CSCI653: High Performance Computing & Simulations

Aiichiro Nakano

Collaboratory for Advanced Computing & Simulations
Department of Computer Science
Department of Physics & Astronomy
Department of Chemical Engineering & Materials Science
Department of Biological Sciences
University of Southern California

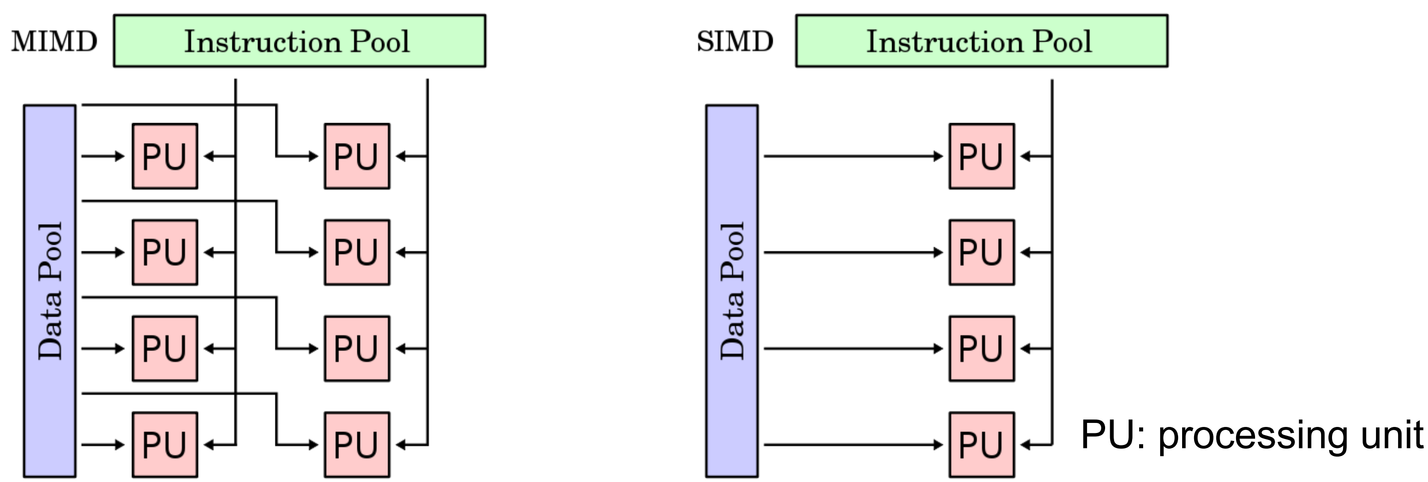
Email: anakano@usc.edu

Do "Your" Science Using High Performance Computing

CSCI653 at a Glance: Applications

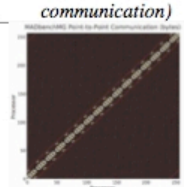
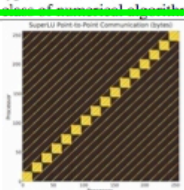
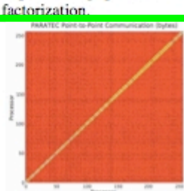
- High performance computing (HPC) with archetypal real-world applications
  > Molecular dynamics (MD): interaction
  > Multiple-instruction multiple-data (MIMD)

  > Quantum dynamics (QD): data parallelism
  > Single-instruction multiple-data (SIMD)
- Hybrid multiscale/multiphysics applications
- Deterministic *vs*. stochastic (to solve intractable) applications
- Data + learning + visualization

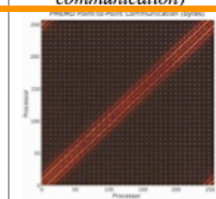
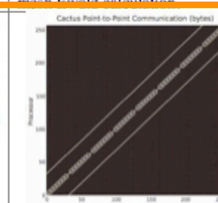


The Landscape of Parallel Computing Research: A View from Berkeley

7 dwarfs (a dwarf is an algorithmic method that captures a pattern of computation and communication) + 6 combinatorial

Dwarf	Description	Communication Pattern (Figure axes show processors 1 to 256, with black meaning no communication)	NAS Benchmark / Example HW
1. Dense Linear Algebra (e.g., BLAS [Blackford et al 2002], ScaLAPACK [Blackford et al 1996], or MATLAB [MathWorks 2006])	Data are dense matrices or vectors. (BLAS Level 1 = vector-vector; Level 2 = matrix-vector; and Level 3 = matrix-matrix.) Generally, such applications use unit-stride memory accesses to read data from rows, and strided accesses to read data from columns.	 The communication pattern of MadBench, which makes heavy use of ScaLAPACK for parallel dense linear algebra, is typical of a much broader class of numerical algorithms.	Block Triadiagonal Matrix, Lower Upper Symmetric Gauss-Seidel / Vector computers, Array computers
2. Sparse Linear Algebra (e.g., SpMV, OSKI [OSKI 2006], or SuperLU [Demmel et al 1999])	Data sets include many zero values. Data is usually stored in compressed matrices to reduce the storage and bandwidth requirements to access all of the nonzero values. One example is block compressed sparse row (BCSR). Because of the compressed formats, data is generally accessed with indexed loads and stores.	 SuperLU (communication pattern pictured above) uses the BCSR method for implementing sparse LU factorization.	Conjugate Gradient / Vector computers with gather/scatter
3. Spectral Methods (e.g., FFT [Cooley and Tukey 1965])	Data are in the frequency domain, as opposed to time or spatial domains. Typically, spectral methods use multiple butterfly stages, which combine multiply-add operations and a specific pattern of data permutation, with all-to-all communication for some stages and strictly local for others.	 PARATEC: The 3D FFT requires an all-to-all communication to implement a 3D transpose, which requires communication between every link. The diagonal stripe describes BLAS-3 dominated linear-algebra step required for orthogonalization.	Fourier Transform / DSPs, Zalink PDSP [Zarlink 2006]

