Exascale Challenges for Programming Models in Fusion Energy Sciences

William M. Tang

Princeton University, Plasma Physics Laboratory
Princeton, NJ USA

Workshop on Programming Models for Exascale Simulations and Modeling Applications (PPME)

Portland, Oregon

August 14-15, 2012

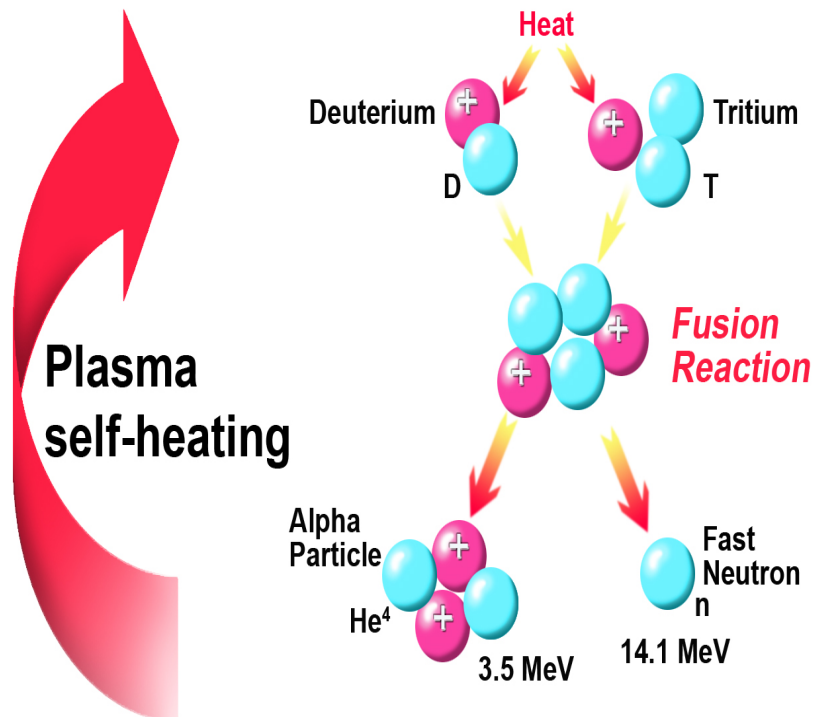
Fusion: an Attractive Energy Source

- 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 minute's worth of fuel in 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 by need for large-scale energy storage nor for long-distance energy transmission
- *Fusion is complementary to other attractive energy sources*

