High-performance Geometric Multigrid (HPGMG) and quantification of performance versatility

This talk:
https://jedbrown.org/files/20160217-CISLVersatility.pdf

Jed Brown jed@jedbrown.org (CU Boulder)

CISL Seminar, NCAR, 2016-02-17

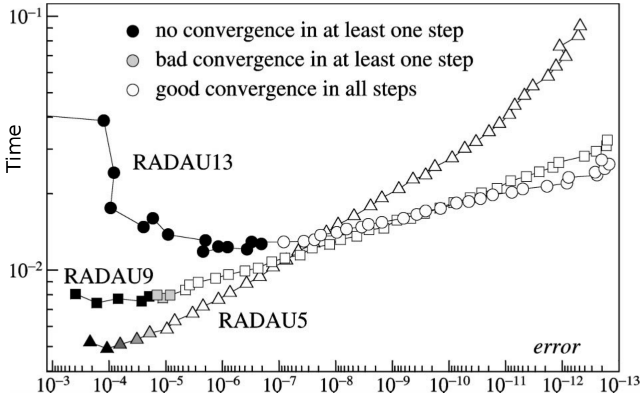
What is performance?

Dimensions

- Model complexity
- Accuracy
- Time
  - per problem instance
  - for the first instance
  - compute time versus human time
- Cost
  - incremental cost
  - subsidized?

- Terms relevant to scientist/engineer
- Compute meaningful quantities – needed to make a decision or obtain a result of scientific value—not one iteration/time step
- No flop/s, number of elements/time steps

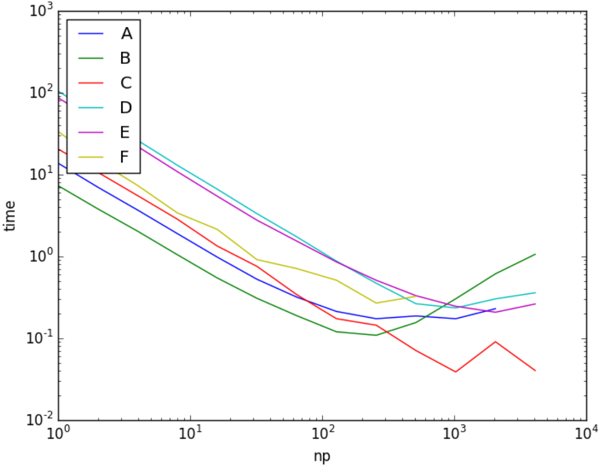
Work-precision diagram: *de rigueur* in ODE community



[Hairer and Wanner (1999)]

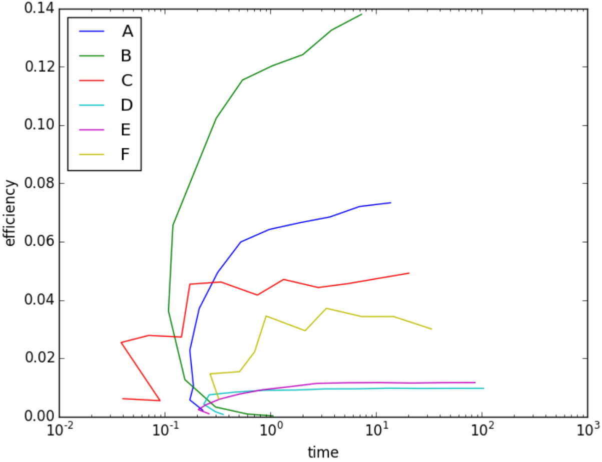
- Tests discretization, adaptivity, algebraic solvers, implementation
- No reference to number of time steps, flop/s, etc.
- Useful performance results inform *decisions* about *tradeoffs*.

