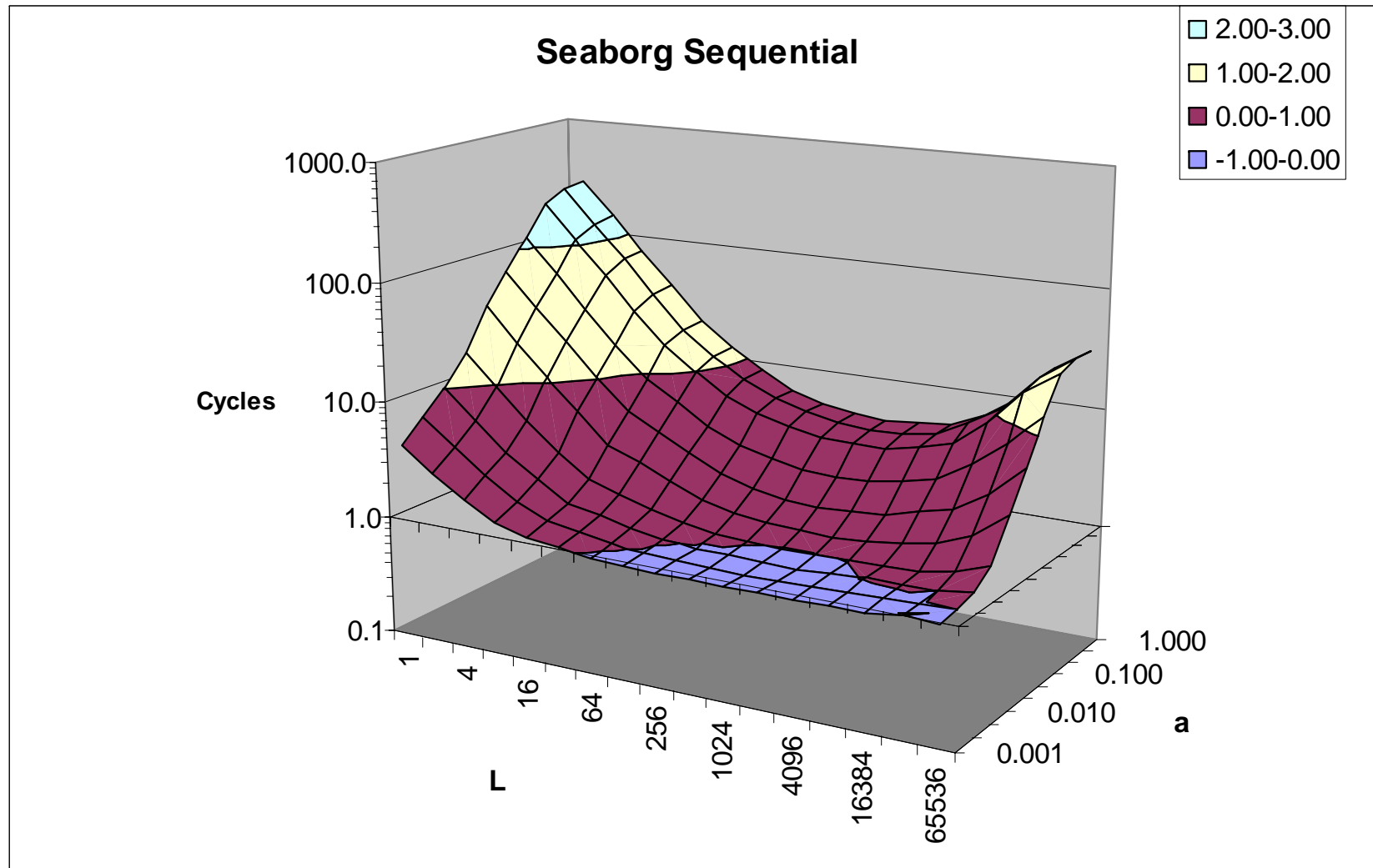
1/L

1=Low

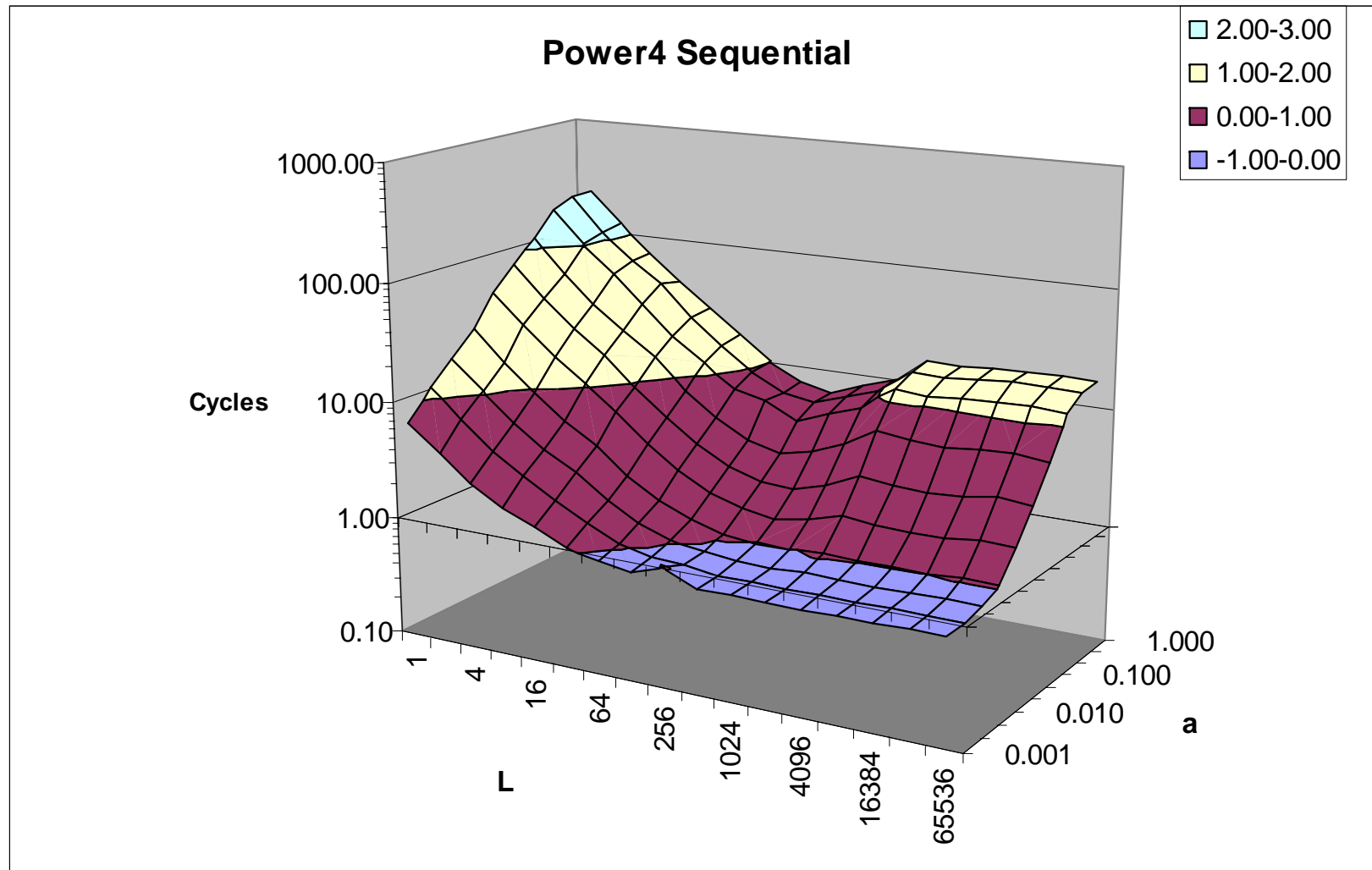
Spatial Locality