In performance optimization

Dwarf	Description	Communication Pattern (Figure axes show processors 1 to 256, with black meaning no communication)	NAS Benchmark / Example HW
4. N-Body Methods (e.g., Barnes-Hut [Barnes and Hut 1986], Fast Multipole Method [Greengard and Rokhlin 1987])	Depends on interactions between many discrete points. Variations include particle-particle methods, where every point depends on all others, leading to an $O(N^2)$ calculation, and hierarchical particle methods, which combine forces or potentials from multiple points to reduce the computational complexity to O(N log N) or O(N).	 PMEMD's communication pattern is that of a particle mesh Ewald calculation.	(no benchmark) / GRAPE [Tokyo 2006], MD-GRAPE [IBM 2006]
5. Structured Grids (e.g., Cactus [Goodale et al 2003] or Lattice-Boltzmann Magneto-hydrodynamics [LBMHD 2005])	Represented by a regular grid; points on grid are conceptually updated together. It has high spatial locality. Updates may be in place or between 2 versions of the grid. The grid may be subdivided into finer grids in areas of interest ("Adaptive Mesh Refinement"); and the transition between granularities may happen dynamically.	 Communication pattern for Cactus, a PDE solver using 7-point stencil on 3D block-structured grids.	Multi-Grid, Scalar Penta-diagonal / QCDOC [Edinburg 2006], BlueGeneL
6. Unstructured Grids (e.g., ABAQUS [ABAQUS 2006] or FIDAP [FLUENT 2006])	An irregular grid where data locations are selected, usually by underlying characteristics of the application. Data point location and connectivity of neighboring points must be explicit. The points on the grid are conceptually updated together. Updates typically involve multiple levels of memory reference indirection, as an update to any point requires first determining a list of neighboring points, and then loading values from those neighboring points.		Unstructured Adaptive / Vector computers with gather/scatter, Tera Multi Threaded Architecture [Berry et al 2006]
7. Monte Carlo (e.g., Quantum Monte Carlo [Aspuru-Guzik et al 2005])	Calculations depend on statistical results of repeated random trials. Considered embarrassingly parallel.	Communication is typically not dominant in Monte Carlo methods.	Embarrassingly Parallel / NSF Teragrid

MD

QD

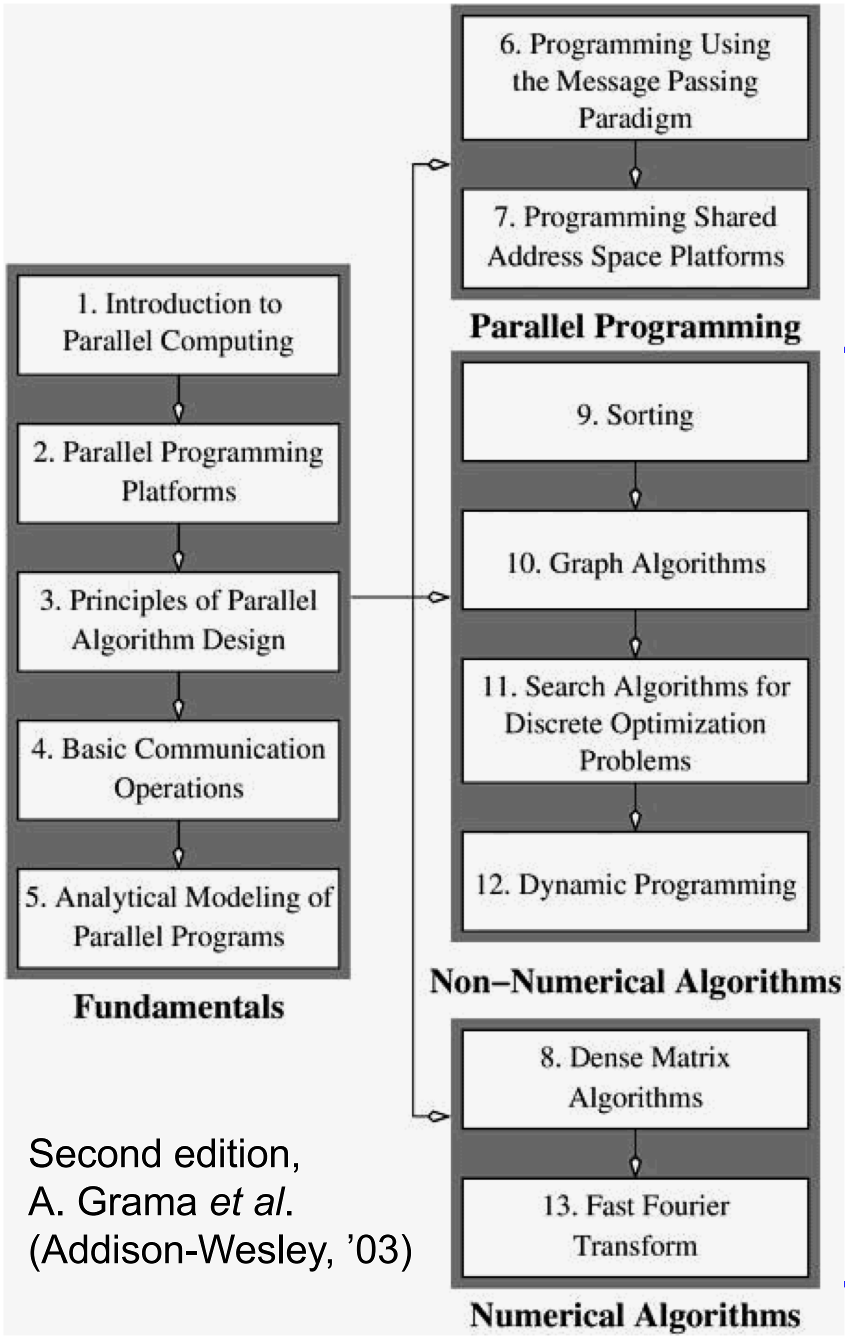
Monte Carlo

https://www2.eecs.berkeley.edu/Pubs/TechRpts/2006/EECS-2006-183.html

CSCI653: Algorithms & Tools

- ***Just one thing*: Divide-conquer-“recombine” (DCR) algorithm; it’s data locality!**
- ***Parallel computing = decomposition (who does what)*: Scalability analysis; performance optimization**
- ***Programming languages*: MPI (distributed memory) + OpenMP (shared memory) + CUDA|OpenMP4.5 (heterogeneous accelerator)**

