Preparing Applications for Next-Generation HPC Architectures

Andrew Siegel
Argonne National Laboratory
Exascale Computing Project (ECP) is part of a larger US DOE strategy
Exascale Computing Project

• Department of Energy project to develop usable exascale ecosystem

• Exascale Computing Initiative (ECI)
  1. 2 Exascale platforms (2021)
  2. Hardware R&D
  3. System software/middleware
  4. 25 Mission critical application projects
<table>
<thead>
<tr>
<th>Pre-Exascale Systems</th>
<th>Exascale Systems</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>2013</strong></td>
<td><strong>2021-2023</strong></td>
</tr>
<tr>
<td>Mira</td>
<td>Argonne Intel/Cray TBD Open</td>
</tr>
<tr>
<td>Titan</td>
<td>ORNL Cray/NVidia K20 Open</td>
</tr>
<tr>
<td>Sequoia</td>
<td>LLNL IBM BG/Q Secure</td>
</tr>
<tr>
<td>Theta</td>
<td>LBNL Cray/Intel Xeon/KNL Open</td>
</tr>
<tr>
<td>Trinity</td>
<td>LANL/SNL Cray/Intel Xeon/KNL Secure</td>
</tr>
<tr>
<td>Summit</td>
<td>ORNL IBM/NVidia P9/Volta Open</td>
</tr>
<tr>
<td>ORNL IBM/NVidia P9/Volta Secure</td>
<td>LBNL TBD Open</td>
</tr>
<tr>
<td>Secure</td>
<td>Open</td>
</tr>
<tr>
<td>NERSC-9</td>
<td>Frontier ORNL TBD Open</td>
</tr>
<tr>
<td>Crossroads</td>
<td>El Capitan LLNL TBD Secure</td>
</tr>
<tr>
<td>LLNL IBM BG/Q Cray/Intel Xeon/KNL Secure</td>
<td></td>
</tr>
</tbody>
</table>
Building an Exascale Machine

• Why is it difficult?
  – Dramatically improve power efficiency to keep overall power 20-40MW
  – Provide useful FLOPs: algorithms with efficient (local) data movement

• What are the risks?
  – End up with Petscale performance on real applications
  – Exascale on carefully chosen benchmark problems only
Microprocessor Transistors / Clock (1970-2015)
## Fastest Computers: HPL Benchmark

<table>
<thead>
<tr>
<th>Rank</th>
<th>Site</th>
<th>System</th>
<th>Cores</th>
<th>Rmax [TFlop/s]</th>
<th>Rpeak [TFlop/s]</th>
<th>Power [kW]</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>National Supercomputing Center in Wuxi</td>
<td>Sunway TaihuLight - Sunway MPP, Sunway SW26010 260C 1.45GHz, Sunway NRCPC</td>
<td>10,649,600</td>
<td>93,014.6</td>
<td>125,435.9</td>
<td>15,371</td>
</tr>
<tr>
<td>2</td>
<td>National Super Computer Center in Guangzhou</td>
<td>Tianhe-2 (MilkyWay-2) - TH-IVB-FEP Cluster, Intel Xeon E5-2692 12C 2.20GHz, TH Express-2, Intel Xeon Phi 31S1P NUDT</td>
<td>3,120,000</td>
<td>33,862.7</td>
<td>54,902.4</td>
<td>17,808</td>
</tr>
<tr>
<td>3</td>
<td>Swiss National Supercomputing Centre (CSCS)</td>
<td>Piz Daint - Cray XC50, Xeon E5-2690v3 12C 2.6GHz, Aries interconnect, NVIDIA Tesla P100 Cray Inc.</td>
<td>361,760</td>
<td>19,590.0</td>
<td>25,326.3</td>
<td>2,272</td>
</tr>
<tr>
<td>4</td>
<td>Japan Agency for Marine-Earth Science and Technology</td>
<td>Gyoukou - ZettaScaler-2.2 HPC system, Xeon D-1571 16C 1.3GHz, Infiniband EDR, PEZY-SC2 700Mhz ExaScaler</td>
<td>19,860,000</td>
<td>19,135.8</td>
<td>28,192.0</td>
<td>1,350</td>
</tr>
<tr>
<td>5</td>
<td>DOE/SC/Oak Ridge National Laboratory United States</td>
<td>Titan - Cray XK7, Opteron 6274 16C 2.20GHz, Cray Gemini interconnect, NVIDIA K20x Cray Inc.</td>
<td>560,640</td>
<td>17,590.0</td>
<td>27,112.5</td>
<td>8,209</td>
</tr>
</tbody>
</table>
## Fastest Computers: HPCG Benchmark