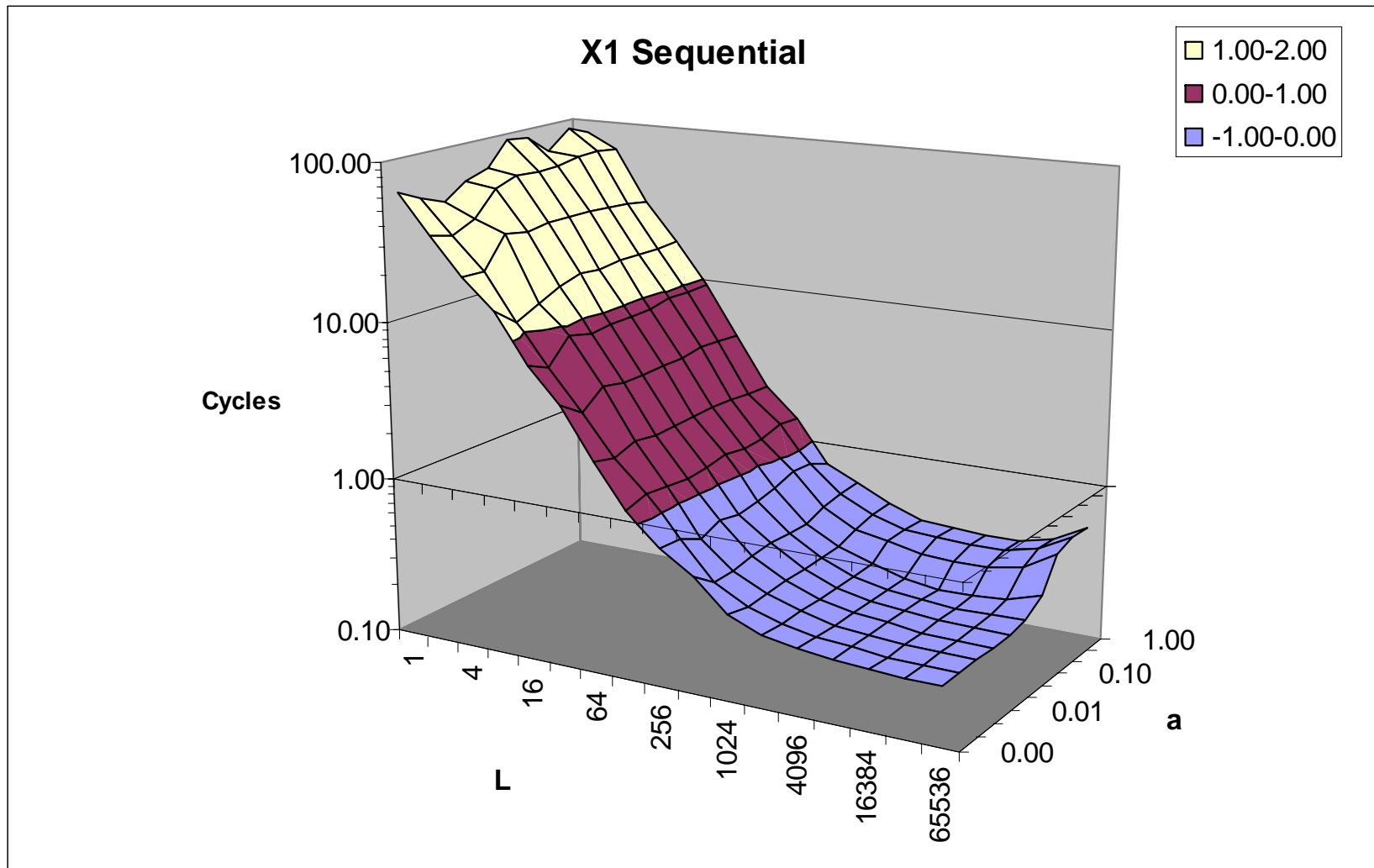
Apex-Map Sequential



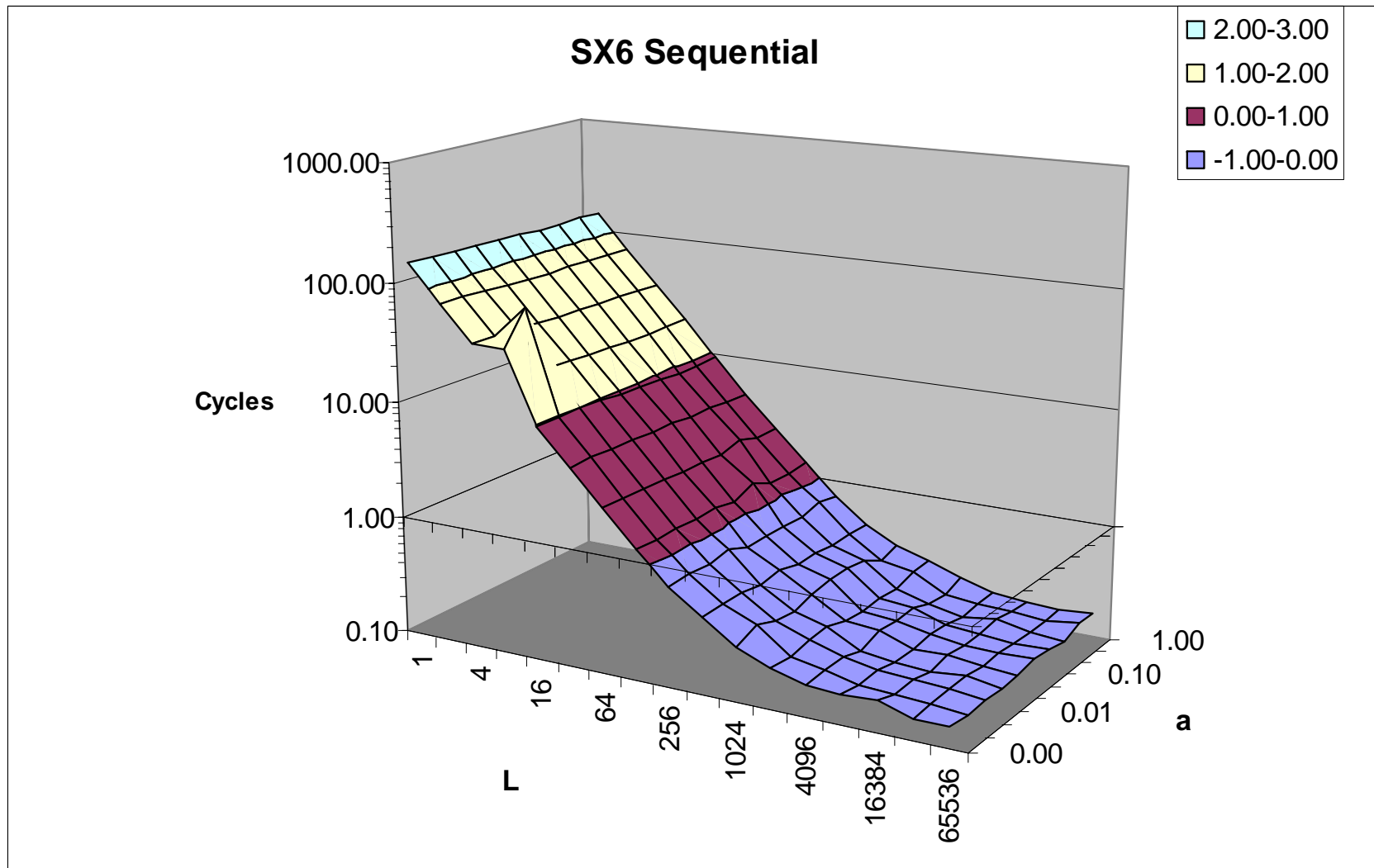
Apex-Map Sequential