Introduction to Parallel Computing



MPI

OpenMP

SIMD, MIMD *etc.*

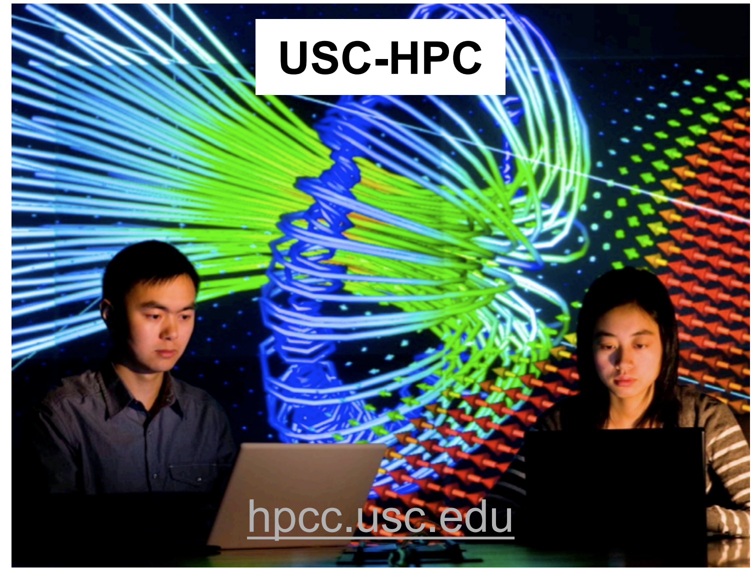
Decomposition

Scalability analysis

HWs

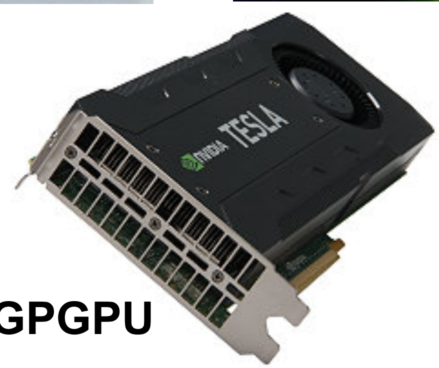
Second edition,
A. Grama *et al*.
(Addison-Wesley, ’03)

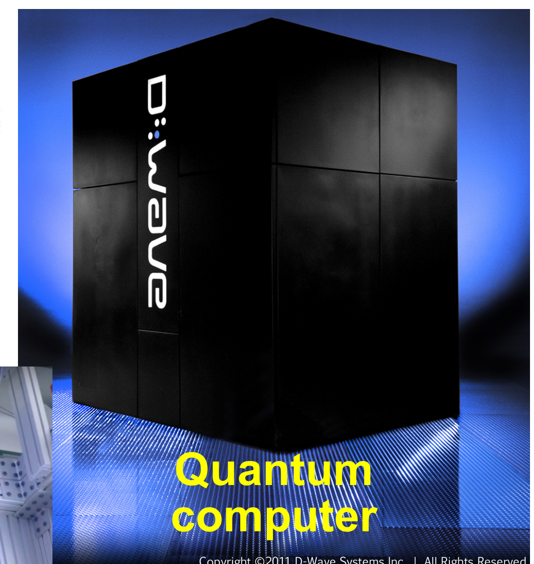
High Performance Computing (HPC)



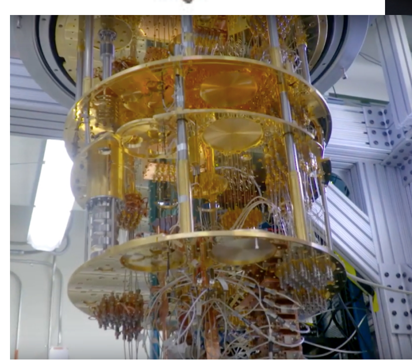
- *USC HPC (Center for High Performance Computing & Communication)*: **13,440 CPU-core GPU-accelerated 0.62 petaflop/s cluster**
- *USC ISI (Information Sciences Institute)*: **1,098-qubit D-Wave quantum computer**

GPGPU





QPU



petaflop/s = 10^{15} mathematical operations per second

Current & Future HPC Platforms

- **Won two DOE supercomputing awards to develop & deploy metascalable (“design once, scale on future platforms”) simulation algorithms (2017-2020)**

U.S. DEPARTMENT OF ENERGY
INCITE
LEADERSHIP COMPUTING

- **Quantum simulations on full 800K cores**

Innovative & Novel Computational Impact on Theory & Experiment

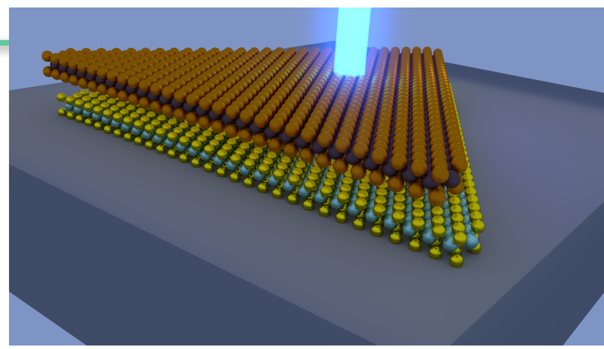
Title: “Petascale Simulations for Layered Materials Genome”

