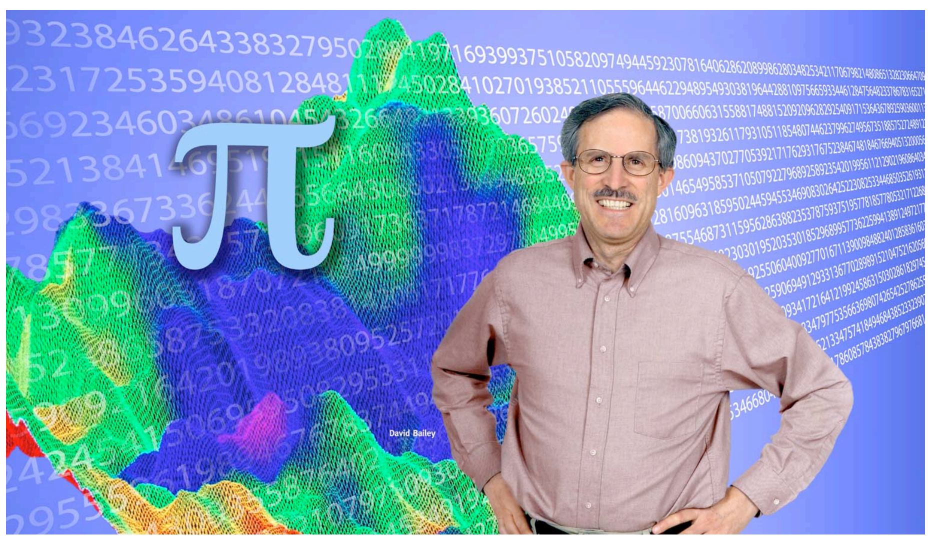
Computing:
The Third Mode of Scientific Discovery

David H Bailey

Lawrence Berkeley National Laboratory

`http://crd.lbl.gov/~dhbailey`



Laplace: Father of Modern Scientific Computing?

Computer simulations can be seen as the modern realization of a centuries-old dream known as the "clockwork universe." This was stated most clearly by Pierre Simon Laplace in 1773:

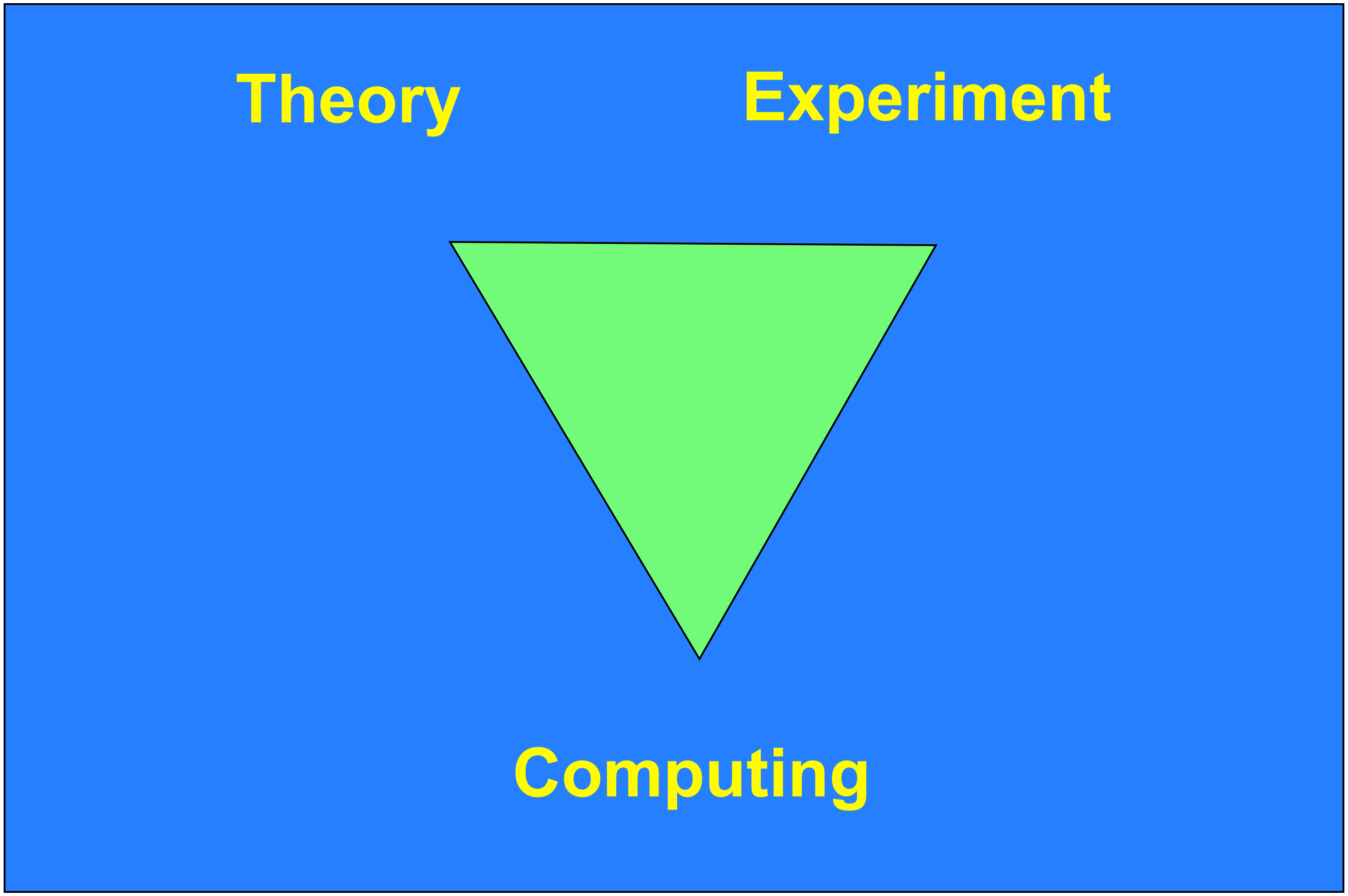
- An intelligence knowing all the forces acting in nature at a given instant, as well as the momentary positions of all things in the universe, would be able to comprehend in one single formula the motions of the largest bodies as well as of the lightest atoms in the world, provided that its intellect were sufficiently powerful to subject all data to analysis; to it nothing would be uncertain, the future as well as the past would be present to its eyes.

We now know that this dream, taken to its logical extreme, is unrealistic:

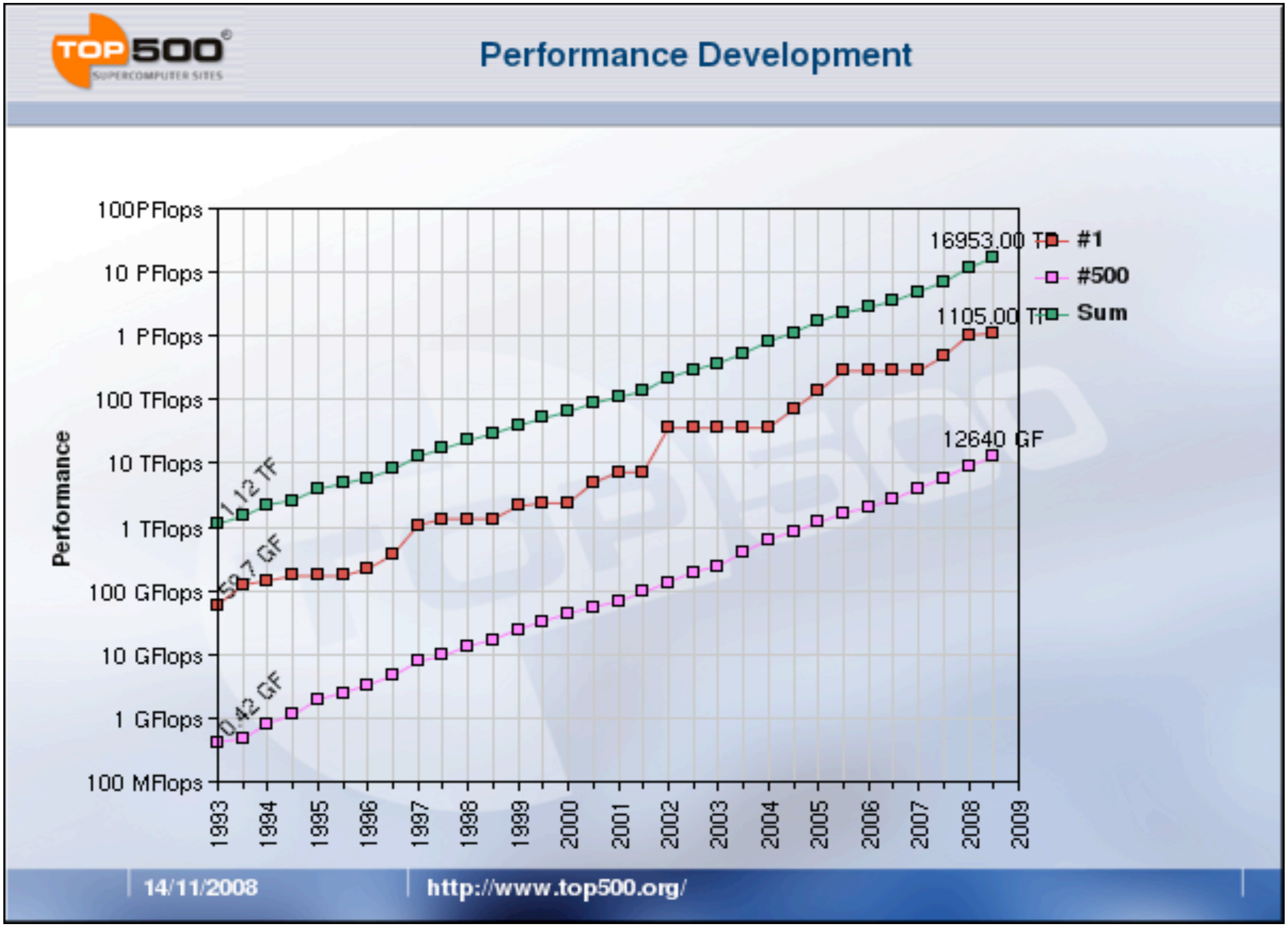
- Chaos theory teaches us that even simple physical systems exhibit chaos: slight changes to the present state are exponentially magnified.
- Quantum theory teaches us that it is fundamentally impossible to know both positions and velocities at any instant to arbitrarily high accuracy.