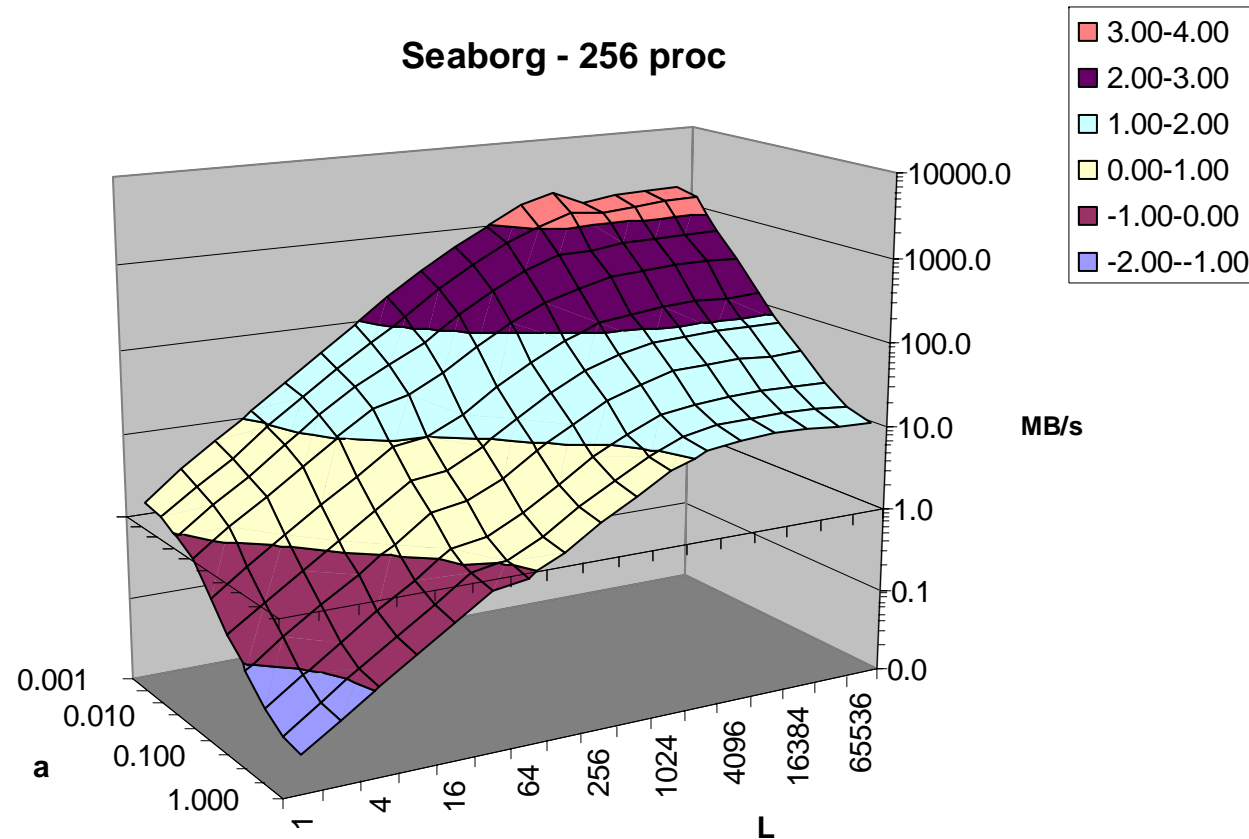
Apex-Map Sequential



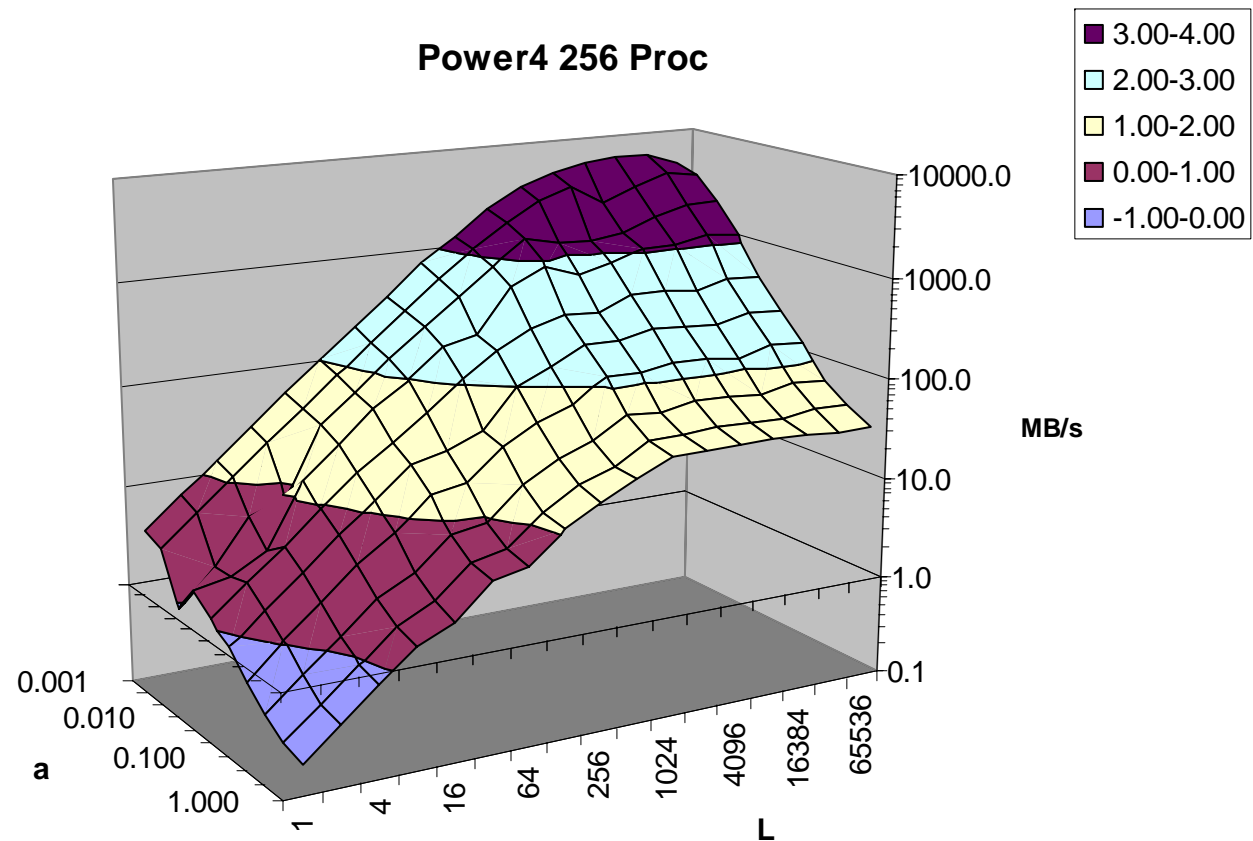
