Particle-in-Cell Fusion Energy Turbulence Simulations at Extreme Scale

William M. Tang

Princeton University/Princeton Plasma Physics Laboratory (PPPL)
Princeton, NJ  USA

International Exascale Software Co-Design Workshop (Co-Design 2013)

Guelin, China

October 29-31, 2013
Motivation for Investment in Computing @ Extreme Scale: Applications Drivers

- Scientific & technological challenges:
  - **Energy:** new fuels and reactors (fission & fusion)
  - **Security:** stewardship without nuclear tests
  - **Environment:** Carbon sequestration alternatives & regional climate impacts

- Broader applications of extreme computing and big data-driven discovery will have major impact on applied & fundamental science
  - **Renewable energy and energy storage**
  - **Prediction and control of materials in extreme environments**
  - **Understanding dark energy and dark matter**
  - **Clean and efficient combustion in advanced engines**

Rapid International Advances in HPC

Chart shows most capable system for each year in **TOP500**

Teraflop = $10^{12}$ floating point operations per second

**Pre-eminence in 21st Century R&D ➔ Leading role in harvesting “Big Data & Extreme Computing (BDEC)” opportunities to accelerate progress in key application areas**
Extreme Scale HPC can dramatically benefit many scientific domain applications (including FES) and industry

• **Practical Considerations**: [achieving “buy-in” from general scientific community]
  - Need to distinguish between **voracious** *(more of same - just bigger & faster)* vs. **transformational** *(achievement of major new levels of scientific understanding)*
  - Need to improve significantly on experimental validation together with verification & uncertainty quantification to enhance realistic predictive capability

• **Associated Extreme Scale Computing Challenges:**
  - **Hardware complexity**: Heterogenous multicore (e.g., gpu+cpu, xeon-phi+cpu), power management, memory, communications, resiliency, storage, …
  - **Software challenges**: Operating systems, I/O and file systems, and coding, algorithmic, & solver needs in the face of increased computer architecture complexity … must deal with local concurrency e.g., MPI & OpenMP + threads, CUDA, etc. → **rewriting code focused on data movement over arithmetic**!

• References:
Fusion Energy: *Burning plasmas are self-heated and self-organized systems*

Deuterium-Tritium Fusion Reaction

- Deuterium (D)
- Tritium (T)

*Fusion Reaction*
- Alpha Particle (He$^+$) 3.5 MeV
- Fast Neutron (n$^0$) 14.1 MeV

*Energy Multiplication*
*About 450:1*

\[ D^+ + T^+ \rightarrow ^4\text{He}^{++} \ (3.5 \text{ MeV}) + n^0 \ (14.1 \text{ MeV}) \]
Fusion: an Attractive Energy Source

- Abundant fuel, available to all nations
  - Deuterium and lithium easily available for millions of years
- Environmental advantages
  - No carbon emissions, short-lived radioactivity
- Cannot “blow up or melt down,” resistant to terrorist attack
  - Less than a minute’s worth of fuel in the chamber
- Low risk of nuclear materials proliferation
  - No fissile materials required
- Compact relative to solar, wind and biomass
  - Modest land usage
- Not subject to daily, seasonal or regional weather variation; no requirement for local CO$_2$ sequestration
  - Not limited in its application by need for large-scale energy storage nor for long-distance energy transmission
- Fusion is complementary to other attractive energy sources
ITER Goal: Demonstration of the Scientific and Technological Feasibility of Fusion Power

- **ITER** is an ~$20B facility located in France & involving 7 governments representing over half of world’s population
  - dramatic next-step for Magnetic Fusion Energy (MFE) producing a sustained burning plasma
    -- Today: 10 MW(th) for 1 second with gain ~1
    -- ITER: 500 MW(th) for >400 seconds with gain >10

- “**DEMO**” will be demonstration fusion reactor after ITER
  -- 2500 MW(th) continuous with gain >25, in a device of similar size and field as ITER

- Ongoing R&D programs worldwide [experiments, theory, computation, and technology] essential to provide growing knowledge base for ITER operation targeted for ~ 2020