Strong Scaling: efficiency-time tradeoff



- Good: shows absolute time
- Bad: log-log plot makes it difficult to discern efficiency
  - Stunt 3: `http://blogs.fau.de/hager/archives/5835`
- Bad: plot depends on problem size

Strong Scaling: efficiency-time tradeoff



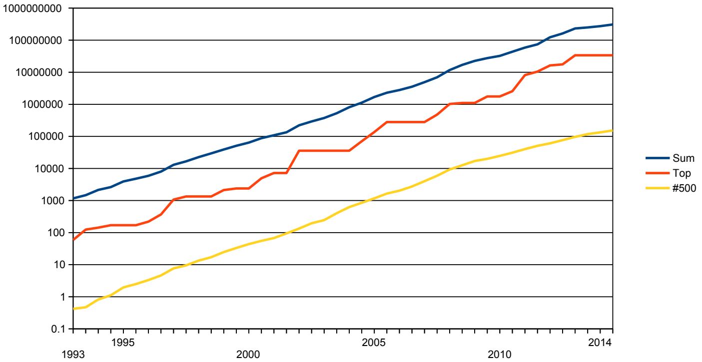
- Good: absolute time, absolute efficiency (like DOF/s/cost)
- Good: independent of problem size for perfect weak scaling
- Bad: hard to see machine size (but less important)

Exascale Science & Engineering Demands

- Model fidelity: resolution, multi-scale, coupling
  - Transient simulation is not weak scaling: $\Delta t \sim \Delta x$
- Analysis using a sequence of forward simulations
  - Inversion, data assimilation, optimization
  - Quantify uncertainty, risk-aware decisions
- Increasing relevance $\implies$ external requirements on time
  - Policy: 5 SYPD to inform IPCC
  - Weather, manufacturing, field studies, disaster response
- "weak scaling" [...] will increasingly give way to "strong scaling"

  [The International Exascale Software Project Roadmap, 2011]
- ACME @ 25 km scaling saturates at $<10\%$ of Titan (CPU) or Mira
  - Cannot decrease Δx: SYPD would be too slow to calibrate
  - "results" would be meaningless for 50-100y predictions, a "stunt run"
- **ACME v1 goal of 5 SYPD is pure strong scaling.**
  - Likely faster on Edison (2013) than any DOE machine –2020
  - Many non-climate applications in same position.

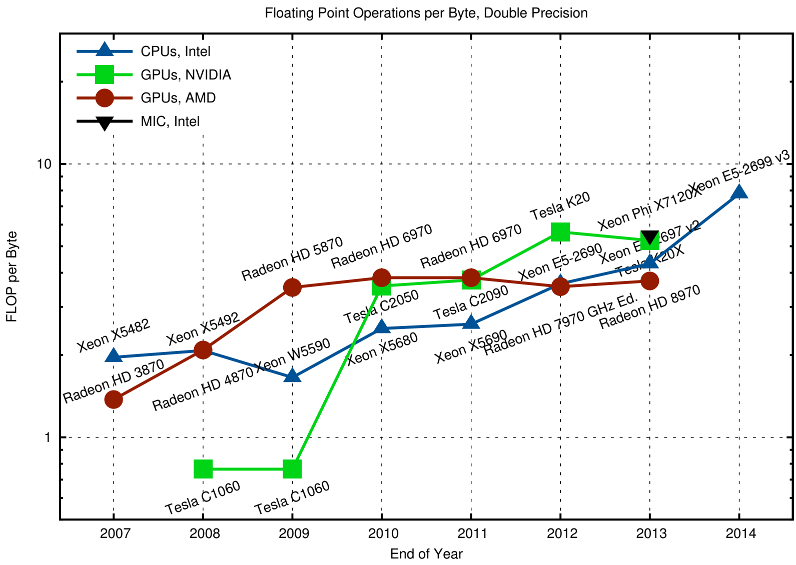
HPL and the Top500 list



- High Performance LINPACK
- Solve $n \times n$ dense linear system: $\mathcal{O}(N^{3/2})$ flops on $N = n^2$ data
- Top500 list created in 1993 by Hans Meuer, Jack Dongarra, Erich Strohmeier, Horst Simon