Apex-Map Sequential



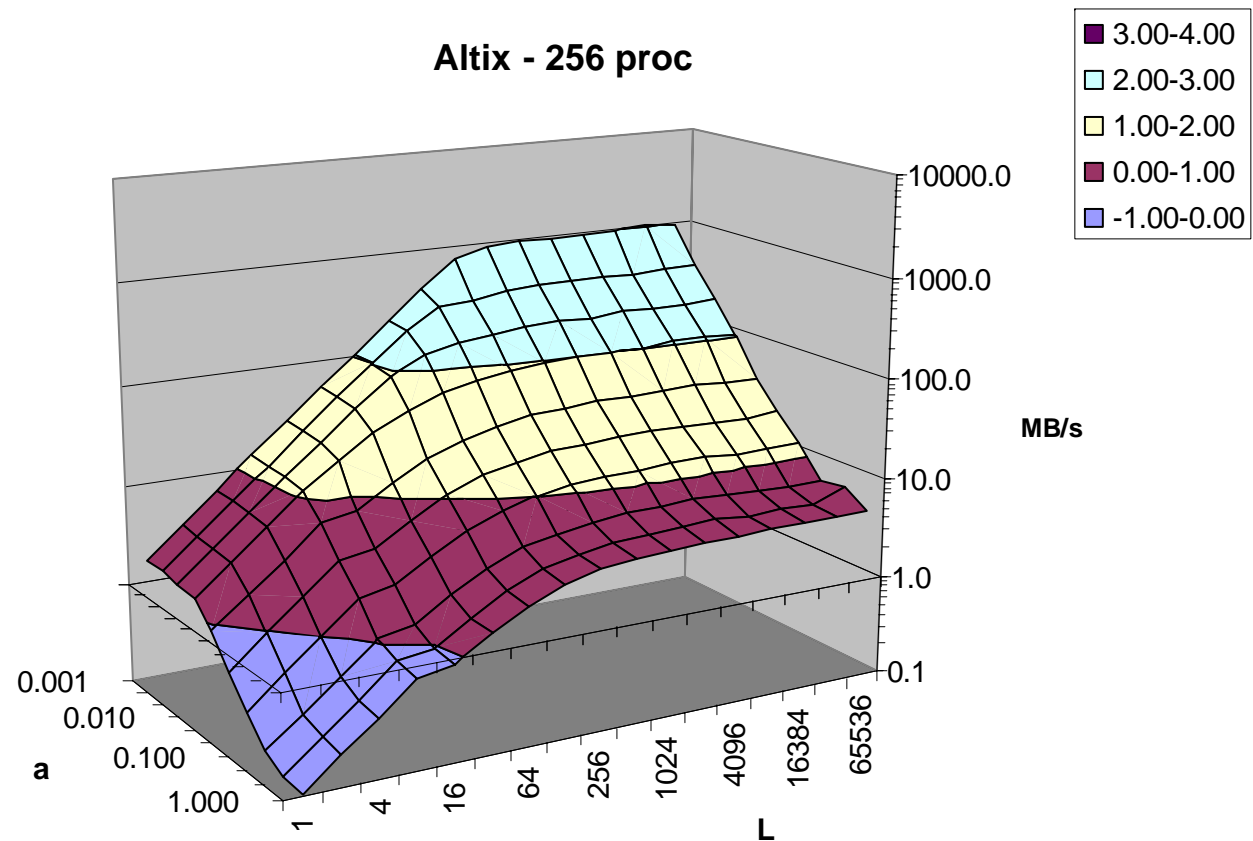
Parallel APEX-Map



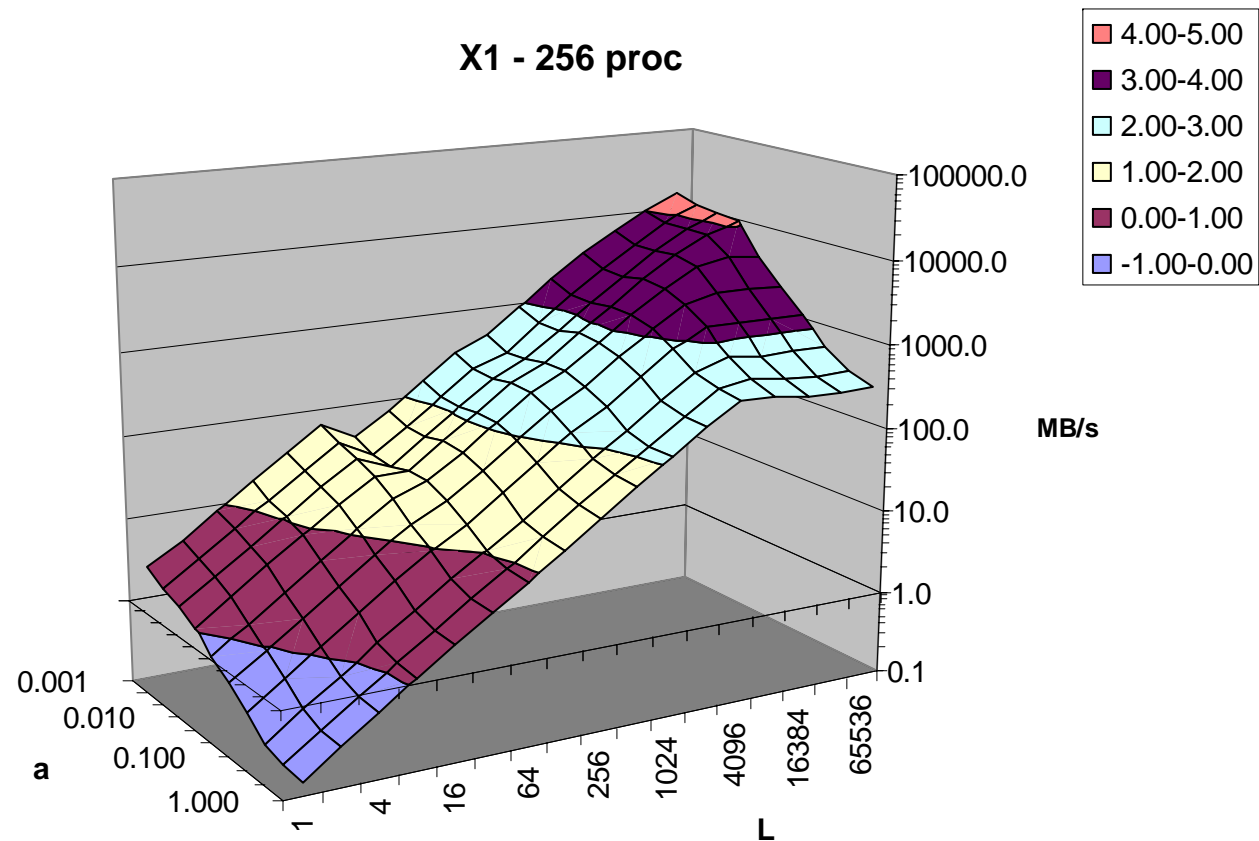