But many physical systems are amenable to computer simulation, and even chaotic systems can be studied computationally.

Computing:
The Third Mode of Scientific Discovery



Progress of Scientific Supercomputers: Data from the Top500 List



A Multi-Disciplinary Symphony of 21st Century Science and Engineering

- Basic physical laws.
- Mathematical formulations of these laws.
- Numerical algorithms to solve the mathematics.
- Computational techniques to implement algorithms.
- Grid generation.
- Multidimensional optimization techniques to explore parameter variations.
- Parallel computing methods.
- Scientific visualization.
- Performance monitoring and analysis.
- Computer system software.
- Computer system hardware.

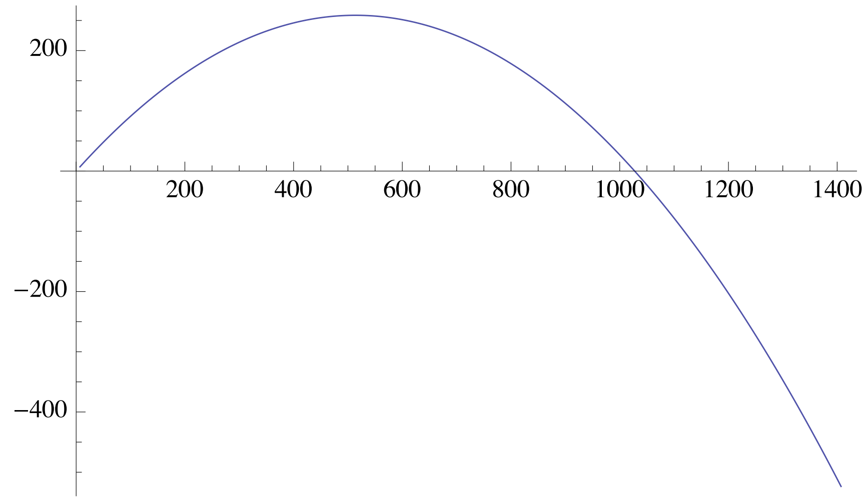
Basic Physical Laws

Many scientific computations are based on Newton's laws, e.g.: force = mass x accel, and gravity= G M_1 M_2 / R^2.

Many large computations are merely repeated applications of these laws on a grid.

Example:

- Suppose initial velocity = 100 m/s in a 45 deg direction (i.e. 70.71 m/s horiz and 70.71 m/s vert), and is subject to gravity.
- Then after 0.1 sec, horiz vel = 70.71 m/s, vert vel = 70.71 – 0.1x9.8 = 69.73 m/s, and object will have risen 0.1x70.71 = 7.071 m.
- Repeated applications of this calculation yields the curve at right. Even more accurate results can be had with a smaller time interval, e.g. 0.01 sec or 0.001 sec.
- This is just for one point-sized object. Large calculations do this for many objects.



Mathematical Formulations of Physical Laws

- All basic laws of physics can be encapsulated into mathematical equations. For example, Maxwell’s equations governing light and electromagnetic radiation can be written as:

$$
\begin{aligned}
\nabla \cdot E &= 4\pi\rho \\
\nabla \times E &= -\frac{1}{c}\frac{\partial B}{\partial t} \\
\nabla \cdot B &= 0 \\
\nabla \times B &= \frac{4\pi}{c}J + \frac{1}{c}\frac{\partial E}{\partial t}
\end{aligned}
$$

Numerical Algorithms

- Advanced numerical algorithms are used to solve the underlying mathematical formulations of physical laws.
- These algorithms dramatically lower the amount of computation normally required – modern scientific computing could not be done without them.
- Examples:
  - Dense linear algebra.
  - Sparse linear algebra.
  - Spectral methods (i.e., fast Fourier transforms).
  - N-body methods.
  - Structured grid methods for partial differential equations.
  - Unstructured grid methods for partial differential equations.
  - Distribute-reduce schemes.
  - Sorting, searching.
  - Optimization, maximization, minimization.
  - Many others.

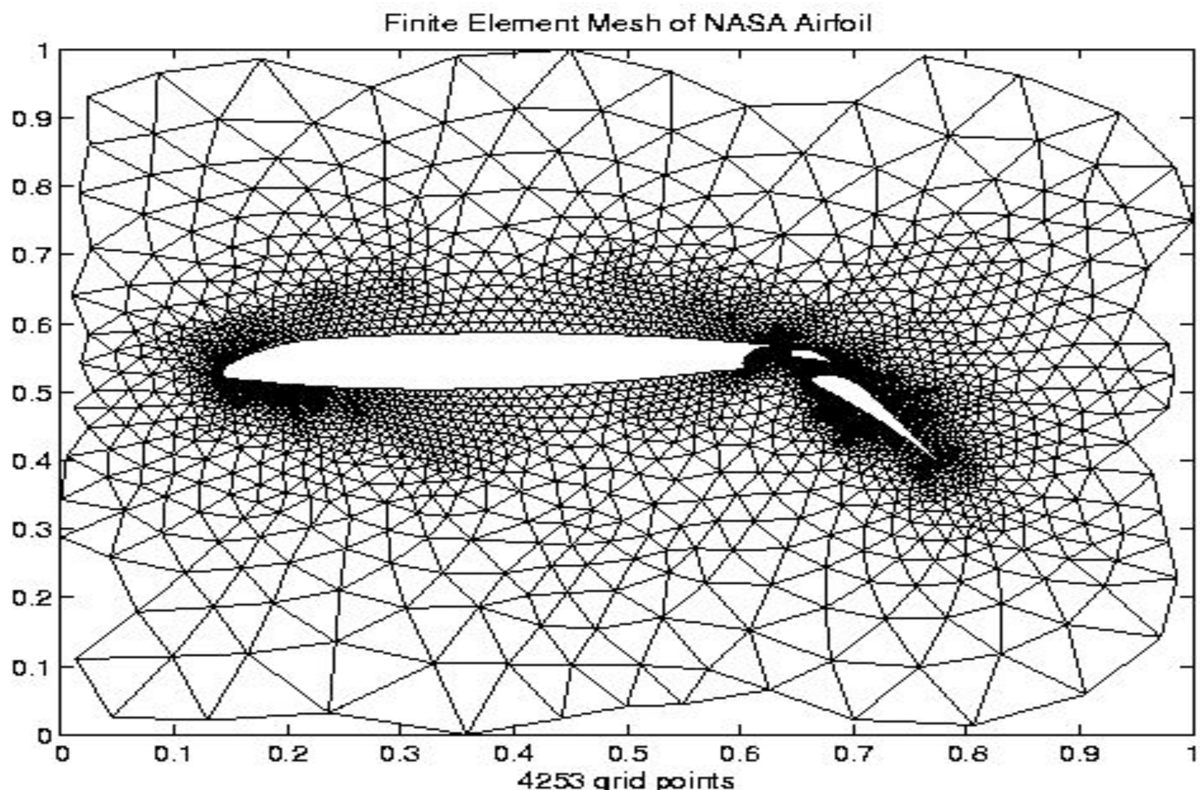
Computational Techniques

Because of the realities of modern computer architectures, efficient implementations of basic algorithms require considerable sophistication:

- Data locality is now very important:
  - The “hierarchical” design of processors (Level I cache, Level II cache, Level III cache, main memory, disk memory) means that considerable attention must be given to arranging computations for good performance.
  - It is often more efficient to recompute a value, rather than store it to memory and later fetch it.
- Changing computer designs means that computer programs that once were efficient now must be revised.
  - Floating-point multiply operations once were much more expensive than add/subtract operations, so codes were changed to exploit this. Now there is no difference.
  - Computation (add, subtract, multiply, divide, compare) was once much more expensive than fetching and storing data to memory. Now the opposite is true.

Grid Generation

- In many computational simulations, a key aspect of the computer implementation is the construction of a grid of points or polygons that encompass the physical object under study.
- Techniques for generating efficient and effective grids are an active field of research.



Optimization Techniques

- After a computational simulation program has been successfully developed, say to simulate the operation of a jet engine, then researchers run many instances of this simulation with variations of the input parameters, in an attempt to find an optimal configuration.
- Advanced optimization techniques permit these searches to be done significantly more efficiently than by exhaustive search.
- The resulting discipline of multidimensional optimization now is an essential part of high-performance computing.
- On some large HPC systems, almost all jobs perform multidimensional optimization.

Parallel Computing

- Parallel computing (using multiple processors for a single computation) until recently was the exclusive province of large government laboratories.
- Now, even consumer personal computers have more two, four, eight or more "cores," and it is essential to take advantage of these cores to fully utilize the power of the system.
- Scientists who do not convert their programs to utilize parallel computers will soon be left behind.
- *Parallel computing requires considerable sophistication*. Constructs must be inserted into computer programs to control:
  - Data layout.
  - Broadcast of data from the control node to other nodes.
  - Synchronization between nodes.
  - Communication between nodes.
  - Collection of data from all nodes back to a single node.
  - Parallel input and output of data to external disk drives.

Amdahl's Law

A simple principle first enunciated by Gene Amdahl in the 1960s places limits on speedup from parallel processing:

- If some fraction P of a computation is amenable to parallelization with speedup S, then the maximum possible speedup of the entire calculation is no more than $1 / ((1-P) + P/S)$.

For instance, if 99.9% of a computation can be effectively run in parallel on 10,000 processors, and the rest must be done serially, then the maximum speedup is only 909 (out of 10,000).

