High Performance Computations Enable Scientific Discoveries - Adapting Computational Physics Codes to New Hardware

[Extended Abstract]

James Amundson
Fermi National Accelerator Laboratory
amundson@fnal.gov

Philippe Canal
Fermi National Accelerator Laboratory
pcanal@fnal.gov

Don Holmgren
Fermi National Accelerator Laboratory
djholm@fnal.gov

Chris Green
Fermi National Accelerator Laboratory
greenc@fnal.gov

James Kowalkowski
Fermi National Accelerator Laboratory
jbk@fnal.gov

Qiming Lu
Fermi National Accelerator Laboratory
qlu@fnal.gov

Marc Paterno
Fermi National Accelerator Laboratory
paterno@fnal.gov

Saba Sehrish
Fermi National Accelerator Laboratory
ssehrish@fnal.gov

ABSTRACT

Physics research as we know it today would not be possible without computers. Evaluating fundamental physical theories at the smallest and largest physical scales requires the production of enormous quantities of numerical data. Designing, building, and commissioning highly complex accelerators and detectors requires the simulation of enormous event samples similar to those collected for an experiment. Breakthroughs in physics research depend on the use of leading-edge computing technologies. The Fermilab Scientific Computing Division together with university and laboratory collaborators is advancing the use of state-of-the-art computing platforms that include GP-GPUs and Blue Gene to enable scientific discoveries by evolving the software of four application toolkits. Here we show the purpose, goals, and progress of this work within Lattice QCD calculations (QUDA), computational cosmology (ART - Adaptive Refinement Tree), detector simulation (Geant4), and accelerator modeling (Synergia).

Lattice QCD: Ordinary matter consists of protons and neutrons (nucleons), each of which contains three point-like particles called quarks. The strong force, one of the four fundamental forces of nature, binds quarks to each other. It is transmitted between quarks by particles called gluons. The theory describing the interactions of quarks and gluons is called Quantum Chromodynamics (QCD). Theorists must resort to large-scale numerical simulations to solve QCD for many of the most important Standard Model problems. The technique is called Lattice QCD (LQCD). Lattice QCD codes spend much of their time inverting very large very sparse matrices. For example, a $48 \times 48 \times 48 \times 144$ problem, typical for current simulations, has a complex matrix of size 47.8 million $\times$ 47.8 million. The matrix has 1.15 billion non-zero elements (about one in every 2 million). Individual LQCD calculations require many TFlop/sec-yrs of computations. They can only be achieved by using large-scale parallel machines. At Fermilab, LQCD simulations run on parallel computers, such as Infiniband-coupled clusters. Individual jobs use hundreds to thousands of processors. On these clusters, the software relies on message passing using MPI for coordinating the cooperative operations of the processors. Multicore processors on clusters have been used to great advantage by LQCD. Our results show that using 8 cores per node, 90% efficiency can be achieved. QUDA is a software package for carrying out the time-consuming components of an LQCD application on the NVIDIA CUDA platform. The QUDA library provides optimized implementations of a number of different discretizations of the continuum QCD fermion operator as well as a range of conjugate-gradient-based iterative solvers for these fermion actions. Our QUDA results on a cluster with NVIDIA M2050 GPGPUs show that performance improves with the number of cores up to 16, which is normally the number of cores per node on many top 500 supercomputers.

Computational Cosmology (ART): Large-scale surveys of the sky are collecting massive amounts of data that hold the key to a deeper understanding of the Universe. High performance computing is a powerful tool of discovery in extracting insights and making precision predictions for these observations. The Adaptive Refinement Tree (ART) code is a high-performance cosmology code. It is one of the first to resolve dark matter substructures within halos and one of the key tools in first discussing the 'cusp problem' and the 'missing satellite problem' of cold dark matter models. The ART code allows for modeling a wide range of physical processes. Specifically, the current version of the code includes the following physical ingredients (in addition to gravity, dark matter, and gas dynamics): (i) detailed atomic physics of the cosmic plasma, including a novel method for modeling radiative cooling of the gas; gas cooling functions implemented in ART are the most accurate of all existing cosmological codes; (ii) the effects of cosmic radiation on the gas; and (iii) formation of stars and their feedback on the cosmic gas. Individual hydro simulations running in ART consume between 2M and 3M CPU-hours. For the simulations to be practical, the code needs to scale to at least 50K cores, and scaling to 200K cores is desirable. The scaling is achieved by fusing a small number of OpenMP threads into massively parallelized MPI tasks. The 3-D computational domain is decomposed using a space-filling curve (SFC). Fermilab has been working to improve the multithreading and multiprocessing scaling of the hybrid MPI and OpenMP ART program. Our goal is to achieve scaling and performance on low-latency-interconnect clusters, our initial studies shows a factor of 1.5 performance improvement.

Detector Simulation (Geant4): Detectors measure properties of particles produced by fundamental interactions between elementary particles that cannot be directly observed. Geant4 is a toolkit for simulating the passage of particles through matter. It tracks particles through a geometry composed of different materials, records any particle interactions such as electrons and nuclei encountered, and handles the creation of other new particles during these interactions. The current implementation of Geant4 contains many features that hinder our ability to make use of modern parallel architectures, for example Geant4 heavy reliance on the object-oriented features of C++ for developing class hierarchies and its serial design. As experiments evolve, the demands on both the amount of simulated data and precision of the simulations will grow. Geant4 must evolve in order to both take advantage of the emerging computing technologies and to be able to meet the future simulation demands in a cost effective way. At Fermilab, the efforts to explore the completely re-engineering of Geant4 framework in collaboration with DOE's ASCR program and CERN computing team include adding track-level parallelism, vectorizing track dispatcher, coalescing memory access, and work balancing between CPU and GPU. The initial results of these particle propagation studies show ≈ 20 times runtime improvement using AMD+M2090, and ≈ 18 times improvement using Intel+K20M.

Accelerator Modeling (Synergia): Particle accelerators are among the most complex and expensive tools ever built for scientific discovery. Accelerators at the highest energies and intensities require detailed understanding of intensity-dependent accelerator physics phenomena. Synergia is an advanced particle-in-cell accelerator simulation package for intensity-dependent effects. It is being developed at Fermilab as a part of the ComPASS SciDAC project. Synergia scales from desktops to supercomputers. In a typical Synergia simulation a bunch of $10^6 - 10^8$ particles needs to be propagated for $10^5 - 10^7$ simulation steps, each of which requires a field solve on a grid whose typical size is $64\times64\times128$. Simulations may utilize a single bunch or a train of up to a thousand bunches. The largest Synergia simulation run was done on ALCF Intrepid BG/P using $131,072$ cores to simulate over 10^{13} particles in 1024 bunches with excellent weak scaling in the number of bunches. At Fermilab, Synergia is being optimized for multi-core and GPU architectures, and we achieve speedup of 40 using GPUs.

Categories and Subject Descriptors

J [**Computer Applications**]: Miscellaneous; J.2 [**Physical Science and Engineering**]: [Cosmology, Physics]

General Terms

Applications, applications using multicore and/or GPUs, scientific applications