  ➔ Realistic HPC-enabled simulations required to cost-effectively plan, “steer,” & harvest key information from expensive (~$1M/long-pulse) ITER shots
Microturbulence in Fusion Plasmas – Mission Importance: Fusion reactor size & cost determined by balance between loss processes & self-heating rates

- “Scientific Discovery” - Transition to favorable scaling of confinement produced in simulations for ITER-size plasmas
  - $a / \rho_i = 400$ (JET, largest present lab experiment) through
  - $a / \rho_i = 1000$ (ITER, ignition experiment)

- Multi-TF simulations using GTC global PIC code [Z. Lin, et al, 2002] deployed a billion particles, 125M spatial grid points; 7000 time steps $\rightarrow$ 1st ITER-scale simulation with ion gyroradius resolution

- BUT, compelling understanding of plasma size scaling demands higher physics fidelity requiring much greater computational resources + new algorithms & modern diagnostics for VV&UQ

$\rightarrow$ Excellent Scalability of Global PIC Codes on modern HPC platforms enables much greater resolution/physics fidelity to improve understanding

$\rightarrow$ BUT - further improvements for efficient usage of current LCF’s demands code re-write featuring modern CS/AM methods addressing locality, low-memory-per-core, …
Particle Simulation of the Boltzmann-Maxwell System

- The Boltzmann equation (Nonlinear PDE in Lagrangian coordinates):
  \[
  \frac{dF}{dt} = \frac{\partial F}{\partial t} + v \cdot \frac{\partial F}{\partial x} + \left( E + \frac{1}{c} v \times B \right) \cdot \frac{\partial F}{\partial v} = C(F).
  \]

- “Particle Pushing” (Linear ODE’s)
  \[
  \frac{dx_j}{dt} = v_j, \quad \frac{dv_j}{dt} = \frac{q}{m} \left( E + \frac{1}{c} v_j \times B \right)_{x_j}.
  \]

- Klimontovich-Dupree representation,
  \[
  F = \sum_{j=1}^{N} \delta(x - x_j) \delta(v - v_j),
  \]

- Poisson’s Equation: (Linear PDE in Eulerian coordinates (lab frame))
  \[
  \nabla^2 \phi = -4\pi \sum_{\alpha} q_{\alpha} \sum_{j=1}^{N} \delta(x - x_{\alpha j})
  \]

- Ampere’s Law and Faraday’s Law  [Linear PDE’s in Eulerian coordinates (lab frame)]
Basic Particle-in-Cell Method

- Charged particles sample distribution function
- Interactions occur on a grid with the forces determined by gradient of electrostatic potential (calculated from deposited charges)
- *Grid resolution dictated by Debye length (“finite-sized” particles) up to gyro-radius scale*

**Specific PIC Operations:**
- “SCATTER”, or deposit, charges as “nearest neighbors” on the grid
- Solve Poisson Equation for potential
- “GATHER” forces (gradient of potential) on each particle
- Move particles (*PUSH*)
- Repeat…
### Performance Speed Up Comparison Results (IBM BG-Q vs. BG-P)

<table>
<thead>
<tr>
<th>M0180 ppc =100</th>
<th>Our test</th>
<th>ALCF</th>
<th>IBM</th>
</tr>
</thead>
<tbody>
<tr>
<td>Speed up per node (Q/P ratio)</td>
<td>10.7</td>
<td>10.7</td>
<td>11.2</td>
</tr>
</tbody>
</table>

Speed up per node **comparison with ALCF and IBM results** for “M0180” problem size (i.e., 180 grid-points in radial direction) using “GTC-Princeton” code – Fortran-90 version

→ Test Case for phase-space resolution with particles/cell (ppc) =100 for 100 time-steps
→ “Time to Solution” improvement from **BG-Q hardware**
  * 4X (core) 2X (frequency) 2X (SIMD) = 16 (“theoretical”)
Features of new C-Version of “GTC-Princeton” Code

• “GTCP-C” code based on a greatly optimized version of the C-version of the original GTC code (non-optimized version introduced at SC 2011)
  • “C” (instead of usual Fortran) to rapidly incorporate CS community advances in multi-threading for low-memory-per-core systems

