Multi-core Optimizations for Synergia and ART

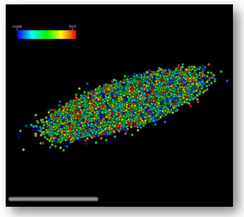
Qiming Lu[1], James Amundson[2], Nick Gnedin[3] | [1,2] Scientific Computing Division, [3] Particle Physics Division, Fermilab | Batavia, IL 60510

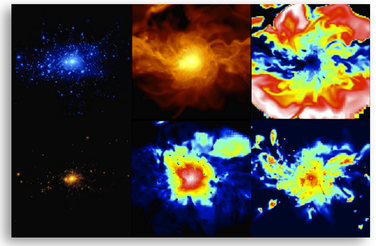
Contacts: ([1]qlu, [2]amundson, [3]gnedin)@fnal.gov

Abstract

- Optimized the performance and scaling of **Synergia** and **ART** for multi-socket multi-core architectures including BlueGene/Q and GPU
- Multiple hybridization and optimization options, including communication avoidance, interchangeable multi-threading kernel with OpenMP or CUDA, customized FFT, etc., each demonstrating much better scaling behavior than the pre-optimization code
- Extend strong scaling and peak performance by at least a factor of 2. We expect different optimization schemes to be optimal on different architectures.
- Further tailor the code for BG/Q with optimized communication divider, redundant field solver, and FFT methods. The final code of Synergia scales up to 128K cores with over 90% efficiency running on Mira (BG/Q at Argonne)

- **Synergia** is a beam dynamics simulation code capable of modeling single-particle and multi-particle dynamics for advanced accelerator simulations
  - Large scale parallel space charge Particle-In-Cell method with external fields and beam-beam interactions on regular grids

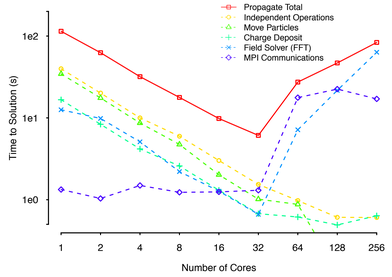




- **A**daptive **R**efinement **T**ree (**ART**) is a high-performance cosmology code and one of the first to resolve dark matter substructures within halos
  - Multiple physics processes including hydro (PDEs), particle (PIC), gravity, radiation transfer, etc. using adaptive mesh refinement method

Problems

- Contradiction of small grid sizes (limited scalability) to the need of very large particle numbers (~100 million – 1 billion) in the PIC simulation
- The presence of the external field has caused the particles to move rapidly between steps. Moving particles beyond nearest neighbors of the process yields as much as $O(n^2)$ message passing
- Extremely in-balanced computations due to the on-demand and dynamic mesh refinements



Poor scaling of pre-optimized Synergia due to spiking communication costs and in-balanced computations

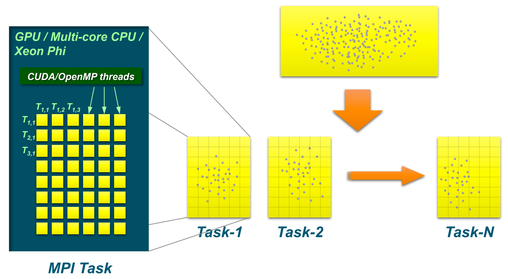
Optimization Approaches

Hybrid = High-level MPI + Thread Parallelism at Finer Granularity

- For PIC, use fixed domain and particle partitioning (direct Lagrangian) for MPI
- For AMR, use space-filling curve (SFC) at MPI level for load balancing

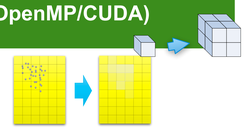
Inter-changeable computation kernels for different hardware architectures, or heterogeneous computations

Reduction in both collective and point-to-point communication costs due to (a) lesser MPI ranks, and (b) use of shared memory within an MPI rank. Thus, scaling has been largely improved



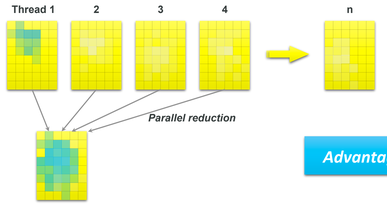
Parallel Charge / Mass Deposition Using Threads (OpenMP/CUDA)

- One macro particle contributes to multiple mesh grids
- Collaborative updating in shared memory needs proper ***synchronization*** / ***critical region protection***
- Due to enormous number of particles, collision-free deposition is preferred



Distributed Deposition

- Each thread duplicates the spatial domain in its thread-local storage
- Parallel reduction after the deposition done locally in each thread



Limitations

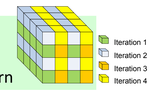
High memory usage
Limited thread concurrency
Reduction overheads increases with number of threads