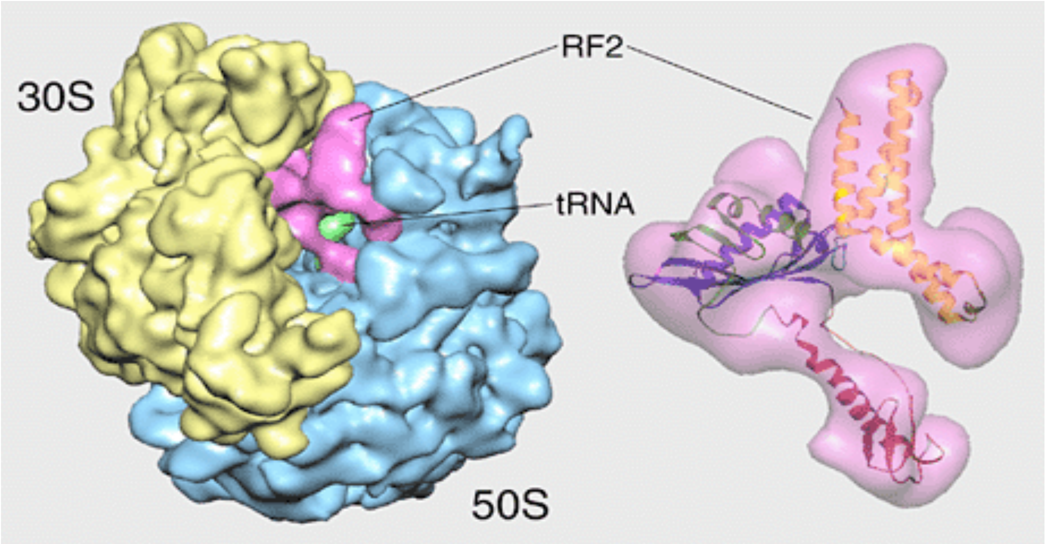
So far scientists working on large-scale computers have been able to keep Amdahl's Law at bay – mostly because the problems they wish to solve are always at the limit of available computer systems.

But how long will our luck hold out?

- In some arenas, such as climate modeling, computer programs are already pressing the limits of available concurrency.

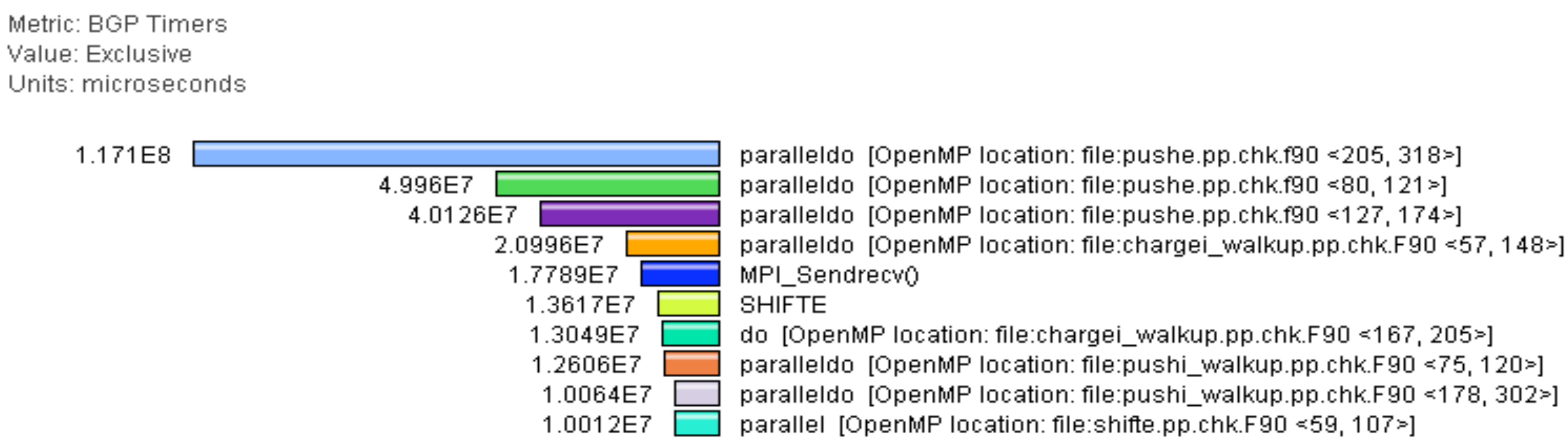
Scientific Visualization

- With the enormous volumes of data now involved in a large scientific simulation, it is no longer possible for a scientist to examine bits of data one by one.
- Instead, sophisticated scientific visualization software must be employed.
- Finding better ways to generate displays, and finding newer approaches to visualization, are active areas of research.



Performance Monitoring and Analysis

- Scientists are often disappointed in the performance (computational speed) of their programs – often only a few percent of the peak theoretical performance is achieved.
- Numerous sophisticated software tools are now available to help scientists better understand the performance of their programs on advanced computer architectures.
- The development of automatic performance tuning tools for scientific computer programs is an active area of research.



System Software

Underlying all of scientific computing is an enormous body of system software, without which scientific computing would not be possible:

- Operating system – Linux is the most widely used, although BSD Unix and IBM's AIX Unix are also used.
- Compilers – before any program is executed, it must be "compiled", i.e., translated to machine instructions.
- Modern compilers employ enormously sophisticated schemes to generate the most efficient possible machine code.
- Support for parallel computing.
- Support for large-scale data storage.
- Support for networking.

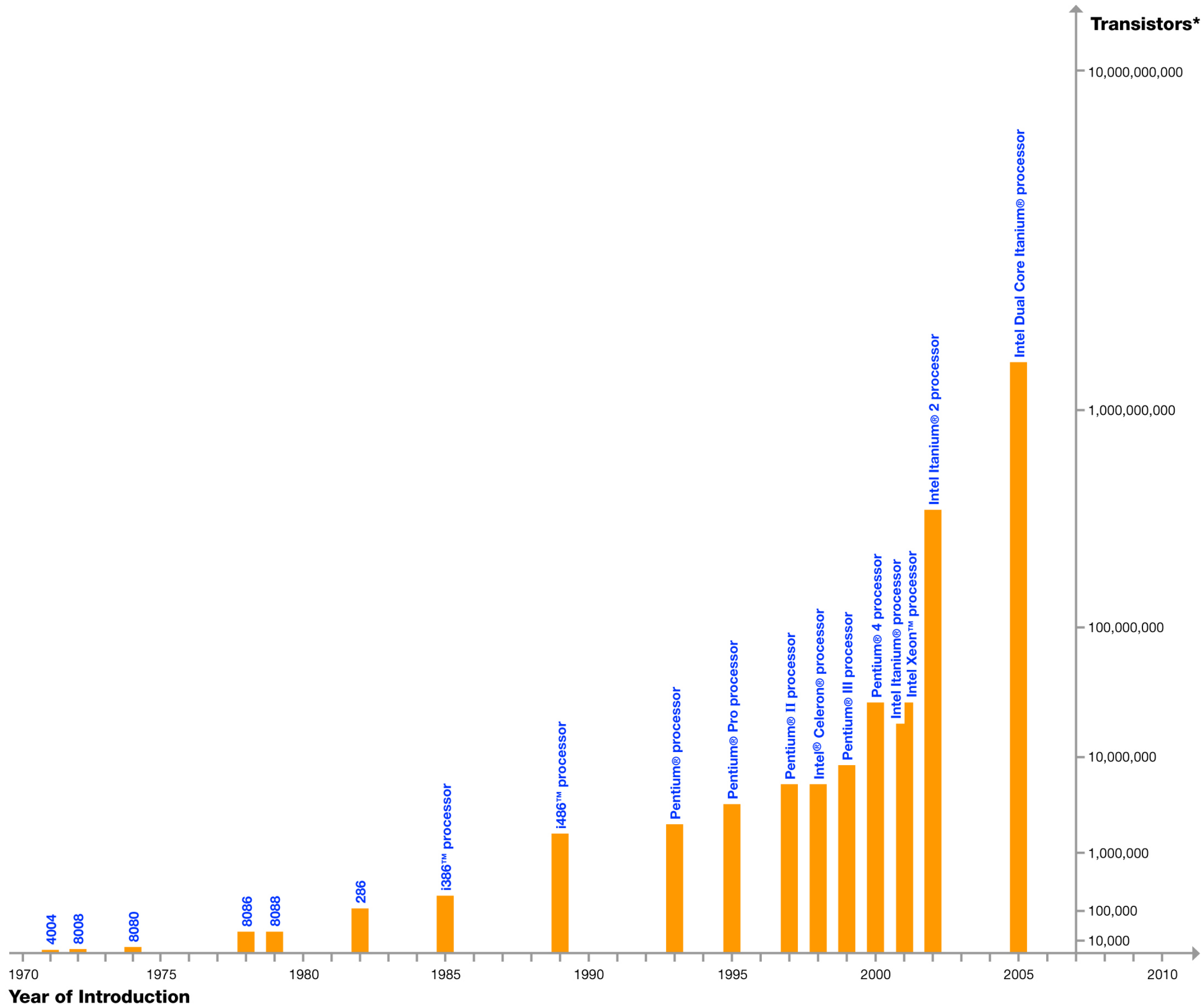
Computer Hardware

A large-scale scientific computer is much more than just a collection of chips – the large-scale system architecture is also important.

- Until about 2000, vector supercomputers were the most common platform for large-scale scientific computing.
- Now most scientific computing is done on large clusters of units, each of which is often an off-the-shelf personal computer system.
- The interprocessor network is very important – without a very high-speed network, many scientific computations would be mired in network congestion.

Moore's Law: 44 Years of Sustained Exponential Growth in Computer Power

From Computer Desktop Encyclopedia
Reproduced with permission.
© 2000 Intel Corporation



*Note: Vertical scale of chart not proportional to actual Transistor count.

In 1965 Gordon Moore (co-founder of Intel with Andy Grove and Robert Noyce of Grinnell College) predicted that the transistor density of semiconductor chips would double every 12-18 months. After 44 years, no end is yet in sight!

LBNL's National Energy Research Scientific Computing Center (NERSC)

- NERSC serves a large population of users:
  ~3000 users, ~400 projects, ~500 codes
- Allocations managed by the Department of Energy:
  - 10% INCITE awards:
    - Open to all of science, not just DOE-funded projects.
    - Large allocations, extra service.
  - 70% Production (ERCAP) awards:
    - From 10,000 CPU-hours to 5,000,000 CPU-hours.
  - 10% each NERSC and DOE reserve for special needs.