• Key additional level of domain decomposition introduced into radial dimension ➔ essential for efficiently carrying out simulations on large-scale plasmas such as ITER
  • Alleviates grid memory requirement issue for large size plasma simulation
  • Improves locality

• Multiple levels of parallelism
  • 2D domain decomposition (toroidal and radial dimensions)
  • Particle decomposition in each domain
  • Multi-threaded, shared memory parallelism implemented with loop-level OpenMP directives

• Improvements over GTCP-FORTRAN code
  • Remove PETSc library for carrying out Poisson field solve
    • Significantly improves code portability to various LCF’s
    • Introduces loop level parallelism in the Poisson solve

➔ Overall Software Improvement gives another 50% gain in “Time to Solution” beyond factor of 10 in moving from IBM BG-P (“Intrepid”) to BG-Q (“Mira”) @ALCF
Reference Fortran Version of GTC-P
(includes radial domain decomposition)

Run on BG-Q (16 cores/node)

Wall-clock time per step (s)

Charge | Push | Shift | Poisson | Field | Smooth

Nodes ➔

A 128
B 512
C 2048
D 8192 (ITER size)
Optimized C Version of GTC-Princeton

Run on BG-Q (16 cores/node)

- Nodes ➔ 128, 512, 2048, 8192 (ITER size)

Wall-clock time per step (s)
Strong Scaling Study of GTC-P (C-version) on Mira

**D (ITER-scale) Problem for 100 Time Steps**

<table>
<thead>
<tr>
<th>Radial partitions</th>
<th>Time on Mira</th>
<th>Ideal</th>
<th>“Eff” efficiency</th>
</tr>
</thead>
<tbody>
<tr>
<td>32</td>
<td>527.5</td>
<td>527.5</td>
<td>100%</td>
</tr>
<tr>
<td>64</td>
<td>265.1</td>
<td>263.8</td>
<td>99%</td>
</tr>
<tr>
<td>128</td>
<td>137.1</td>
<td>131.9</td>
<td>96%</td>
</tr>
<tr>
<td>256</td>
<td>72.6</td>
<td>65.9</td>
<td>91%</td>
</tr>
<tr>
<td>512</td>
<td>41.4</td>
<td>33</td>
<td>80%</td>
</tr>
</tbody>
</table>

- 64-way toroidal partitioning on all numerical experiments
- BG/Q (Mira) system → *use 4 processes/node, 16 threads/process*
New Physics Insights on Fusion Confinement Scaling Enabled by Computing at Extreme Scale

DOE INCITE Project on “Kinetic Simulations of Fusion Energy Dynamics @ Extreme Scale”

Objectives

• Develop modern software capable of using low memory supercomputers to carry out high physics fidelity first principles simulations of multiscale tokamak plasmas for magnetic fusion energy (MFE)

• Fusion Physics & HPC Challenges:
  → Key decade-long MFE estimates of confinement scaling with device size (“Bohm to Gyro-bohm” trend) need much higher resolution to be realistic/reliable.
  → Major algorithmic advances needed for MFE global PIC codes to effectively engage computing at extreme scale.

Impact

• Understanding the physics governing MFE confinement scaling → one of highest priority research areas for success of next-step burning plasma experiments (e.g., ITER)

• GTC-Princeton (“GTC-P”) makes efficient use of DoE’s LCF’s to carry out ITER scale simulations with unprecedented resolution in phase-space & time.

Accomplishments

• Production-run simulations of turbulence dynamics governing confinement physics for large-scale MFE plasmas (e.g., ITER) have been successfully carried out for the first time with very high phase-space resolution and long temporal duration.