Principal Investigator: Aiichiro Nakano, University of Southern California
Co-Investigator: Priya Vashishta, University of Southern California

https://www.anl.gov/article/whos-using-the-user-facilities

786,432-core IBM Blue Gene/Q

AURORA | EARLY SCIENCE PROGRAM



Early Science Projects for Aurora Supercomputer Announced

Metascalable layered materials genome

Investigator: Aiichiro Nakano, University of Southern California

Nation’s first exaflop/s computer, Intel A21 (2021)

exaflop/s = 10^{18} mathematical operations per second

- **One of the 10 initial simulation users of the next-generation DOE supercomputer**

https://www.alcf.anl.gov/projects/aurora-esp

CACS@A21 in the Global Exascale Race

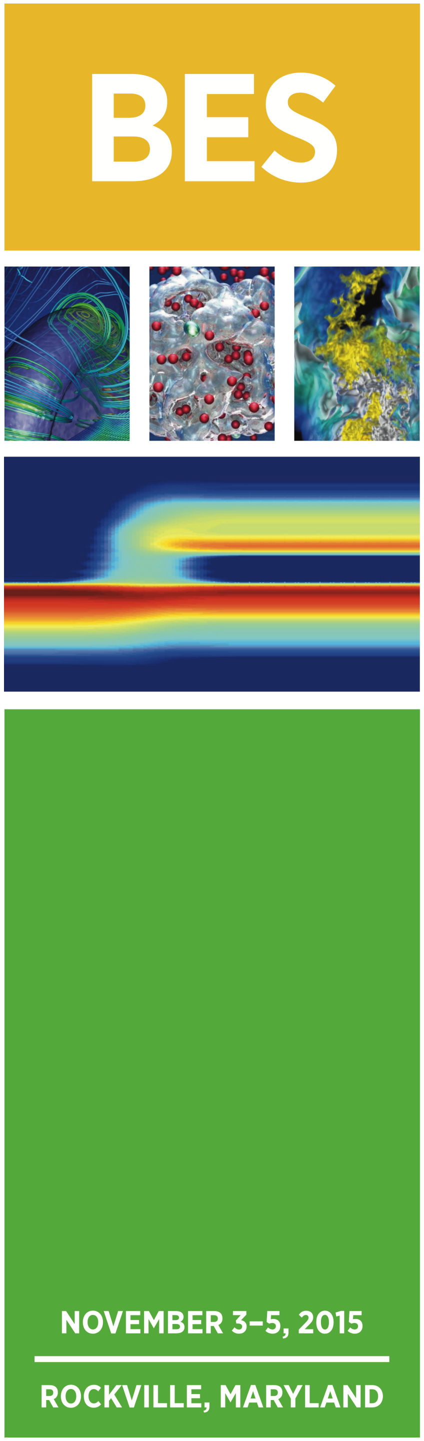
SUPERCOMPUTING R. F. Service, *Science* **359**, 617 ('18)

Design for U.S. exascale computer takes shape

Competition with China accelerates plans for next great leap in supercomputing power

By **Robert F. Service**

In 1957, the launch of the Sputnik satellite vaulted the Soviet Union to the lead in the space race and galvanized the United States. U.S. supercomputer researchers are today facing their own Lemont, Illinois. That's 2 years earlier than planned. "It's a pretty exciting time," says Aiichiro Nakano, a physicist at the University of Southern California in Los Angeles who uses supercomputers to model materials made by layering stacks of atomic sheets like graphene. pace reflects a change of strategy by DOE officials last fall. Initially, the agency set up a "two lanes" approach to overcoming the challenges of an exascale machine, in particular a potentially ravenous appetite for electricity that could require the output of a small nuclear plant.

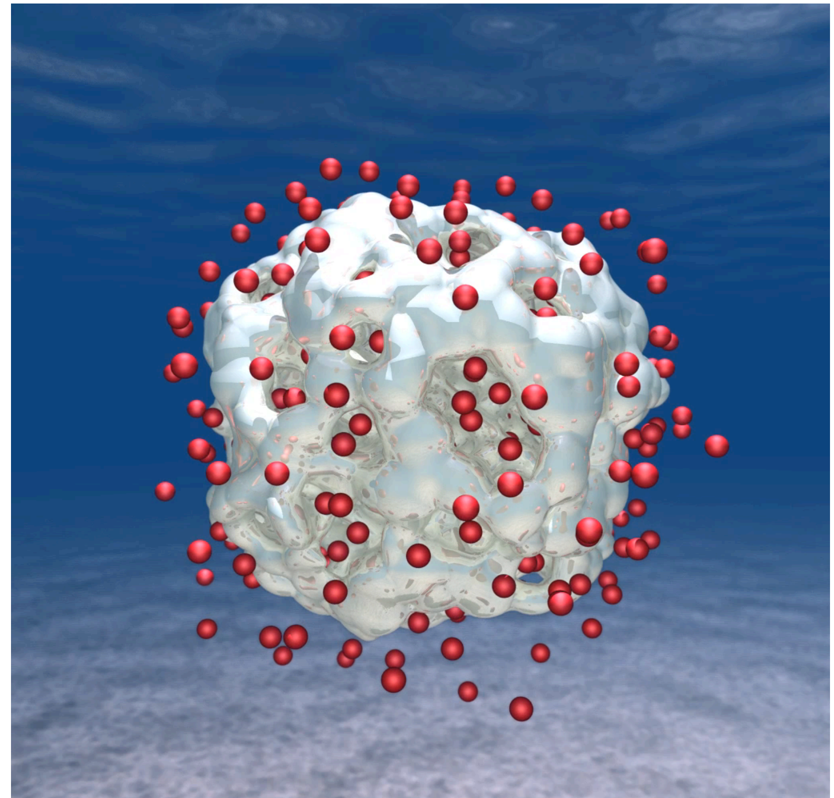


BASIC ENERGY SCIENCES

EXASCALE REQUIREMENTS REVIEW

An Office of Science review sponsored jointly by
Advanced Scientific Computing Research and Basic Energy Sciences