NERSC 2009 Configuration

Current flagship computer system:
- Cray XT4
- 9,740 nodes; 19,480 cores
- 78 Tbyte main memory
- 355 Tflops/s theoretical peak
- ~25 Tflops/s sustained on real scientific work

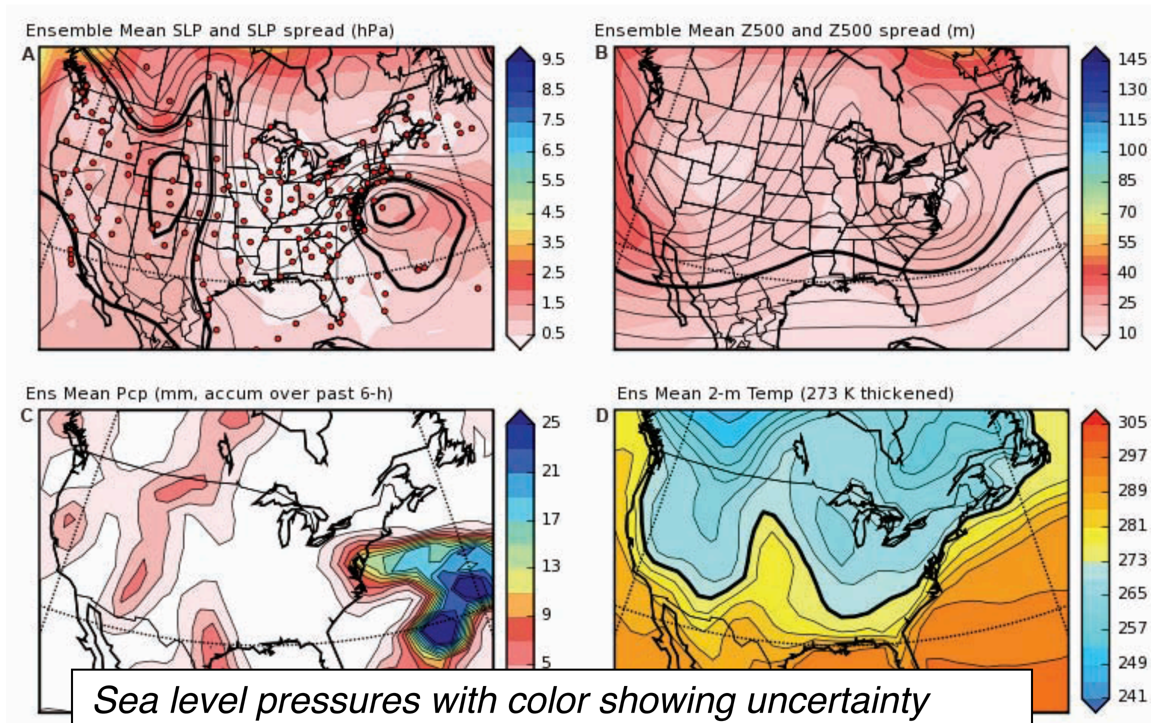
Newly announced upgrade (3Q 2010):
- Cray XT-5, 6400 nodes, 153,600 cores
- 1.9+ GHz AMD Opteron chips
- 1.17 Pflop/s peak performance
- Expect 100 Tflop/s sustained
- 217 TB DDR3 memory total
- Gemini Interconnect
- 2 PByte disk, 80 GByte/s bandwidth
- Liquid cooled

Validating Climate Models

- "20th Century Reanalysis" using an Ensemble Kalman filter to fill in missing climate data since; can be used for validation of models.
- Principal investigator: G. Compo, Univ. of Boulder.

- Science Results:
  - Reproduced 1922 Knickerbocker storm and dust storms of 1930s.
  - Building maps every 6 hours 1982-2008.
- Scaling Results:
  - Scales to 2400 cores.
  - Switched to higher resolution algorithm with Franklin access.

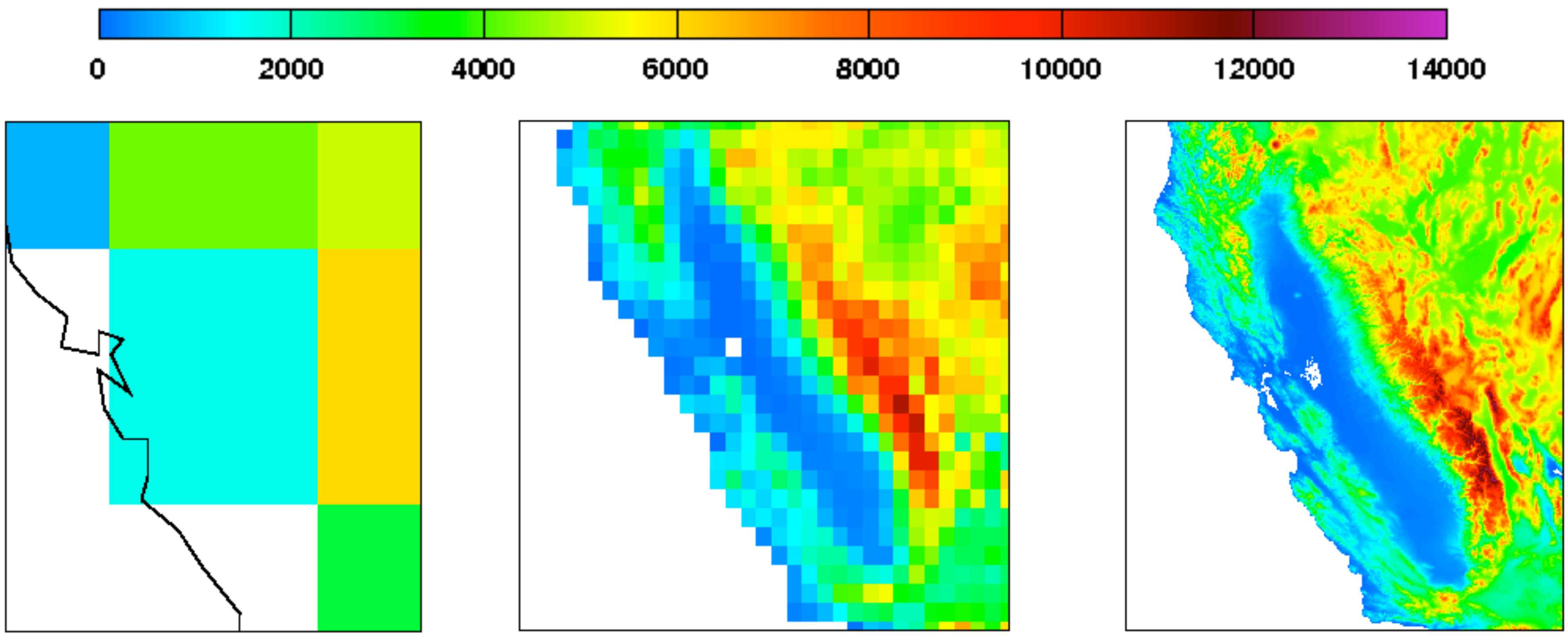
Granted 1,000,000 CPU-hours in 2009.



Sea level pressures with color showing uncertainty (a&b); precipitation (c); temperature (d). Dots indicate measurements locations (a).

Global Cloud System Resolving Models

Surface Altitude (feet)



200km
Typical resolution of
IPCC AR4 models

25km
Upper limit of climate
models with cloud
parameterizations

1km
Cloud system resolving
models
are a transformational
change

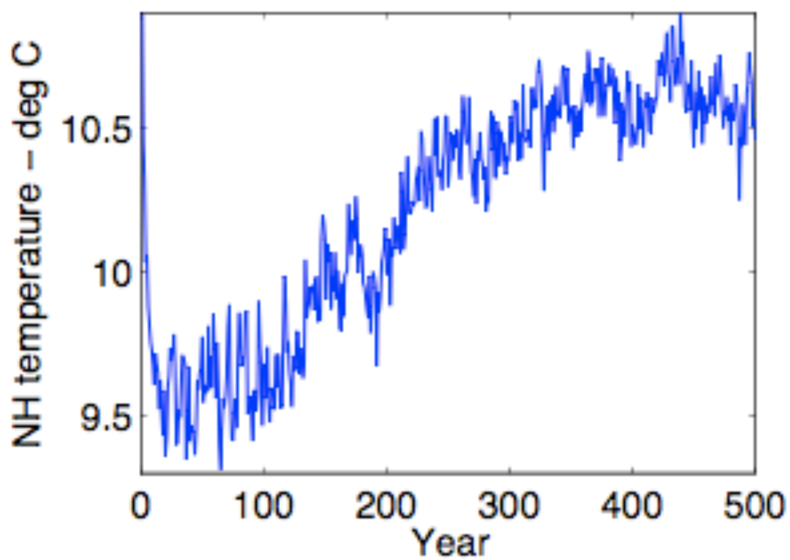
Climate Modeling with CCSM

- Climate Change Simulations with CCSM: Moderate and High Resolution Studies.
- Principal investigator: Warren Washington, NCAR.

- Science Results:
  - 2000-2100 simulation on preserving polar bear habitat by reducing non-CO_2 emissions.
  - Separating human and natural forcings in climate change.
  - Sulfate and carbon sulfate impact isolated.

  Granted 12,000,000 CPU-hours in 2009.

  Typical runs utilize 5000-6000 cores.