<table>
<thead>
<tr>
<th>Rank</th>
<th>Site Description</th>
<th>Computer Details</th>
<th>Cores</th>
<th>HPL Rmax (Pflop/s)</th>
<th>TOP500 Rank</th>
<th>HPCG (Pflop/s)</th>
<th>Fraction of Peak</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>RIKEN Advanced Institute for Computational Science, Japan</td>
<td>K computer – , SPARC64 VIIIfx 2.0GHz, Tofu interconnect</td>
<td>705,024</td>
<td>10.510</td>
<td>10</td>
<td>0.603</td>
<td>5.3%</td>
</tr>
<tr>
<td>2</td>
<td>NSCC / Guangzhou, China</td>
<td>Tianhe-2 (MilkyWay-2) – TH-IVB-FEP Cluster, Intel Xeon 12C 2.2GHz, TH Express 2, Intel Xeon Phi 31S1P 57-core NUDT</td>
<td>3,120,000</td>
<td>33.863</td>
<td>2</td>
<td>0.580</td>
<td>1.1%</td>
</tr>
<tr>
<td>3</td>
<td>DOE/NNSA/LANL/SNL, USA</td>
<td>Trinity – Cray XC40, Intel Xeon E5-2698 v3 300160C 2.3GHz, Aries</td>
<td>979,072</td>
<td>14.137</td>
<td>7</td>
<td>0.546</td>
<td>1.8%</td>
</tr>
<tr>
<td>4</td>
<td>Swiss National Supercomputing Centre (CSCS), Switzerland</td>
<td>Piz Daint – Cray XC50, Intel Xeon E5-2690v3 12C 2.6GHz, Cray Aries, NVIDIA Tesla P100 16GB</td>
<td>361,760</td>
<td>19.590</td>
<td>3</td>
<td>0.486</td>
<td>1.9%</td>
</tr>
<tr>
<td>5</td>
<td>National Supercomputing Center in Wuxi, China</td>
<td>Sunway TaihuLight – Sunway MPP, SW26010 260C 1.45GHz, Sunway NRCPC</td>
<td>10,649,600</td>
<td>93.015</td>
<td>1</td>
<td>0.481</td>
<td>0.4%</td>
</tr>
</tbody>
</table>
Preparing Applications for Exascale

1. What are challenges?

1. What are we doing about it?
Harnessing FLOPS at Exascale

• Will an exascale machine require too much from applications?
  – Extreme parallelism
  – High computational intensity (not getting worse)
  – Sufficient work in presence of low aggregate RAM (5%)
  – Focus on weak scaling only: High machine value of $N^{1/2}$
  – Localized high bandwidth memory
  – Vectorizable with wider vectors
  – Specialized instruction mixes (FMA)
  – Sufficient instruction level parallelism (multiple issue)
  – Amdahl headroom
ECP Approach to ensure useful exascale system for science

• 25 applications projects: each project begins with a mission critical science or engineering *challenge problem*

• The challenge problem represents a capability currently beyond the reach of existing platforms.

• Must demonstrate
  – Ability to execute problem on exascale machine
  – Ability to achieve a specified Figure of Merit
The software cost of Exascale

• What changes are needed
  – To build/run code? *readiness*
  – To make efficient use of hardware? *Figure of Merit*

• Can these be expressed with current programming models?

ECP Applications – Distribution of Programming Models

<table>
<thead>
<tr>
<th>Node\Internode</th>
<th>Explicit MPI</th>
<th>MPI via Library</th>
<th>PGAS, CHARM++, etc.</th>
</tr>
</thead>
<tbody>
<tr>
<td>MPI</td>
<td>High</td>
<td>High</td>
<td>N/A</td>
</tr>
<tr>
<td>OpenMP</td>
<td>High</td>
<td>High</td>
<td>Low</td>
</tr>
<tr>
<td>CUDA</td>
<td>Medium</td>
<td>Low</td>
<td>Low</td>
</tr>
<tr>
<td>Something else</td>
<td>Low</td>
<td>Low</td>
<td>Low</td>
</tr>
</tbody>
</table>

Bottom Line: All MPI and MPI+OpenMP ubiquitous
Heavy dependence on MPI built into middleware (PetsC, Trilinos, etc)
Will we need new programming models?