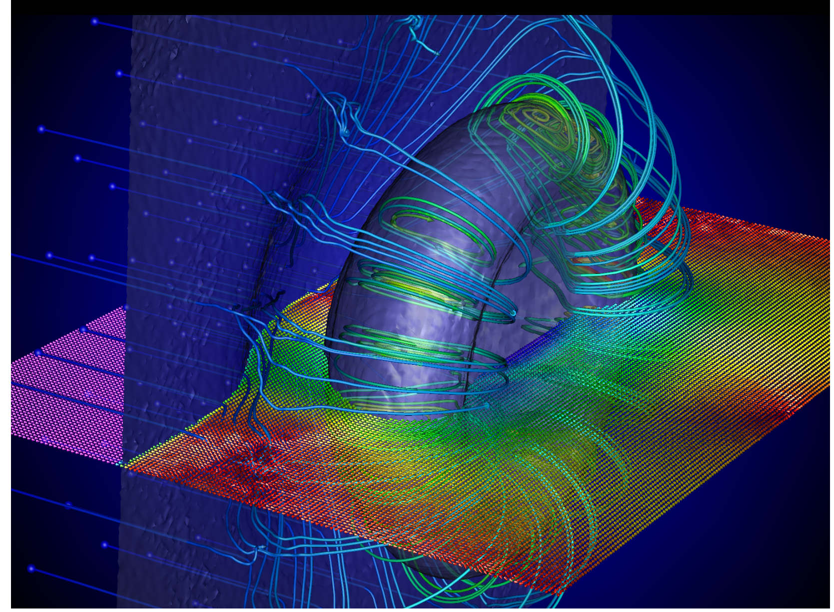
QD



16,661-atom quantum dynamics

Shimamura *et al*., *Nano Lett*. **14**, 4090 ('14)

MD



10^9-atom reactive molecular dynamics

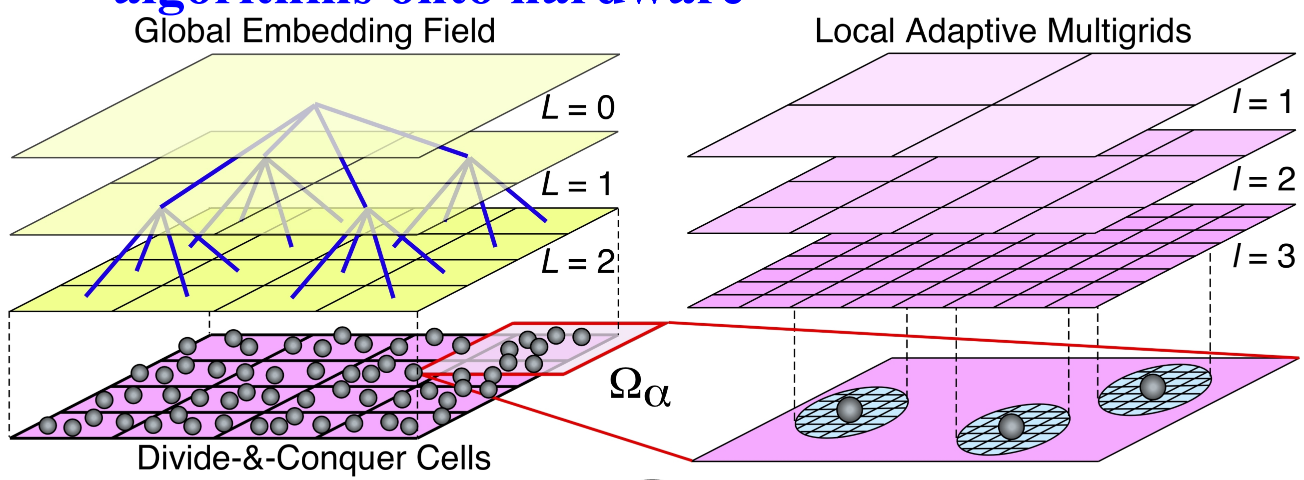
Shekhar *et al*., *Phys. Rev. Lett*. **111**, 184503 ('13)

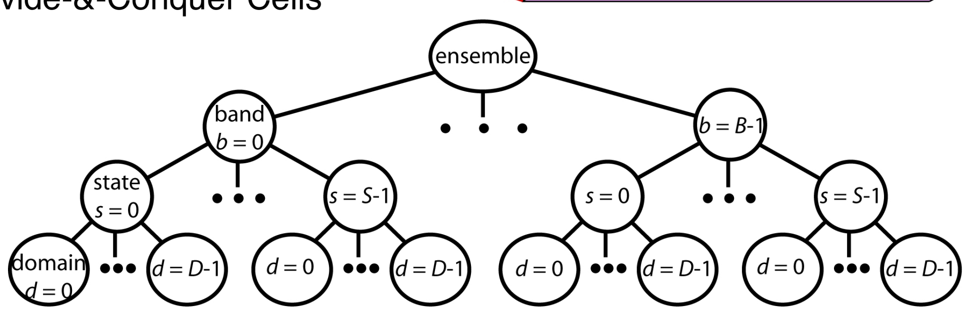
U.S. DEPARTMENT OF ENERGY

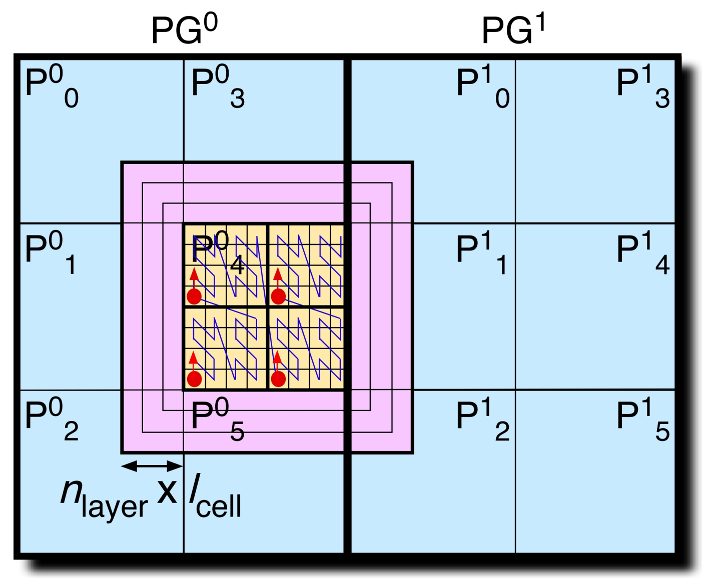
A Metascalable Dwarf

A metascalable (or "design once, scale on new architectures") parallel-computing framework for broad applications (*e.g.* equation solvers, constrained optimization, search, visualization and graphs)