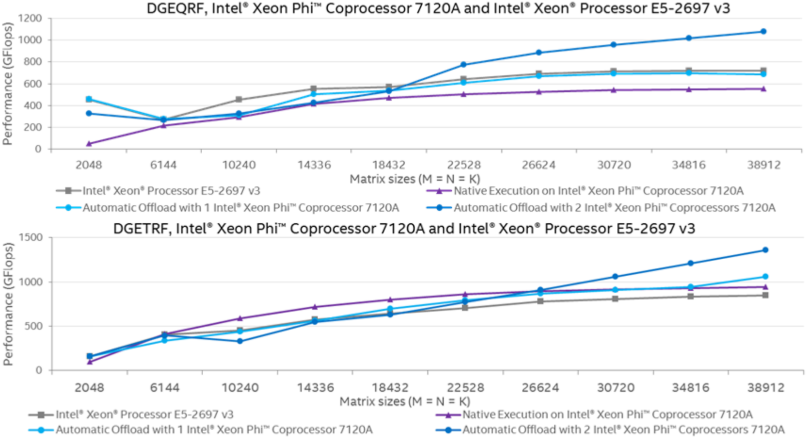
Role of HPL

- The major centers have their own benchmark suites (e.g., CORAL)
- Nobody (vendors or centers) will say they built an HPL machine
- HPL ranking and peak flop/s are still used for press releases
- Machines need to be justified to politicians holding the money
  - Politicians are vulnerable to propaganda and claims of inefficient spending
- It is naive to believe HPL has no influence on procurement or on scientists' expectations


Floating Point Operations per Byte, Double Precision
CPUs, Intel
GPUs, NVIDIA
GPUs, AMD
MIC, Intel
FLOP per Byte
10
1
2007
2008
2009
2010
2011
2012
2013
2014
End of Year
Xeon X5482
Xeon X5492
Xeon W5590
Xeon X5680
Xeon X5690
Xeon E5-2690
Xeon E5-2697 v2
Xeon E5-2699 v3
Xeon Phi X7120X
Radeon HD 3870
Radeon HD 4870
Radeon HD 5870
Radeon HD 6970
Radeon HD 6970
Radeon HD 7970 GHz Ed.
Radeon HD 8970
Tesla C1060
Tesla C1060
Tesla C2050
Tesla C2090
Tesla K20
Tesla K20X

[c/o Karl Rupp]

Arithmetic intensity is not enough



- QR and LU factorization have same complexity.
- Stable QR factorization involves more synchronization.
- Synchronization is much more expensive on Xeon Phi.

How much parallelism out of how much cache?

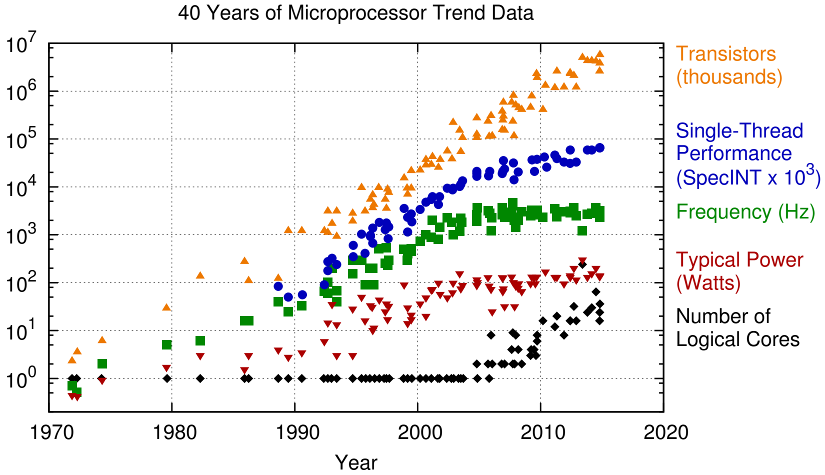
Processor	v width	threads	F/inst	latency	L1D	L1D/#par
Nehalem	2	1	2	5	32 KiB	1638 B
Sandy Bridge	4	2	2	5	32 KiB	819 B
Haswell	4	2	4	5	32 KiB	410 B
BG/P	2	1	2	6	32 KiB	1365 B
BG/Q	4	4	2	6	32 KiB	682 B
KNC	8	4	4	5	32 KiB	205 B
Tesla K20	32	*	2	10	64 KiB	102 B