• Potentially large software cost + risk to adopting new PM

• However, abstract machine model underlying both MPI and OpenMP have shortcomings, e.g.
  – Locality for OpenMP
  – Cost of synchronization for typical MPI bulk synchronous

• Good news: Standards are evolving aggressively to meet exascale needs

• Concerns remain, though
  – Can we reduce software cost with hierarchical task-based models?
  – Can we retain performance portability?
  – What role do non-traditional accelerators play?
How accelerators affect programmability

• Given performance per watt, specialized accelerators (LOC/TOC combinations) lie clearly on path to exascale

• Accelerators are heavier lift for directive-based language like OpenMP or OpenACC

• Integrating MPI with accelerators (e.g. GPUDirect on Summit)

• Low apparent software cost might be fool’s gold

• What we have seen: Current situation favors applications that follow 90/10 type rule
Programming Model Approaches

• Power void of MPI and OpenMP leading to zoo of new developments in programming models.
  – This is natural and not a bad thing, will likely coalesce at some point

• Plans include MPI+OpenMP but …
  – On node: Many project are experimenting with new approaches that aim at device portability: OCCA, KOKKOS, RAJA, OpenACC, OpenCL, Swift
  – Internode: Some projects are looking beyond MPI+X and adopting new or non-traditional approaches: Legion, UPC++, Global Arrays
Many applications depend on MPI implicitly via middleware, eg.
- Solvers: Petsc, Trilinos, Hypre
- Frameworks: Chombo (AMR), Meshlib

Major focus is to ensure project-wide that these developments lead the applications!
Rethinking algorithmic implementations

- Reduced communication/data movement
  - Sparse linear algebra, Linpack, etc.
- Much greater locality awareness
  - Likely must be exposed by programming model
- Much higher cost of global synchronization
  - Favor maxim asynchrony where physics allows
- Value to mixed precision where possible
  - Huge role in AI, harder to pin down for PDEs
- Fault resilience?
  - Likely handled outside of applications
Beyond implementations

• For applications we see hardware realities forcing new thinking beyond implementation of known algorithms
  – Adopting Monte Carlo vs. Deterministic approaches
  – Exchanging on-the-fly recomputation vs. data table lookup (e.g. neutron cross sections)
  – Moving to higher-order methods (e.g. CFD)
  – The use of ensembles vs. time-equilibrated ergodic averaging
Co-design with hardware vendors

• HPC vendors need deep engagement with applications prior to final hardware design

• *Proxy Applications* are a critical vehicle for co-design
  – ECP includes Proxy Apps Project
  – Focus on motif coverage
  – Early work with performance analysis tools and simulators

• Interest (in theory) in more complete applications.
First HACC Tests on the OLCF Early-Access System

Scope & Objectives

- **Computational Cosmology:** Modeling, simulation, and prediction for new multi-wavelength sky observations to investigate dark energy, dark matter, neutrino masses, and primordial fluctuations
- **Challenge Problem:** Meld capabilities of Lagrangian particle-based approaches with Eulerian AMR methods for a unified exascale approach to: 1) characterize dark energy, test general relativity, 2) determine neutrino masses, 3) test theory of inflation, 4) investigate dark matter
- **Main drivers:** Establish 1) scientific capability for the challenge problem, and 2) full readiness of codes for preexascale systems in Years 2 and 3

Impact

- Well prepared for the arrival of Summit in 2018 to carry out impactful HACC simulations
- With CRK-HACC we have developed the first cosmological hydrodynamics code that can run at scale on a GPU-accelerated system
- The development of these new capabilities will have a major impact for upcoming cosmological surveys

Project Accomplishment

- HACC was successfully ported to Summitdev
- The HACC port included migration of the HACC short-range solver from OpenCL to CUDA
- We demonstrated expected performance comparing to Titan and validated the new CUDA version
- We implemented CRK-HACC on Summitdev and carried out a first set of tests
Monte Carlo performance optimization for full core problems

Scope and objectives

- **Small Modular Reactor (SMR) Challenge Problems** require simulation of very large number of Monte Carlo particle histories to achieve sufficient statistical accuracy
- Current goal is to enhance computational performance based on previous profiling studies
- Additional goal is to improve generation of data libraries for windowed multipole method (WMP)
  - WMP was previously limited to select number of isotopes in nuclear data libraries

Impact

- Improved Monte Carlo particle tracking rate allows reduction in statistical errors
- WMP is now a viable route forward for production Monte Carlo solvers
- Optimization approaches provide insight into optimization strategies for other latency-bound application areas

Project accomplishment

- Realized substantial performance gains on CPU, Intel Xeon Phi, and Nvidia GPU architectures
  - 2-3x speedup across all architectures
- Developed new vector-fitting approach for generation of WMP data libraries
  - Allows processing of data for all nuclides
- Demonstrated KPP figure-of-merit projection of 20 for Summit supercomputer relative to Titan
  - Approximated using previous generation P100 GPU, actual value expected to be larger