Advantages

Collision-free
Minimum overheads at low thread counts

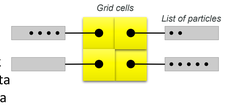
Interleaved Deposition

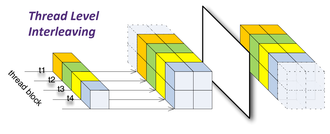
- Sort particles into their corresponding cells at each step
- Deposit particles based on color-coded cells in an interleaved pattern



Parallel bucket sort (concurrency = # of cells)

Indexed list for quick accessing particle data and reduced raw data movements



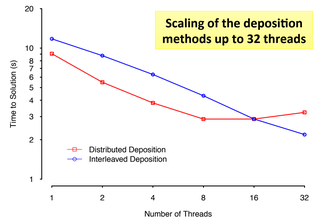


Non-trivial overheads of sorting particles when thread counts are low

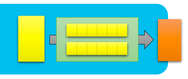
Limitations

Advantages

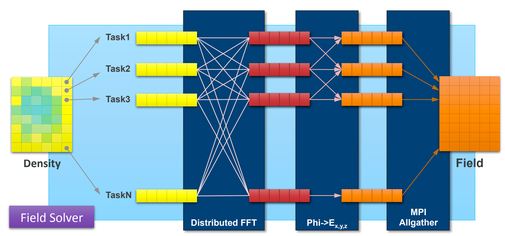
Collision-free
High concurrency, and very good scalability
Scalable sorting overheads
Fixed memory usage
Better data locality at moving particles

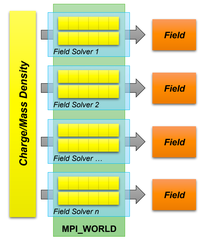


Redundant Computation (Field Solver)



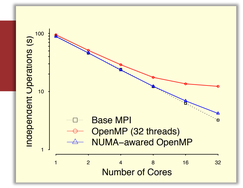
- One complete field solver per MPI sub-group to avoid global communications
- Scalability is no longer limited by the domain sizes and partitioning choices
- Meanwhile, avoid negative scaling of the field solver for large MPI groups





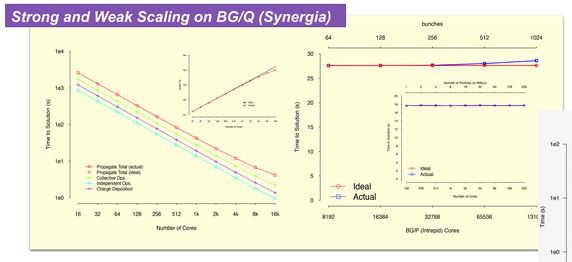
Other Optimization Approaches

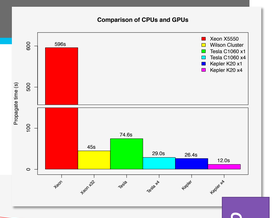
- Zero-copy FFT implementation using MPI extended types
- NUMA-awareness on multi-socket multi-core machines
- BlueGene/Q QPX units and cache prefetching, etc.

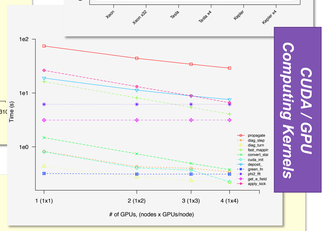


Scaling & Performances

Strong and Weak Scaling on BG/Q (Synergia)

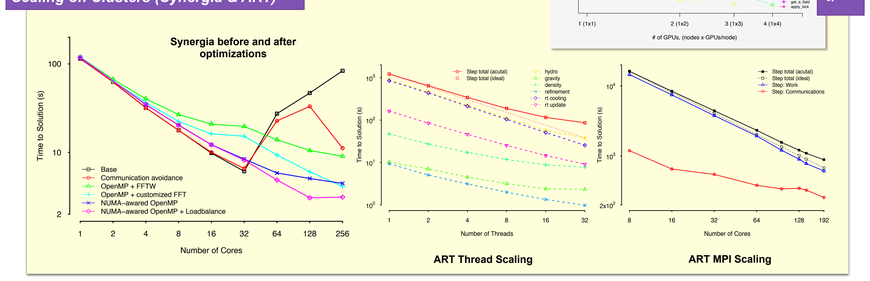






CUDA / GPU Computing Kernels

Scaling on Clusters (Synergia & ART)



- Significantly improved the scaling and performance of both Synergia and ART. Peak performances have been improved by a factor of 2.3 (Synergia) and 1.5 (ART)
- Both have reached > 85% multi-thread efficiency at 8 threads, and around 60-80% (depends on simulation problems) at 32 threads on multi-socket multi-core hardware architectures. The delay was mostly caused by non-parallelizable communications and serial algorithms
- Ideal weak scaling for the number of particles up to 256 million in PIC simulations
- Synergia is able to utilize 131,072 cores with over 90% scaling efficiency for multi-bunch simulations