- Most “fast” algorithms do about $O(N)$ flops on N data
- xGEMM does $O(N^{3/2})$ flops on N data
- Exploitable parallelism limited by cache and register load/store
- L2/L3 performance highly variable between architectures

Vectorization versus memory locality

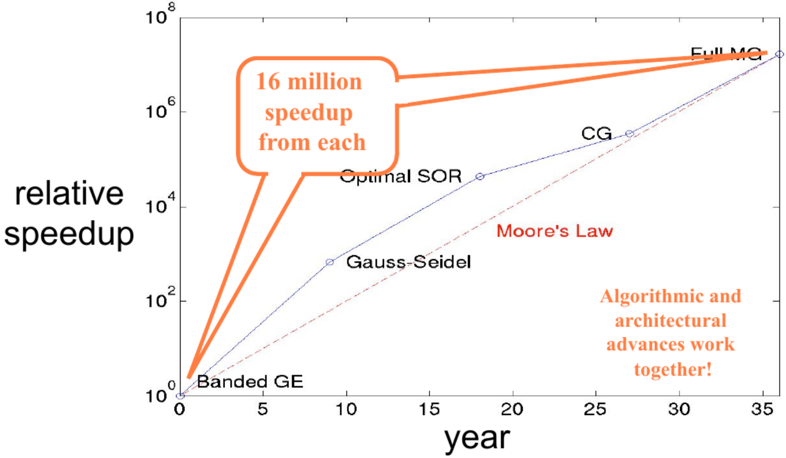
- Each vector lane and pipelined instruction need their own operands
- Can we extract parallelism from smaller working set?
  - Sometimes, but more cross-lane and pipeline dependencies
  - More complicated/creative code, harder for compiler
- Good implementations strike a brittle balance (e.g., Knepley, Rupp, Terrel; HPGMG-FE)
- Applications change discretization order, number of fields, etc.
  - CFD: 5-15 fields
  - Tracers in atmospheric physics: 100 species
  - Adaptive chemistry for combustion: 10-10000 species
  - Crystal growth for mesoscale materials: 10-10000 fields
- AoS or SoA?
  - Choices not robust to struct size
  - AoS good for prefetch and cache reuse
  - Can pack into SoA when necessary

SPECint is increasing despite stagnant clock



- Karl Rupp's update to figure by Horowitz et al.

Algorithms keep pace with hardware (sometimes)



[c/o David Keyes]

- Opportunities now: uncertainty quantification, design
- Incentive to find optimal algorithms for more applications

What does “representative” mean?

- Diverse applications
  - Explicit PDE solvers (seismic wave propagation, turbulence)
  - Implicit PDE solvers and multigrid methods (geodynamics, structural mechanics, steady-state RANS)
  - Irregular graph algorithms (network analysis, genomics, game trees)
  - Dense linear algebra and tensors (quantum chemistry)
  - Fast methods for N-body problems (molecular dynamics, cosmology)
  - Cross-cutting: data assimilation, uncertainty quantification
- Diverse external requirements
  - Real-time, policy, manufacturing
  - Privacy
  - In-situ processing of experimental data
  - Mobile/energy limitations

Necessary and sufficient

Goodhart's Law

When a measure becomes a target, it ceases to be a good measure.