Impact

- Improved Monte Carlo particle tracking rate allows reduction in statistical errors
- WMP is now a viable route forward for production Monte Carlo solvers
- Optimization approaches provide insight into optimization strategies for other latency-bound application areas

---

**Table:**

<table>
<thead>
<tr>
<th>Machine</th>
<th>Device Type</th>
<th>Device Rate (n/s)</th>
<th># Devices</th>
<th>Full Machine Rate (n/s)</th>
<th>FOM</th>
</tr>
</thead>
<tbody>
<tr>
<td>Titan</td>
<td>16 core CPU</td>
<td>$7.1 \times 10^8$</td>
<td>18,688</td>
<td>$1.32 \times 10^7$</td>
<td>1.0</td>
</tr>
<tr>
<td>Summit</td>
<td>1x P100 GPU</td>
<td>$9.7 \times 10^8$</td>
<td>27,600*</td>
<td>$2.68 \times 10^6$</td>
<td>20.3</td>
</tr>
</tbody>
</table>

*Based on latest data from d محمود.gov with ~6000 nodes, 6 GPUs per node

---

**Graph:**

- Accuracy of windowed multipole method relative to reference data.
- GPU MC performance on depleted fuel benchmark.
FY18-Q1: Deploy production sliding mesh capability with linear solver benchmarking

Scope & Objectives

- **ExaWind Objective:** Create a computational fluid and structural dynamics platform for exascale predictive simulations of wind farms
- **Challenge Problem:** Predictive simulation of a wind plant composed of $O(100)$ wind turbines sited over $O(100)$ km$^2$ with complex terrain
- This milestone is a necessary and critical step in moving towards MW-scale-turbine simulations
  - Establishes baseline performance for a fully resolved sub-MW-scale turbine in an operating configuration

Impact

- The new sliding mesh capability provides a pathway for efficient simulation of rotating meshes in wind turbine simulations
- Simulating a 1.3B element mesh is a milepost on the pathway to the extreme mesh sizes required for MW-scale-turbine simulations
- Coupling of Nalu with Hypre and MueLu provides insight into, and a comparison platform for, two fundamentally different AMG approaches (classic and smoothed aggregation); highlighted areas for future work

Project Accomplishment

- Deployed and verified a design-order hybrid CVFEM/DG sliding-mesh interface for wind turbine simulations
- Completed transition to Kokkos for interior topology matrix contributions for wind applications
- Coupled the Nalu solver with the Hypre AMG preconditioner and the TIOGA overset library
- Under the ECP ALCC ExaWind allocation on Cori, established baseline timing results for a fully resolved sub-MW-scale turbine
  - Detailed timing breakdown for MueLu/Belos and Hypre solvers
- Successfully simulated sub-MW-scale fully resolved turbine with 1.3B elements

Simulation results for a fully-resolved sub-MW-scale turbine for which the rotor resides in an embedded, rotating “disk” of fluid that is coupled to the surrounding fluid via a sliding-mesh interface. Shown are velocity shadings from the upwind (left) and downwind (right) perspectives.
PeleC Embedded Boundary Capability

Scope & Objectives

• The goal of this project is to provide a simulation capability for first-principles (DNS) and near-first principles (DNS/LES hybrids) simulations of turbulence-chemistry interactions in conditions relevant to practical combustion devices, including turbulence, mixing, spray vaporization, low-temperature ignition, and flame propagation.

Impact

• Accurate simulation of combustion at high pressure such as conditions in a diesel engine requires modeling non-ideal fluid behavior, particularly for large hydrocarbons
• Four year demonstration problem is a single sector of a gas turbine combustion; the geometry of the flame holder is needs to be captured to generate recirculation zones that anchor the flame.

Project Accomplishment and Next Steps

• Cartesian cut cell implementation in PeleC allows simulation of complex geometry using explicit diffusion treatment and method of lines approach to hyperbolic treatment
• Capability demonstration is ~30x faster than start of project baseline and 5x slower than proof of concept created by AMReX and tailored for gamma-law gas dynamics
• Calculation of diffusive and advective fluxes needs to be coordinated to improve computational throughput and reduce memory usage
• Performance engineering of initial code for more general cases (multivalued, vector potential) is next major step
Summary

• Major challenge for mission-critical HPC applications to get proportional performance moving toward exascale

• From application perspective high risk in being passive
  – Engage now with HPC vendors
  – Be aware of emerging technologies, particularly new ideas for programmability
  – Drive new science/engineering opportunities and numerical approaches by key features of hardware