Material Modeling for Geoscience

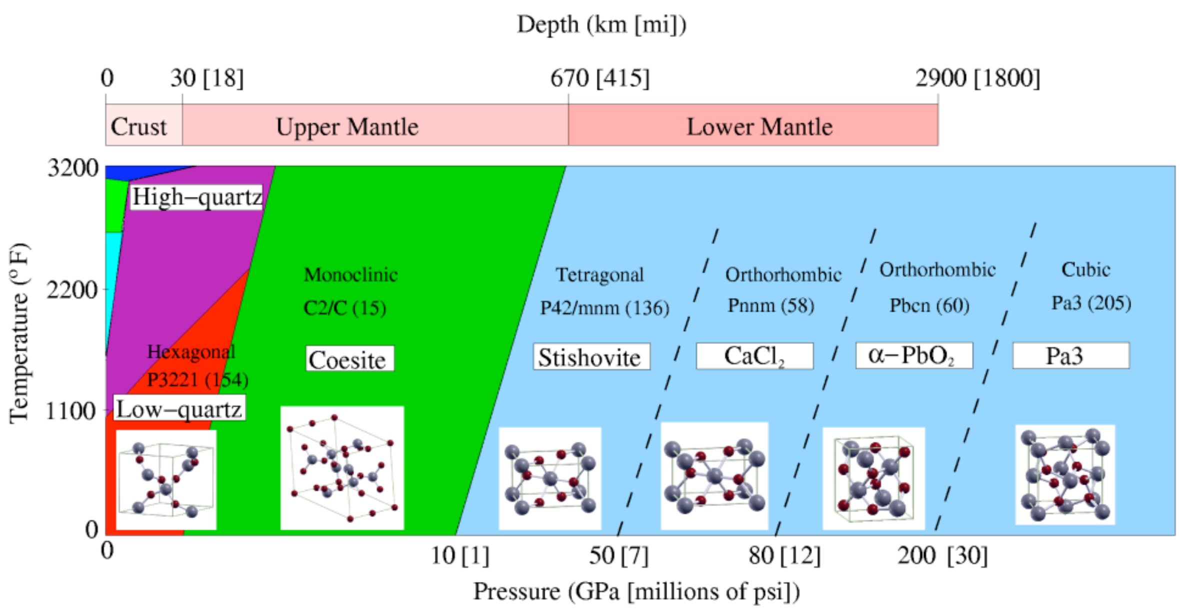
- Calculation: Simulation of seismic waves through silicates, which make up 80% of the Earth's mantle; important for understanding structures in oil well drilling, carbon sequestration, earthquakes, etc.
- PI: John Wilkins, Ohio State University

Science Result:

– Seismic analysis shows jumps in wave velocity due to structural changes in silicates under pressure

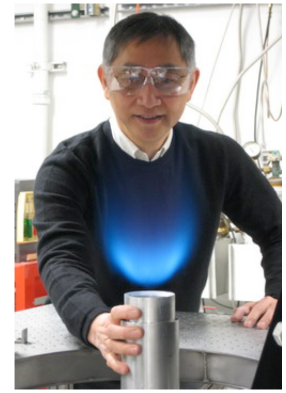
First use of Quantum Monte Carlo (QMC) for computing elastic constants.

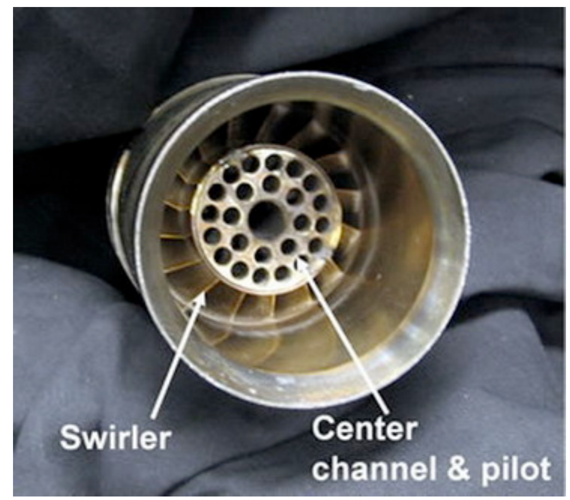
Typical runs utilize 8,000 cores.



Low-Swirl Combustion Burner Simulation

- Low-swirl burners were invented in 1991 at LBNL.
- They are now being developed for near-zero-emission gas turbines (2007 R&D 100 Award).
- They dramatically reduce pollutants by using special "lean premixed" fuels in power generation and transportation.
- Combustion with these fuels can be highly unstable, making robust systems hard to design.



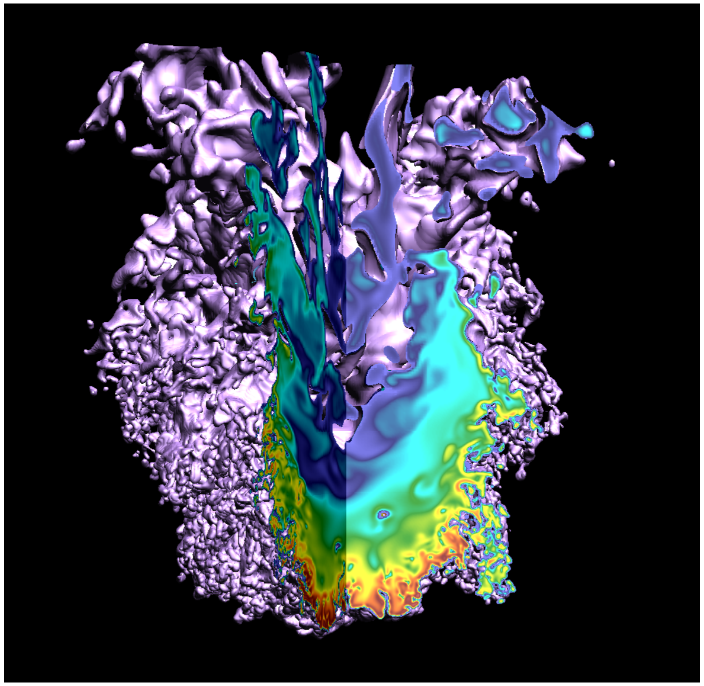


Low-Swirl Burner Simulation

- Computation: Numerical simulation of a lean premixed hydrogen flame in a laboratory-scale low-swirl burner (LMC code). Uses a low Mach number formulation, adaptive mesh refinement (AMR) and detailed chemistry and transport.
- PI: John Bell, LBNL

Science Result:

- Simulations capture cellular structure of lean hydrogen flames and provide a quantitative characterization of enhanced local burning structure.
- LMC dramatically reduces time and memory.
- Scales to 4K cores, typically run at 2K
- Used 9,600,000 CPU-hours in 2008; allocated 5,500,000 CPU-hours in 2009.



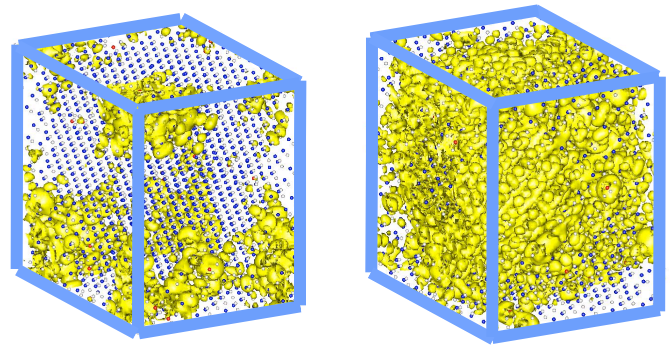
J B Bell, R K Cheng, M S Day, V E Beckner and M J Lijewski, Journal of Physics: Conference Series 125 (2008) 012027.

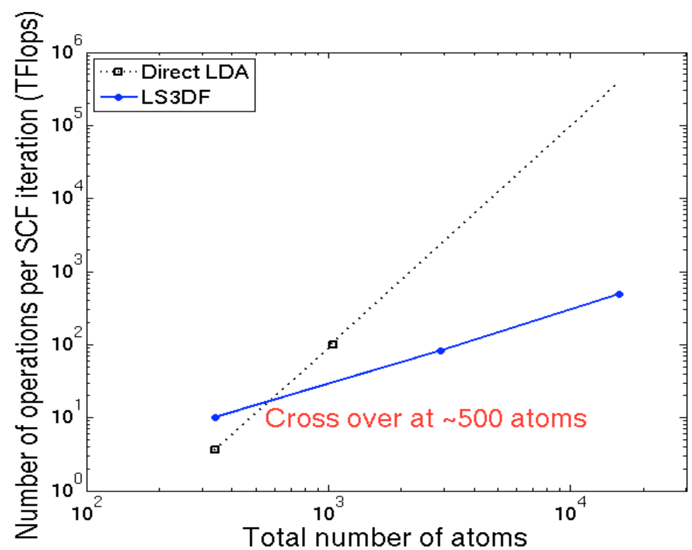
Nanoscience Calculations and Scalable Algorithms

- Calculation: Linear Scaling 3D Fragment (LS3DF). Density Functional Theory (DFT) calculation numerically equivalent to more common algorithm, but scales with O(n) in number of atoms rather than $O(n^3)$.
- Principal investigator: L.W. Wang, LBNL.

Science Results

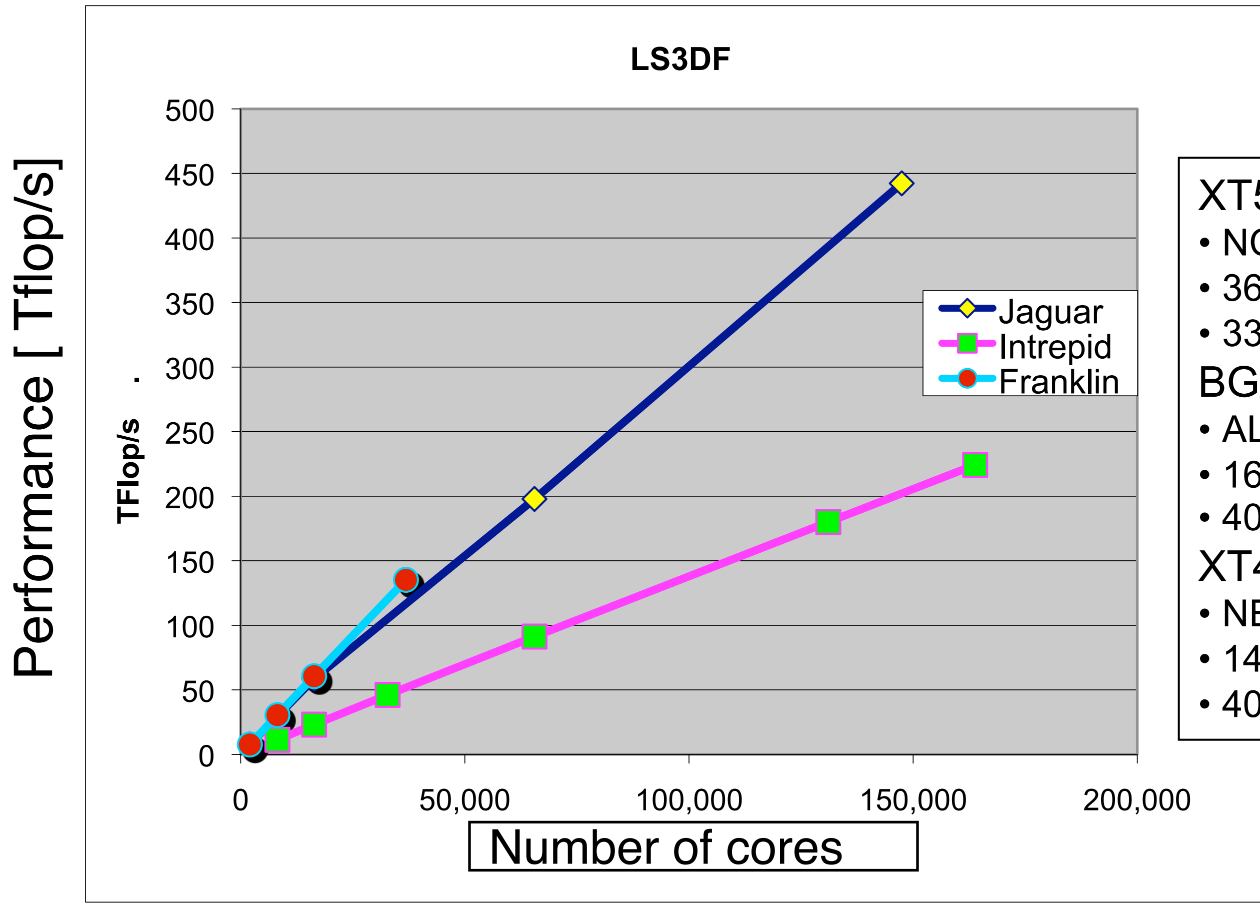
- Calculation of 3500 atom XnTe alloy used to predict efficiency of a new solar cell material.
- Ran on 17,000 cores.
- Took 1 hour vs months for previous $O(n^3)$ algorithm.
- Good efficiency (40% of peak).
- Won ACM Gordon Bell Prize at SC08.