- Features stressed by benchmark **necessary** for some apps
- Performance on benchmark **sufficient** for most apps

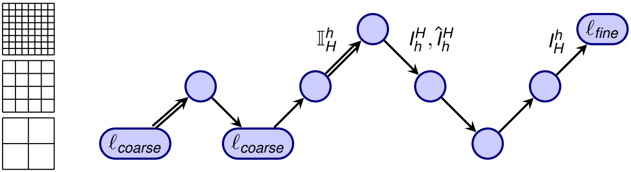
HPGMG: a new benchmarking proposal

- `https://hpgmg.org`, hpgmg-forum@hpgmg.org mailing list
- Mark Adams, Sam Williams (finite-volume), Jed (finite-element), John Shalf, Brian Van Straalen, Erich Strohmeier, Rich Vuduc
- Gathering momentum, SC14 BoF
- Implementations

  Finite Volume: memory bandwidth intensive, simple data dependencies, 2nd and 4th order

  Finite Element: compute- and cache-intensive, vectorizes, overlapping writes
- Full multigrid, well-defined, scale-free problem
- Matrix-free operators, Chebyshev smoothers

Full Multigrid (FMG): Prototypical Fast Algorithm

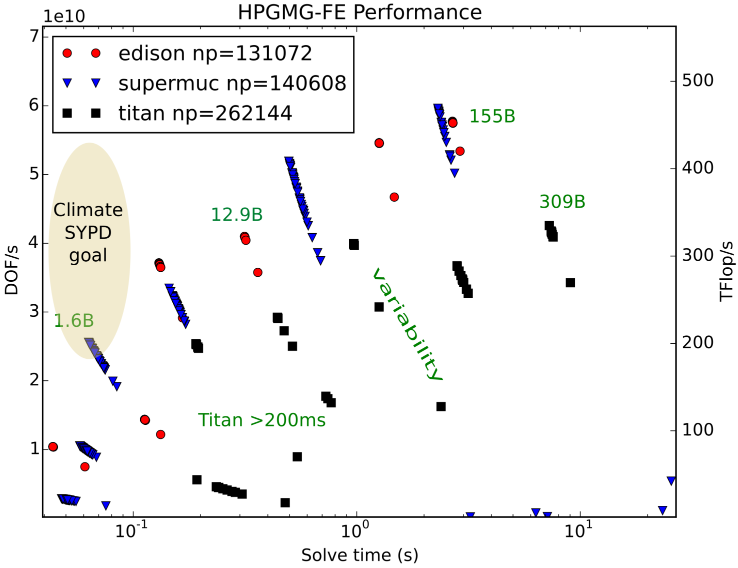


- start with coarse grid
- truncation error within one cycle
- about five work units for many problems
- no “fat” left to trim – robust to gaming
- distributed memory – restrict active process set using Z-order
  - $\mathcal{O}(\log^2 N)$ parallel complexity stresses network
- scale-free specification
  - no mathematical reward for decomposition granularity
  - don’t have to adjudicate “subdomain”