• Co-design interdisciplinary research has now produced “GTC-P” – a modern HPC fusion energy science code that enables efficient use of multi-petascale capabilities on world-class CPU systems such as the IBM BG-Q “Mira” @ ALCF & “Sequoia” @ LLNL to deliver important new scientific insights.
**BG-Q Performance: Weak Scaling Results**

- *Mira @ ANL & Sequoia @ LLNL*
- *C-Version of GTC-P Global GK PIC Code: 200 ppc resolution*
- *Plasma system size increases from A to D with D being ITER*

- **Mira**
- **B200**
- **C200**
- **Sequoia* (16.3 PF)**

- Excellent scaling to all 1,572,864 processor cores (capable of pushing over 130B particles)
- Hybrid MPI+OpenMP in “GTC-P C” took full advantage of highly multi-threaded nodes and large scalable interconnect in BG-Q

*Bei Wang (Princeton U.) & S. Ethier (PPPL)
K-Computer Performance: Weak Scaling Results

- Fujitsu-K Computer @ RIKEN AICS, Kobe, Japan
- C-Version of GTC-P Global GK PIC Code: 200 ppc resolution
- Plasma system size increases from A to D with D being ITER

![Graph showing weak scaling results for GTC-P on K-Computer. The graph plots compute power (millions number of particles per second per step) against the number of nodes. Points A, B, C, and D indicate different system sizes, with D being the largest and representing ITER.](image-url)
Particle Resolution (ppc) Convergence Study
GTC-P C Code for ITER (D-size) Case on BG-Q

Time History of Thermal Diffusivity from ITG Instability
Recent High Resolution Ion Transport Scaling Results enabled by “Mira” at ALCF
[vertical axis represents transport level and horizontal axis the plasma size with ITER at 1000]

New Trends: “rollover” significantly more gradual than established earlier in much lower resolution, shorter duration studies with magnitude of transport now reduced by ~ 2
Summary: Programming Model Challenges in Moving toward Extreme Scales

- **Locality**: Need to improve data locality (e.g., by sorting particles according to their positions on grid)
  -- due to physical limitations, moving data between, and even within, modern microchips is more time-consuming than performing computations!

- **Latency**: Further exploration of highly multi-threaded algorithms to address memory latency motivated, e.g., by positive results from present studies

- **Flops vs. Memory**: Need to utilize Flops (cheap) to better utilize Memory (limited & expensive to access)

- **Advanced Architectures**: Need more “demo-apps” that deploy innovative algorithms within modern science codes on low memory per core architectures – (e.g, BG/Q, Fujitsu-K, Titan, Tianhe-1A, TH-2, .....
  -- multi-threading within nodes, maximizes locality while minimizing communications
  -- large future simulations (PIC ➔ very high-resolution (ppc) production runs for long-duration in large-plasma-size scaling studies)

**Achievements & Progress:**

1. Excellent performance scaling & “time-to-solution” achieved with C version of “GTC-Princeton” code on BG/Q (Mira & Sequoia), on Fujitsu-K Computer (Japan);
2. Significant progress on GPU-CPU Titan (OLCF); &
3. New studies have begun on Intel-MIC systems (Stampede and TH-2)
Extra Slides
“GTC-GPU” Code \((K. \ Ibrahim, \ LBNL; \ B. \ Wang, \ Princeton \ U; \ et \ al.)\)


- Physics content in new GTC-GPU code is same as in all other versions of “GTC-Princeton”
- Challenge: massive fine-grained parallelism and explicit memory transfers between multiple memory spaces within a compute node
- Approach:
  -- particle-related computational phases on GPU’s while grid-related phases stay on CPU’s
  -- particle operations kept on GPU’s where explicit memory transfers between multiple memory spaces are minimized in order to maximize performance
  -- Cuda, & OpenMP used within a node and MPI between nodes
  -- CPU-GPU hybrid version uses shared memory to achieve coalesced global memory access (for improved memory bandwidth); good scaling demonstrated on Titan-Dev
  -- dealt with Amdhal’s law on speedup by parallelizing using atomics – [charge deposition phase has iterations with loop-carried dependency] Findings: \(\Rightarrow\) Memory locality improves performance of most routines but degrades performance for atomics because of access conflicts

\(\Rightarrow\) Conflicting requirements for locality and conflict avoidance make optimizing performance on GPUs both interesting and challenging

* Transaction Memory futures (Intel-Xeon Phi)?
Weak Scaling Study of GTC-P on Mira & Titan

- GTCP-C Titan points beyond 8192 nodes (dashed line) are extrapolated
- 32768 nodes represents 2/3 of Mira BG-Q system