Parallel APEX-Map



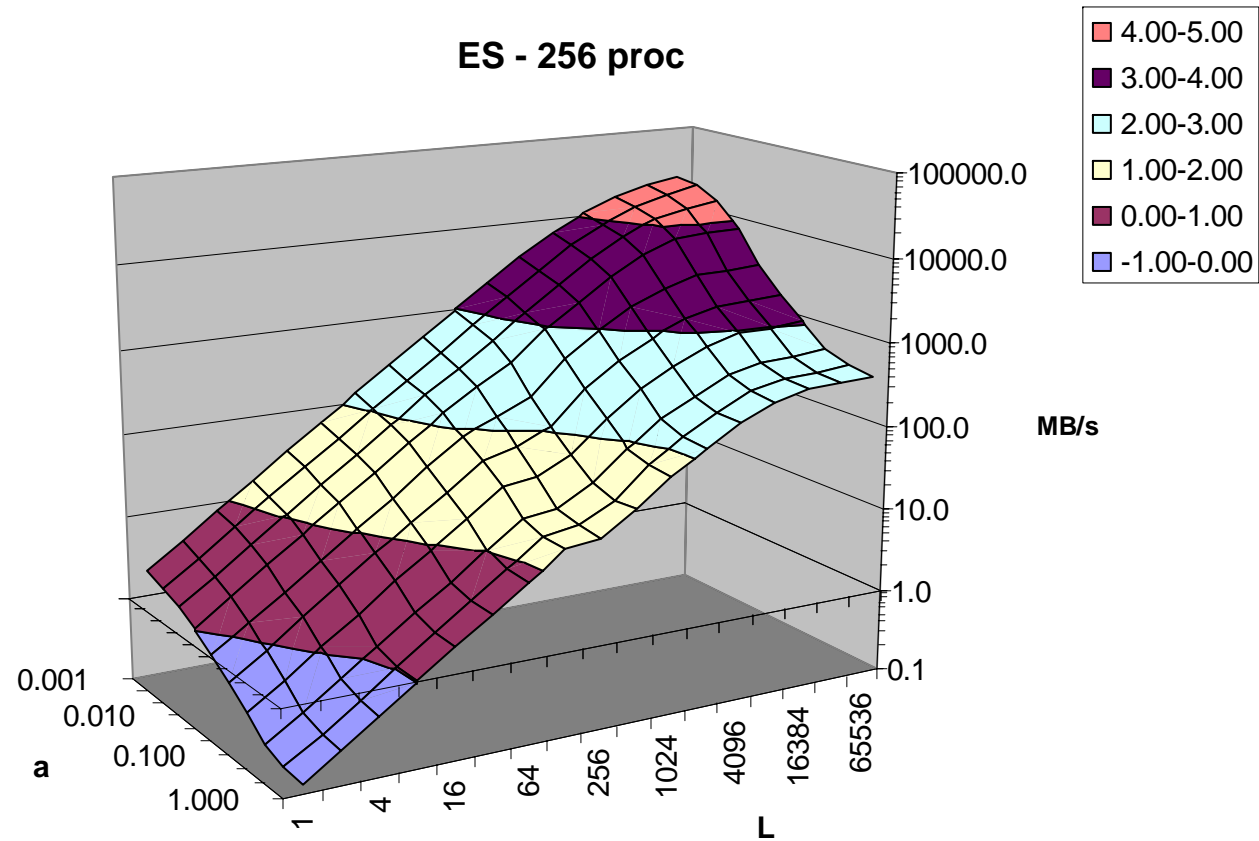
Parallel APEX-Map



Parallel APEX-Map



Parallel APEX-Map



Summary

- Three sets of tools (applications benchmarks, performance monitoring, quantitative architecture characterization) have been shown to provide critical insight into applications performance
- Need better quantitative data and measurements (like the ones discussed here) to help applications to scale to the next generation of platforms