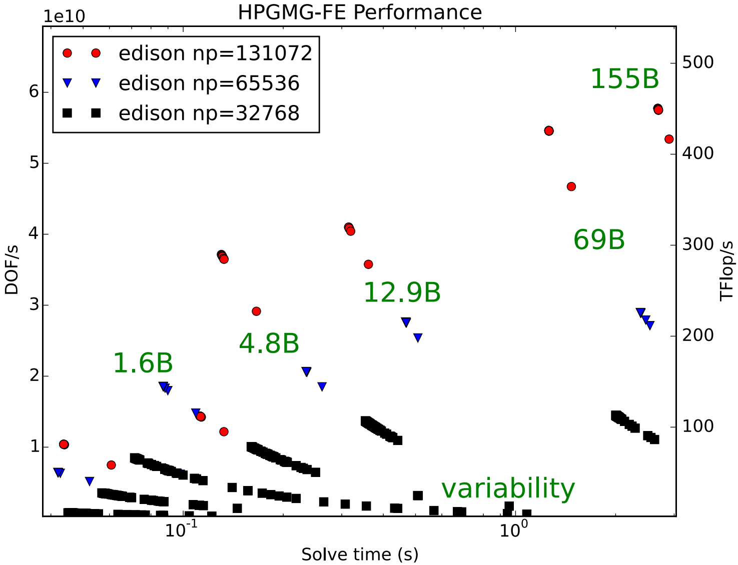
Multigrid design decisions

- Q_2 finite elements
  - Partition of work not partition of data – sharing/overlapping writes
  - Q_2 is a middle-ground between lowest order and high order
  - Matrix-free pays off, tensor-product element evaluation
- Linear elliptic equation with manufactured solution
- Mapped coordinates
  - More memory streams, increase working set, longer critical path
- No reductions
  - Coarse grid is strictly more difficult than reduction
  - Not needed because FMG is a direct method
- Chebyshev/Jacobi smoothers, $V(3,1)$ cycle
  - Multiplicative smoothers hard to verify in parallel
  - Avoid intermediate scales (like Block Jacobi/Gauss-Seidel)
- Full Approximation Scheme

HPGMG-FE on Edison, SuperMUC, Titan



HPGMG-FE on Edison (Aries, E5-2695v2)



HPGMG-FV, 2015-11, 2nd order

HPGMG Rank	System	Site	HPGMG DOF/s	Fraction of System	Parallelization MPI	OMP	GPU	DOF per Process	Top500 Rank
1	K	RIKEN	2.83E+12	100%	82944	8		72M	4
2	Titan (CPU+GPU)	Oak Ridge	9.16e+11	100%	16384	4	1	32M	2
	(CPU-only)	Oak Ridge	2.53E+11	100%	32768	8		16M	
3	Mira	Argonne	7.21E+11	100%	49152	64		16M	5
4	Edison	NERSC	3.85E+11	100%	131072	1		4M	40
5	Stampede (CPU-only)	TACC	1.49E+11	64%	8192	8		2M	10
6	Hopper	NERSC	1.21E+11	86%	21952	6		2M	72
7	Piz Daint (CPU-only)	CSCS	1.02E+11	78%	4096	8		18M	7
8	SuperMUC	LRZ	7.13E+10	15%	2744	8		16M	23
9	BiFrost	NSC	4.67E+10	100%	1260	16		176M	-
10	Stampede (MIC-only)	TACC	2.16E+10	8%	512	180		16M	7
11	Peregrine (IVB-only)	NREL	1.08E+10	18%	512	12		2M	-
12	Carver	NERSC	1.35E+09	5%	125	4		2M	-
13	Babbage (MIC-only)	NERSC	8.24E+08	30%	27	180		16M	-