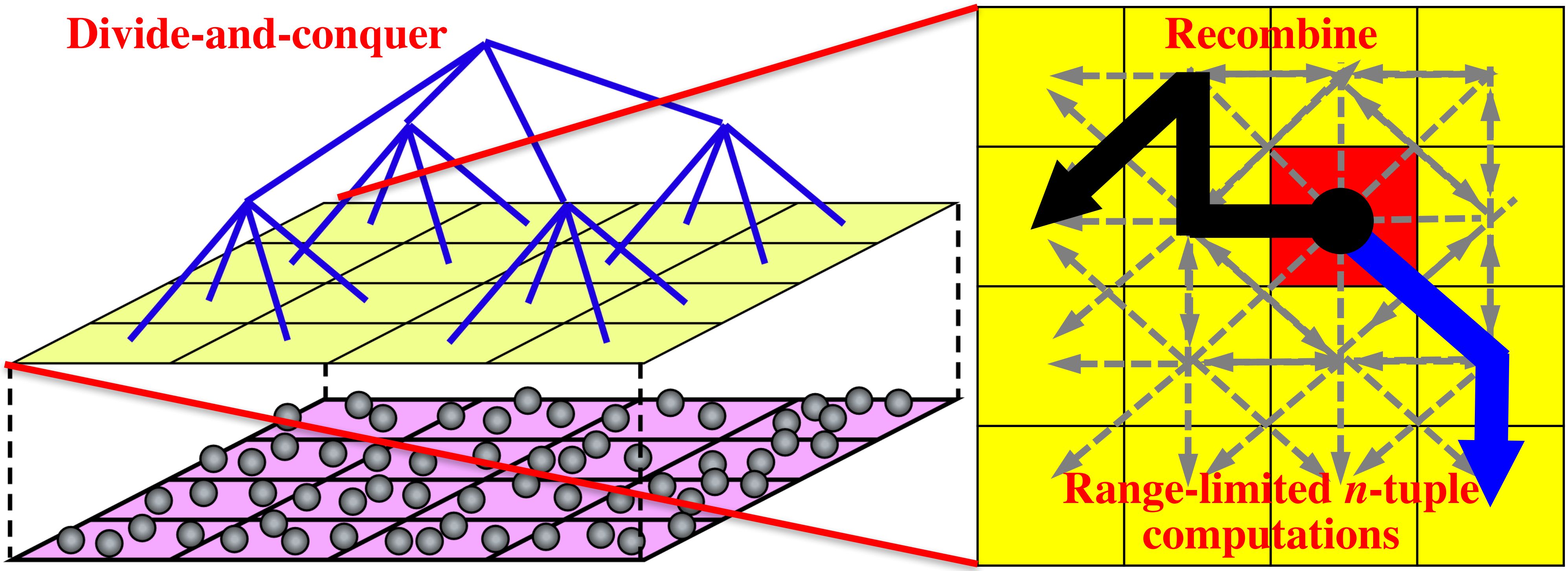
- Divide-conquer-"recombine" (DCR) algorithms based on spatial locality to design linear-scaling algorithms
- Space-time-ensemble parallel (STEP) approach based on temporal locality to predict long-time dynamics
- Tunable hierarchical cellular decomposition (HCD) to map these scalable algorithms onto hardware







Divide-Conquer-Recombine (DCR) Engines



M. Kunaseth *et al*., *ACM/IEEE SC13*

- **Lean divide-&-conquer density functional theory (LDC-DFT) algorithm minimizes the prefactor of $O(N)$ computational cost**

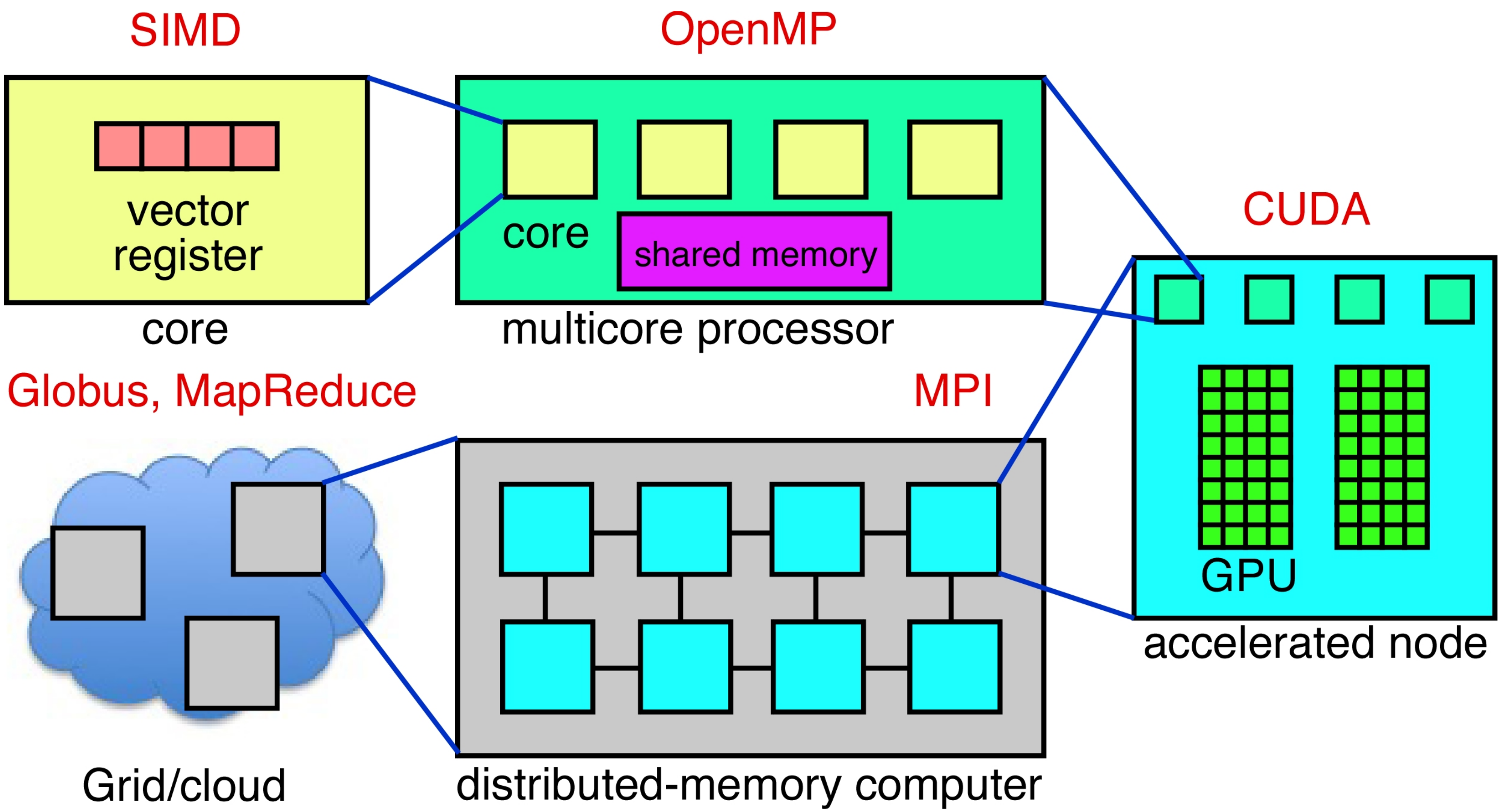
  F. Shimojo *et al*., *J. Chem. Phys.* **140**, 18A529 ('14); K. Nomura *et al*., *IEEE/ACM SC14*