ZnTeO Alloy Calculations Using LS3DF

- Run on Franklin (XT4) at NERSC led to the ACM Gordon Bell Award.
- Subsequent runs on Intrepid (BG/P) at Argonne and later Jaguar (XT5) at ORNL.

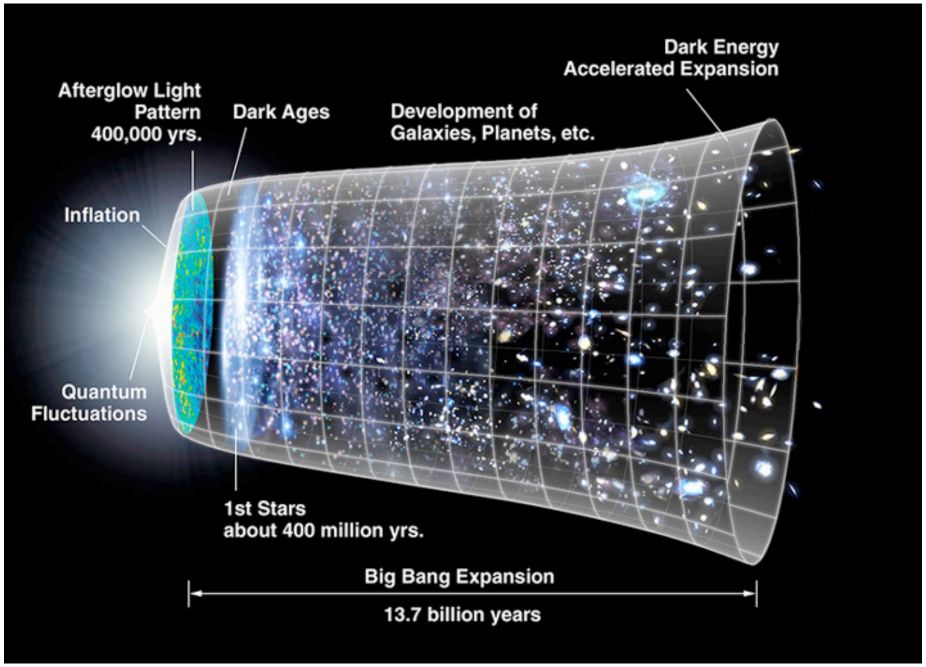


XT5: 442 Tflops
- NCCS Jaguar
- 36,864 cores
- 33% of peak

BG/P: 224 Tflops
- ALCF Intrepid
- 163,840 cores
- 40% of peak

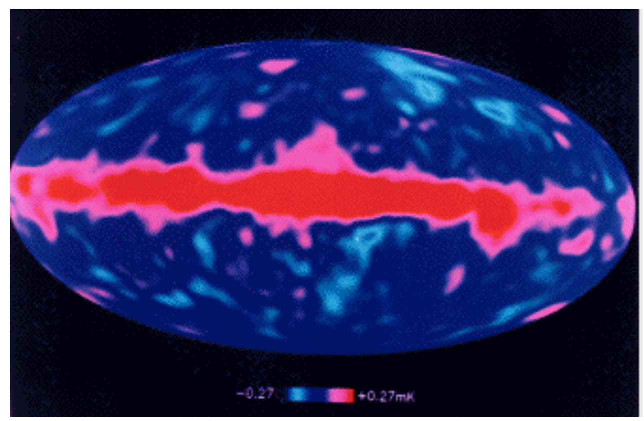
XT4: 139 TFlops
- NERSC Franklin
- 147,146 cores
- 40% of peak

NERSC User George Smoot wins 2006 Nobel Prize in Physics



Cosmic Microwave Background Radiation (CMB): an image of the universe at 400,000 years.

COBE Experiment showed anisotropy of CMB



Cosmic Microwave Data Analysis

- CMB data sets are very large:
  - Very low signal/noise per sample.
  - Higher resolution observations of fainter signals with more detectors at more frequencies.
- CMB data sets have to be analyzed as a whole:
  - Complex data correlations.
  - No simple “divide and conquer” scheme will work.

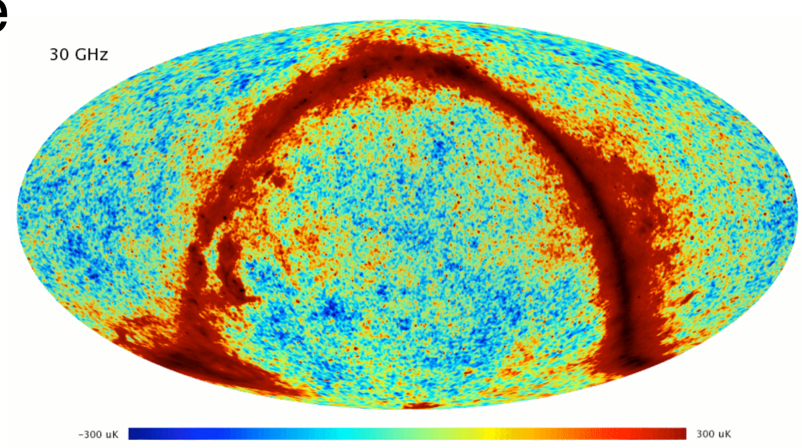
Experiment	Date	Size
Bell Labs	1965	1
COBE	1989	10^{9}
BOOMERanG	2000	10^{9}
WMAP	2001	10^{10}
Planck	2008	10^{11}
PolarBear	2010	10^{12}
CMBpol	>2020	10^{14}

Cosmic Microwave Background Computations

❖Calculation: Planck full focal plane

- 1 year simulation of CMB (T & P), detector noise & foregrounds.
- 74 detectors at 9 frequencies.
- 750 billion observations.
- 54,000 files, 3 TB data.

- Principal investigator: J. Borrill, LBNL.

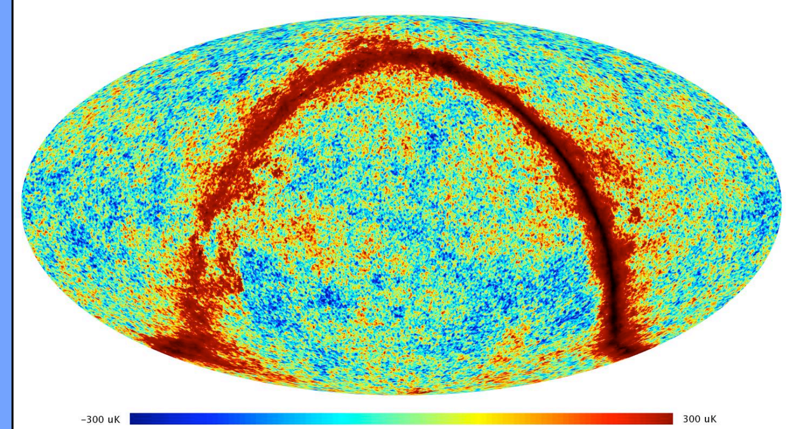


Science Result:

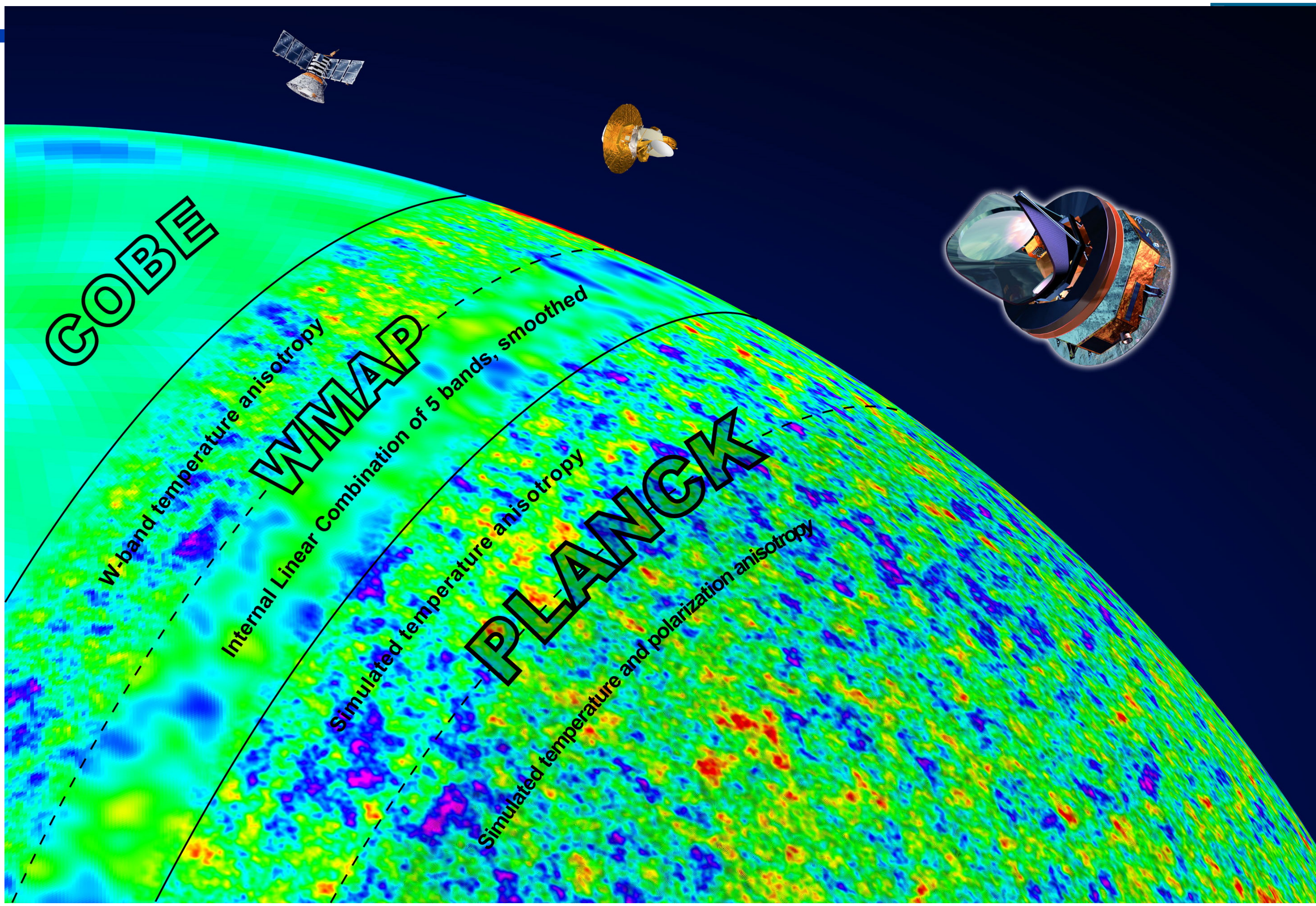
- 9 “routine” 1-frequency maps.
- Unprecedented 9-frequency map with entire simulated Planck data set.

Scaling Results:

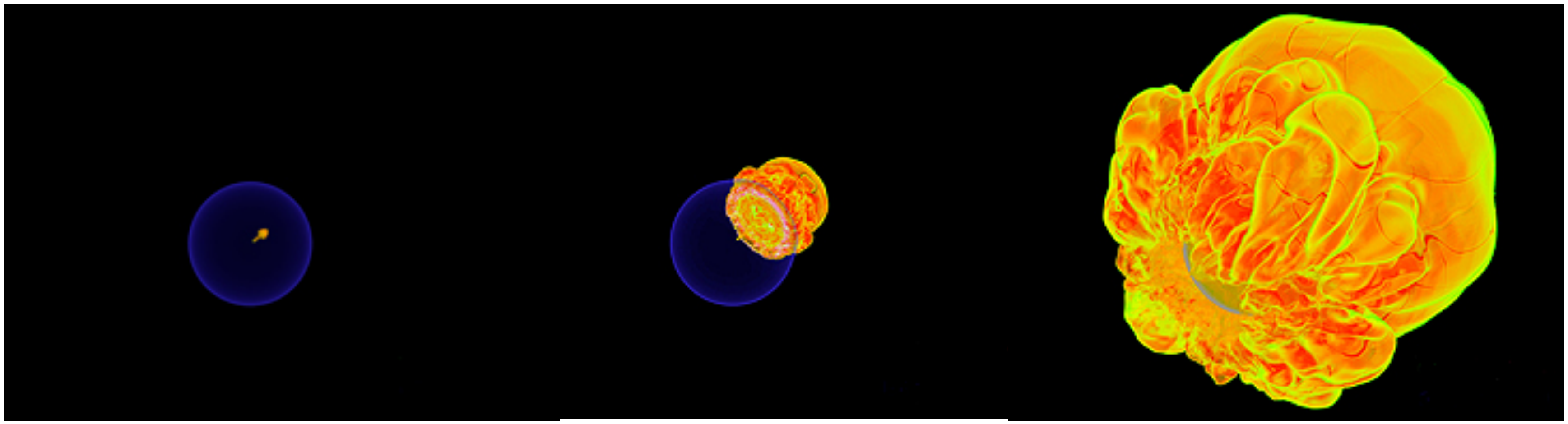
- 9-frequence problem ran for < 1 hour on 16K cores.



Evolution Of CMB Satellite Maps



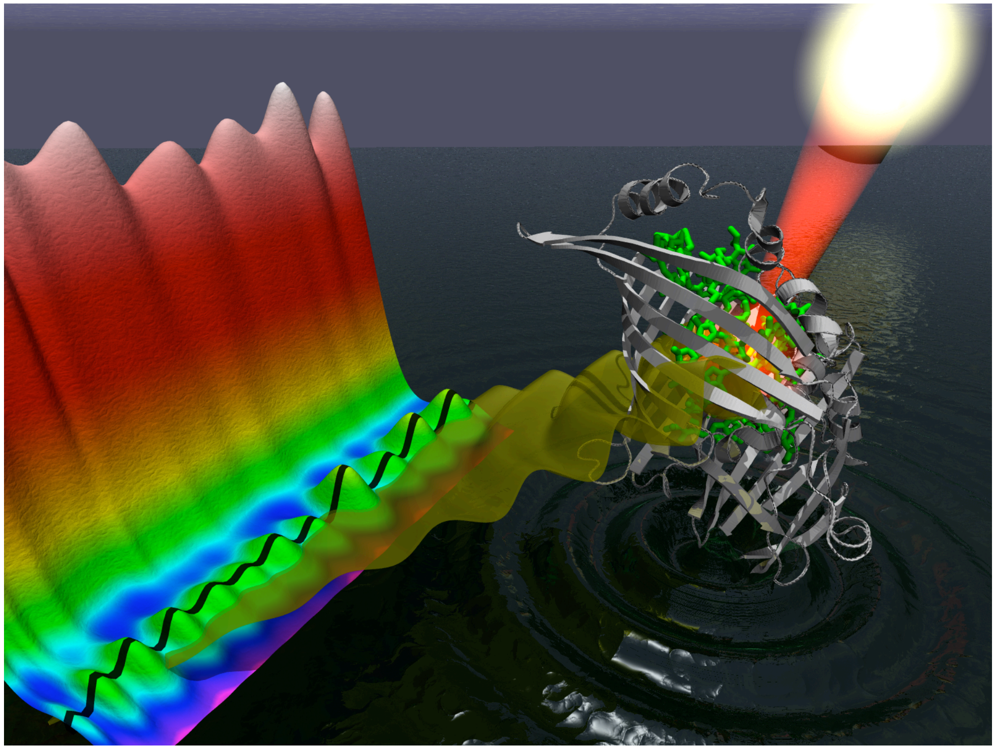
Type 1a Supernovae Simulation



Three phases of the gravitationally confined detonation mechanism. The images show the flame surface and the star at 0.5 s, 1.0 s, and 1.7 s.

- Simulations by D. Lamb et al, U. Chicago done at NERSC and LLNL.
- Show that gravitationally confined detonation may trigger the explosion of Type 1a supernovae.

Quantum Effects of Photosynthesis



Sunlight absorbed by bacteriochlorophyll (green) within the FMO protein (gray) generates a wavelike motion of excitation energy.

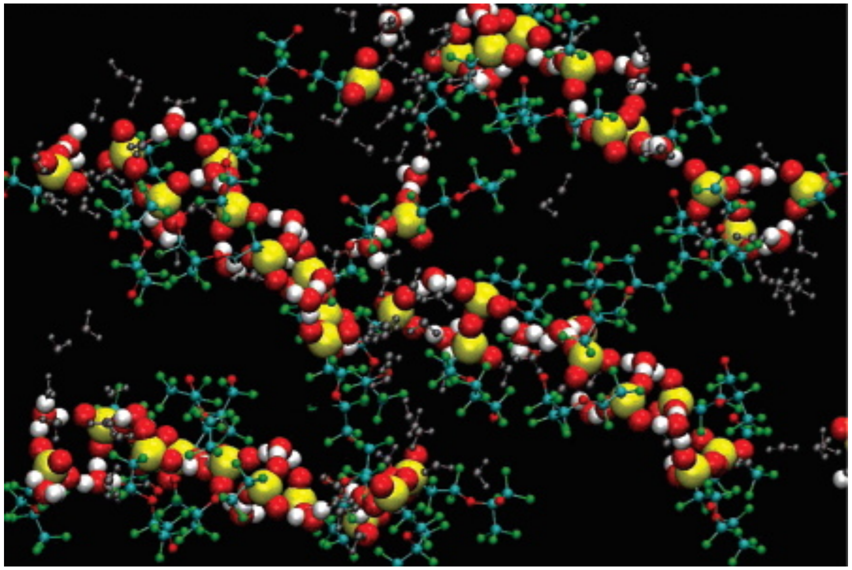
Quantum mechanical properties can be mapped through the use of two-dimensional electronic spectroscopy.

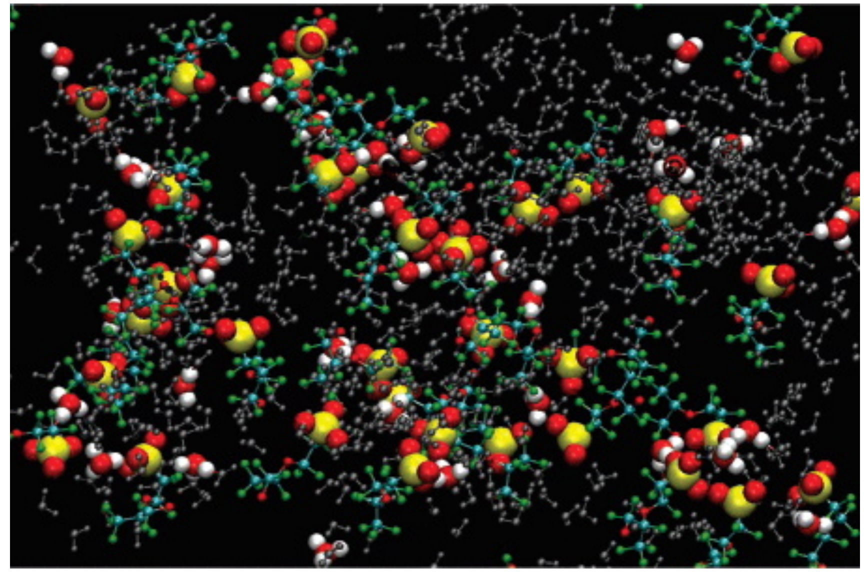
(Image courtesy of G. Engel)

Spectrum modeling and quantum dynamics simulations:

- Oscillations correlate with the quantum coherence in the system.
- Energy transfer pathways shown inside network of photosynthetic pigment-protein complexes.