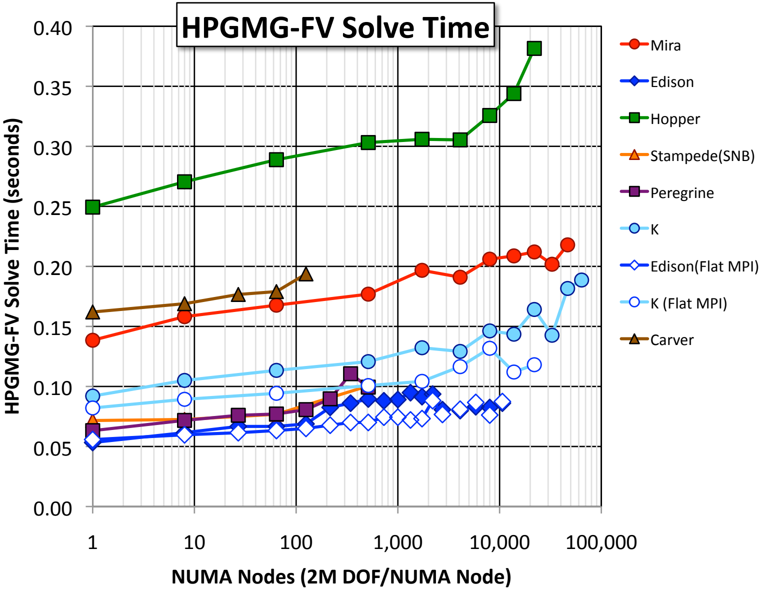
HPGMG-FV, 2015-11, 4th order

	System		HPGMG DOF/s			Parallelization			DOF per	Top500
Rank	Name	Site	h	2h	4h	MPI	OMP	ACC	Process	Rank
1	Mira	ALCF	5.00e11	3.13e11	1.07e11	49152	64		36M	5
			3.95e11	2.86e11	1.07e11	49152	64		36M	
2	Edison	NERSC	2.96e11	2.46e11	1.27e11	10648	12		128M	34
3	Titan (CPU-only)	OLCF	1.61e11	8.25e10	2.37e10	36864	8		48M	2
4	Hopper	NERSC	7.26e10	5.45e10	2.74e10	21952	6		16M	62
5	SuperMUC (22%)	LRZ	7.25e10	5.25e10	2.80e10	4096	8		54M	20
6	Hazel Hen (7%)	HLRS	1.82e10	8.73e09	2.02e09	1024	12		16M	-
7	SX-ACE (vector)	HLRS	3.24e09	1.77e09	7.51e08	256	1		32M	-
8	Babbage (MIC-only)	NERSC	7.62e08	3.16e08	9.93e07	256	45		8M	-

Messaging from threaded code

- Off-node messages need to be packed and unpacked
- Many MPI+threads apps pack in serial – bottleneck
- Extra software synchronization required to pack in parallel
  - Formally $O(\log T)$ critical path, T threads/NIC context
  - Typical OpenMP uses barrier – oversynchronizes
- `MPI_THREAD_MULTIPLE` – atomics and $O(T)$ critical path
- Choose serial or parallel packing based on T and message sizes?
- Hardware NIC context/core now, maybe not in future
- What is lowest overhead approach to message coalescing?

HPGMG-FV: flat MPI vs MPI+OpenMP (Aug 2014)



- c/o Sam Williams

CAM-SE dynamics numbers

- 25 km resolution, 18 simulated seconds/RK stage
- Current performance at strong scaling limit

Edison	3 SYPD
Titan	2 SYPD
Mira	0.9 SYPD

- Performance requirement: 5 SYPD (about 2000x faster than real time)
  - 10 ms budget per dynamics stage
  - Increasing spatial resolution decreases this budget
- ACME strong scaling saturates while too small for the Capability Queue on DOE LCFs
- Null hypothesis: Edison will run ACME faster than any DOE machine through 2020
  - Difficult to get large allocations

Tim Palmer's call for 1km (Nature, 2014)

> Running a climate simulator with 1-kilometre cells over a timescale of a century will require 'exascale' computers capable of handling more than 10^{18} calculations per second. Such computers should become available within the present decade, but may not become affordable for individual institutes for another decade or more.

- Would require 10^4 more total work than ACME target resolution
- 5 SYPD at 1km is like 75 SYPD at 15km, assuming infinite resource and perfect weak scaling
- Two choices:
  1. compromise simulation speed—this would come at a high price, impacting calibration, data assimilation, and analysis; or
  2. ground-up redesign of algorithms and hardware to cut latency by a factor of 20 from that of present hardware
- DE Shaw's Anton is an example of Option 2
- Models need to be constantly developed and calibrated
  - custom hardware stifles algorithm/model innovation
- Exascale roadmaps don't make a dent in 20x latency problem

Outlook

- Application scaling mode must be scientifically relevant
- Algorithmic barriers exist
  - Throughput architectures are not just “hard to program”
- Vectorization versus memory locality
- Over-decomposition adds overhead and lengthens critical path
- Versatile architectures are needed for model coupling and advanced analysis
  - How to include dynamic range in ranking metric?
  - Why is NERSC installing DRAM in Cori?
- Abstractions must be durable to changing scientific needs
- “Energy efficiency” is not if algorithms give up nontrivial constants
- What is the cost of performance variability?
  - Measure best performance, average, median, 10th percentile?
- HPGMG `https://hpgmg.org`
- The real world is messy!