- **Extended-Lagrangian reactive molecular dynamics (XRMD) algorithm eliminates the speed-limiting charge iteration**

  K. Nomura *et al*., *Comput. Phys. Commun*. **192**, 91 ('15); K. Liu *et al*., *IEEE/ACM ScalA18*

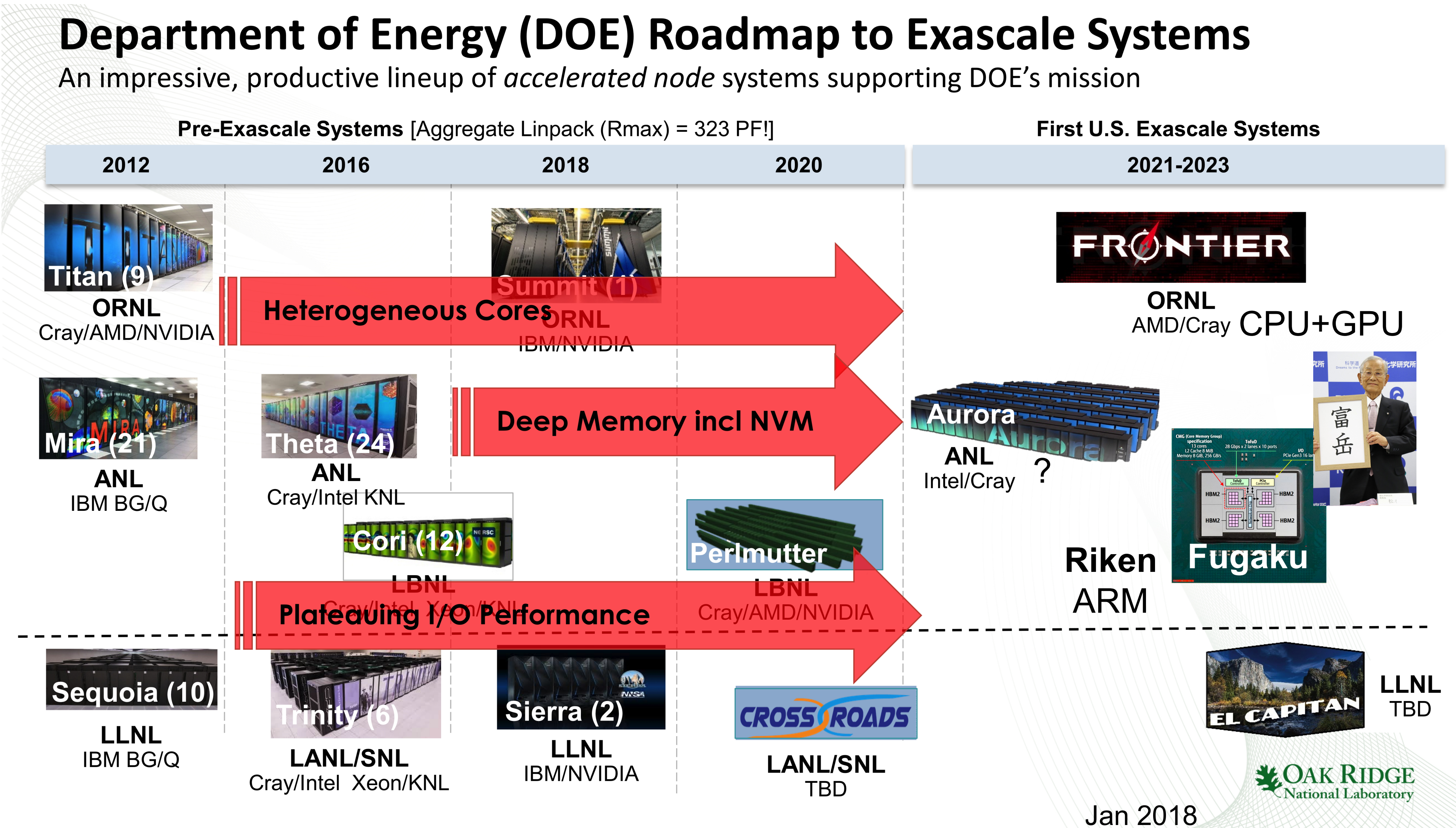
N. Romero *et al*., *IEEE Computer* **48(11)**, 33 ('15)