Chemistry: Better Fuel Cell Design

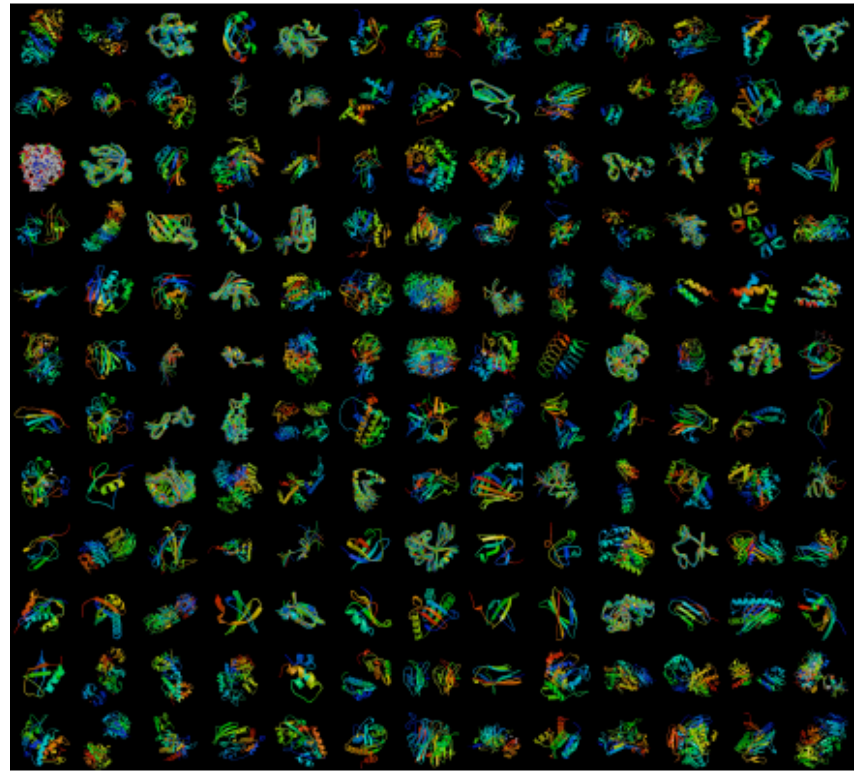




- Polymer electrolyte membrane (PEM) fuel cell requires right amount of hydration for optimal power output.
- Simulations of ionized Nafion varying levels of hydration (λ=3.5 and λ=16) reveal different properties
- 3 Phys. Chem. B papers by R. Devanathan, et al (PNNL).

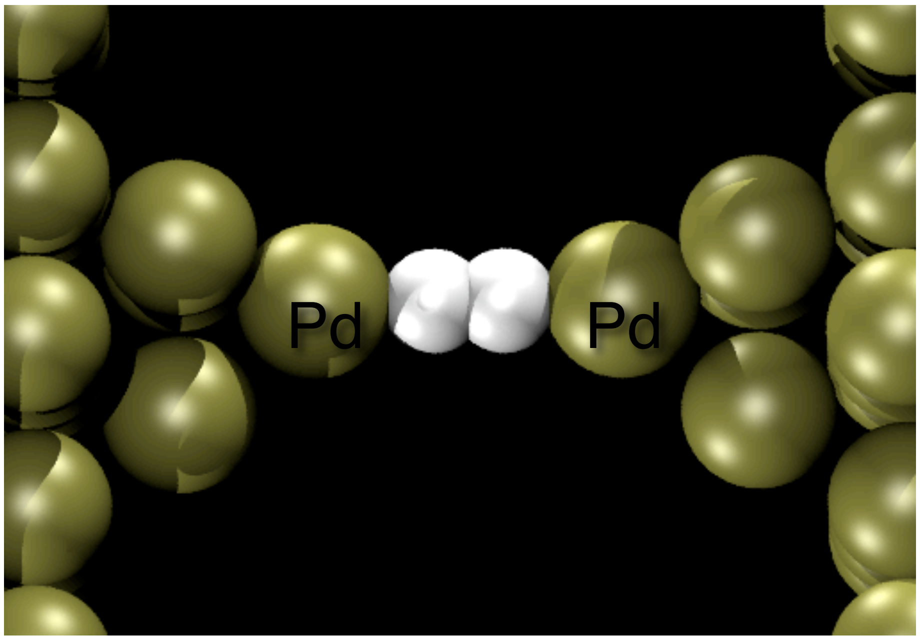
Molecular Dynameomics

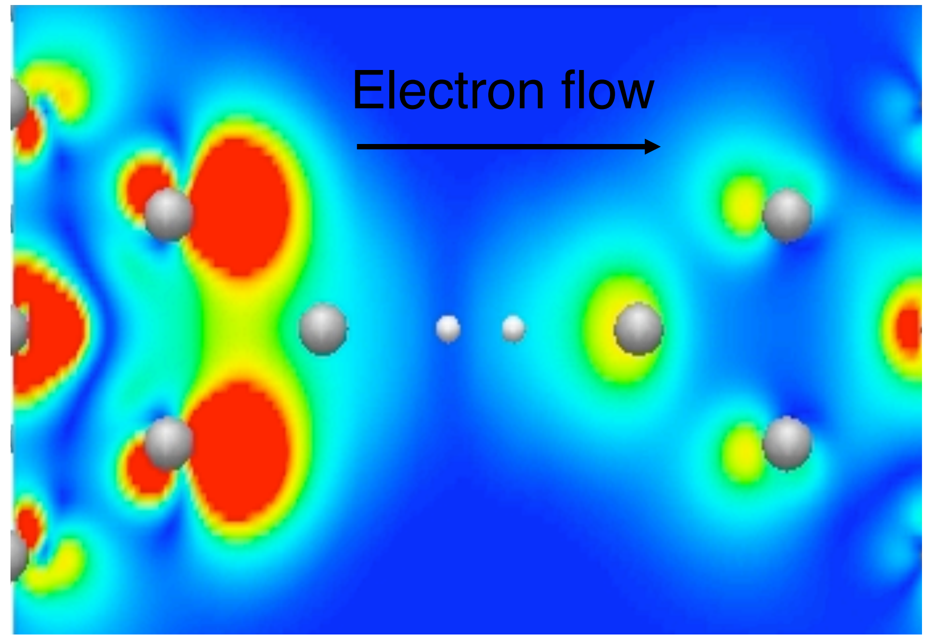
- V. Daggett, U Washington, 2,000,000 CPU-hours.
- Understand protein folding pathways by ‘unfolding’ proteins at high temperature.
- Computed unfolding of 151 most common fold structures at different temperatures.
- Multiple runs of MD calculation for each fold/temperature pair.



The first 156 protein targets

Exploring Nanoelectronics with Theory: Single Molecule Electrical Junctions





Shown at left is a single hydrogen molecule (in white) bridging palladium point contacts. At right, a density plot of the dominant transmitting electronic state reveals a significant reflection of charge at the left Pd contact, leading to a high resistance, consistent with recent experiments. (Red is high electronic density in the plot, blue is low.)

Steven Louie,
Marvin Cohen,
UC Berkeley
Jeff Neaton, LBNL
Molecular Foundry

Discovering and Verifying New Mathematical Formulas by Computer

One surprising new application of high-performance computing is the discovery of previously unknown mathematical identities, via a combination of high-precision calculations (typically 100-1000 digits), combined with integer relation algorithms to identify the calculated constants. Examples:

$$\pi = \sum_{n=0}^{\infty} \frac{1}{16^k} \left(\frac{4}{8k+1} - \frac{2}{8k+4} - \frac{1}{8k+5} - \frac{1}{8k+6} \right)$$

$$\pi^2 = \frac{1}{8} \sum_{k=0}^{\infty} \frac{1}{64^k} \left(\frac{144}{(6k+1)^2} - \frac{216}{(6k+2)^2} - \frac{72}{(6k+3)^2} - \frac{54}{(6k+4)^2} + \frac{9}{(6k+5)^2} \right)$$

$$\sum_{k=1}^{\infty} \frac{1}{k^2 - x^2} = 3 \sum_{k=1}^{\infty} \frac{1}{k^2 \binom{2k}{k} (1 - x^2/k^2)} \prod_{m=1}^{k-1} \left(\frac{1 - 4\, x^2/m^2}{1 - x^2/m^2} \right)$$

$$L_{-7}(2) = \frac{24}{7\sqrt{7}} \int_{\pi/3}^{\pi/2} \log \left| \frac{\tan t + \sqrt{7}}{\tan t - \sqrt{7}} \right| \mathrm{d}t$$

$$\int_0^1 \int_0^1 \int_0^1 \frac{\mathrm{d}r_1 \mathrm{d}r_2 \mathrm{d}r_3 \mathrm{d}s_1 \mathrm{d}s_2 \mathrm{d}s_3}{\sqrt{(r1 - s1)^2 + (r2 - s2)^2 + (r3 - s3)^2}}$$

$$= \frac{1}{15} \left(6 + 6\sqrt{2} - 12\sqrt{3} - 10\pi + 30 \log(1 + \sqrt{2}) + 30 \log(2 + \sqrt{3}) \right)$$

Summary

- Scientific computing has vastly expanded in sophistication and power over the past 40 years, and is now widely regarded as a third mode of scientific discovery, after theory and experiment.
- Modern scientific computing is a multidisciplinary “symphony” involving scientists and engineers from many fields.
- The power of the leading-edge systems has closely followed Moore’s Law in an exponentially upward path, and no end is yet in sight.

Major challenges:

- In order to utilize future systems efficiently, computer programs must possess and exhibit enormous parallelism (10^{10}-way or more).
- Exotic architectures (e.g., hybrid systems employing game processors) will present difficult programming and software challenges.
- Daunting electric power requirements projected for future systems will require innovation in hardware, software and applications.