Hierarchical Parallel Computing



Near-Future High Performance Computing

- **CSCI 653 will bridge the current high performance computing (HPC) — MPI + OpenMP + CUDA — to near-future HPC**



Jeff Vetter (*ATPESC*, July 30, 2019)

Q: Think forward 10 years. How many of you predict that most of our top HPC systems will have the following architectural features?

a) X86 multicore CPU
b) GPU
c) FPGA/Reconfigurable processor
d) Neuromorphic processor
e) Deep learning processor
f) Quantum processor
g) RISC-V processor
h) Some new unknown processor
i) All/some of the above in one SoC

OAK RIDGE National Laboratory

https://extremecomputingtraining.anl.gov/sessions/dinner-talk-the-coming-age-of-extreme-heterogeneity/

Q: Now imagine you are building a new application with ~3M LOC and 20 team members over the next 10 years. What on-node programming model/system do you use?

What X in MPI+X?

a) C, C++, Fortran

b) C++ templates, policies, etc (e.g., AMP, Kokkos, RAJA,)

c) CUDA, cu***, HIP

d) OpenCL, SYCL

e) OpenMP or OpenACC

f) R, Python, Matlab, etc

g) A Domain Specific Language (e.g., Claw, PySL)

h) A Domain Specific Framework (e.g., PetSc)

i) Some new unknown programming approach

j) All/some of the above

OAK RIDGE National Laboratory

Watch David Patterson's IEEE Techtalk tomorrow (see class homepage)

https://extremecomputingtraining.anl.gov/sessions/dinner-talk-the-coming-age-of-extreme-heterogeneity/

Simulation + Data + Learning

- **Exaflop/s computers (Aurora & Frontier, also Fugaku) will be the first supercomputing platforms for simulation + data science + machine learning (*e.g*. low-precision tensor operations in deep learning)**
- **Compute Cambrian explosion @post-exaflop/s**
  https://www.forbes.com/sites/tomcoughlin/2019/04/26/compute-cambrian-explosion/#4f8da71d7986
- **Many traditional HPC applications will become extinct; find your niche (from this class) to survive**

http://www.thefutureiswild.com/

- **Useful resources for survival**
  > **ATPESC lectures:** https://extremecomputingtraining.anl.gov/agenda-2019/
  > **IPAM workshops on Machine Learning for Physics and the Physics of Learning:** https://www.ipam.ucla.edu/programs/long-programs/machine-learning-for-physics-and-the-physics-of-learning/

PostExa: Artificial Intelligence for Science

- **Post-exaflop/s national project**

The DOE National Laboratories are convening four town hall meetings aimed at collecting community input on the opportunities and challenges facing the scientific community in the era of convergence of High Performance Computing (HPC) and artificial intelligence (AI) technologies and the expected integration of large-scale simulation, advanced data analysis, data driven predictive modeling, theory, and high-throughput experiments. The term we are using to represent the next generation of methods and scientific opportunity is “AI for Science”.

- **See the first townhall program at Argonne National Laboratory**
- **See 20-year AI road map (CCC/AAAI, May 2019)**

Now What? Physics in 100 Years

- **Increasingly, the development of algorithms will become a central focus of theoretical physics. ... Triumphs of creative understanding such as universality (suppression of irrelevant details), symmetry (informed iteration), and topology (emergence of discrete from continuous) are preadapted to algorithmic thinking.**
- **The work of designing algorithms can be considered as a special form of teaching, aimed at extremely clever but literal-minded and inexperienced students—that is, computers—who cannot deal with vagueness. At present those students are poorly motivated and incurious, but those faults are curable. Within 100 years they (computers) will become the colleagues and ultimately the successors of their human teachers, with a distinctive style of thought adapted to their talents.**
- **Two developments will be transformative: naturalized artificial intelligence and expanded sensoria.**

F. Wilczek, *Phys. Today* **69(4)**, 32 ('16)

http://cacs.usc.edu/education/cs653/Wilczek-PhysicsIn100Years-PhysToday16.pdf

To sum:

HPC for science (previous CSCI 653)

→

AI for science (now & near future):

Survive the compute Cambrian explosion!

Discussion: assignment 1 + final project

3:30 pm, Friday, Aug. 30, VPD 106

http://cacs.usc.edu/education/cs653-assgn.html

Group project encouraged

Final-Project Publications (2019)

PAR^2: Parallel Random Walk Particle Tracking Method for solute transport in porous media☆ Computer Physics Communications 239 (2019) 265–271

Calogero B. Rizzo [a,*], Aiichiro Nakano [b], Felipe P.J. de Barros [a]

WaterAlignment: Identification of displaced water molecules in molecular docking using Jonker and Volgenant shortest path augmentation for linear assignment☆ Computer Physics Communications *in press*

Dab Brill [c,e,*], Jason B. Giles [e], Ian S. Haworth [e], Aiichiro Nakano [a,b,c,d,f]

sDMD: An Open Source Program for Discontinuous Molecular Dynamics Simulation of Protein Folding and Aggregation Computer Physics Communications *accepted*

Size Zheng[a], Leili Javidpour[b], Muhammad Sahimi[c], Katherine S. Shing[c], Aiichiro Nakano[c]

Adaptive Kinetic Architecture and Collective Behavior: A Dynamic Analysis for Emergency Evacuation
Buildings **2019**, *9*, 44

Angella Johnson [1,*], Size Zheng [2], Aiichiro Nakano [3], Goetz Schierle [1] and Joon-Ho Choi [1]

Boltzmann Machine Modeling of Layered MoS_2 Synthesis on a Quantum Annealer
submitted

Jeremy Liu[2,5], Ankith Mohan[2], Ken-ichi Nomura[1, 3], Aiichiro Nakano[1,2,3,4], Priya Vashishta[1,2,3,4], Ke-Thia Yao[5], Rajiv Kalia[1,2,3,4,*]