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

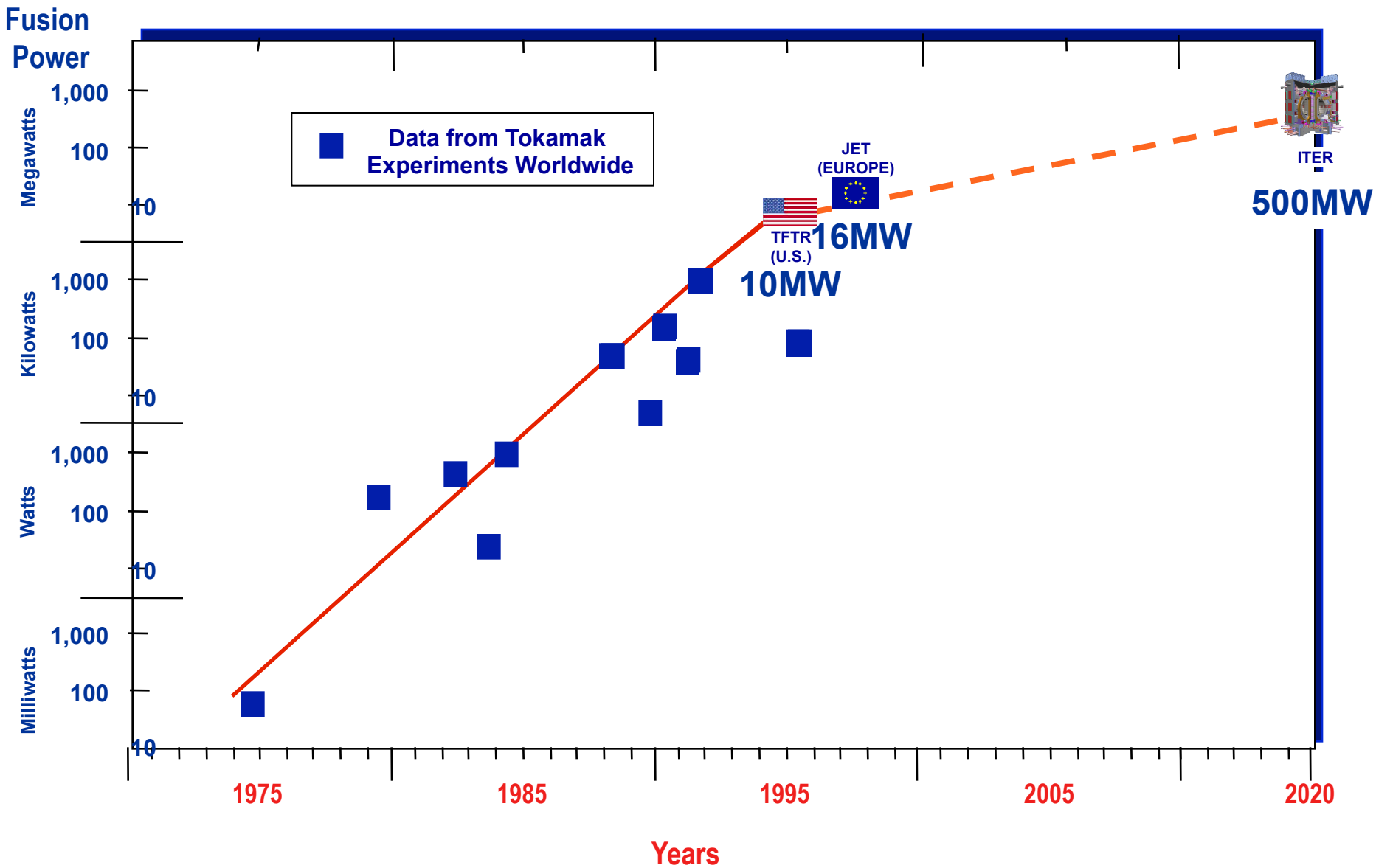
Deuterium-Tritium Fusion Reaction



Energy Multiplication About 450:1

$$D^+ + T^+ \rightarrow {}^4He^{++}\ (3.5\ MeV) + n^0\ (14.1\ MeV)$$

Progress in Magnetic Fusion Energy (MFE) Research



ITER Goal: *Demonstration of the Scientific and Technological Feasibility of Fusion Power*

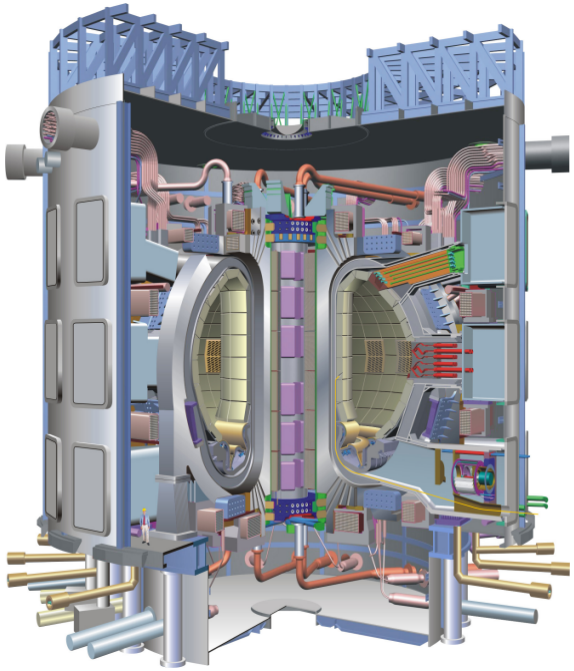
- ITER *is an ~$20B facility located in France & involving 7 governments representing over half of world's population*
  → *dramatic next-step for Magnetic Fusion Energy (MFE) producing a sustained burning plasma*
  -- Today: 10 MW(th) for 1 second with gain ~1
  -- ITER: 500 MW(th) for >400 seconds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

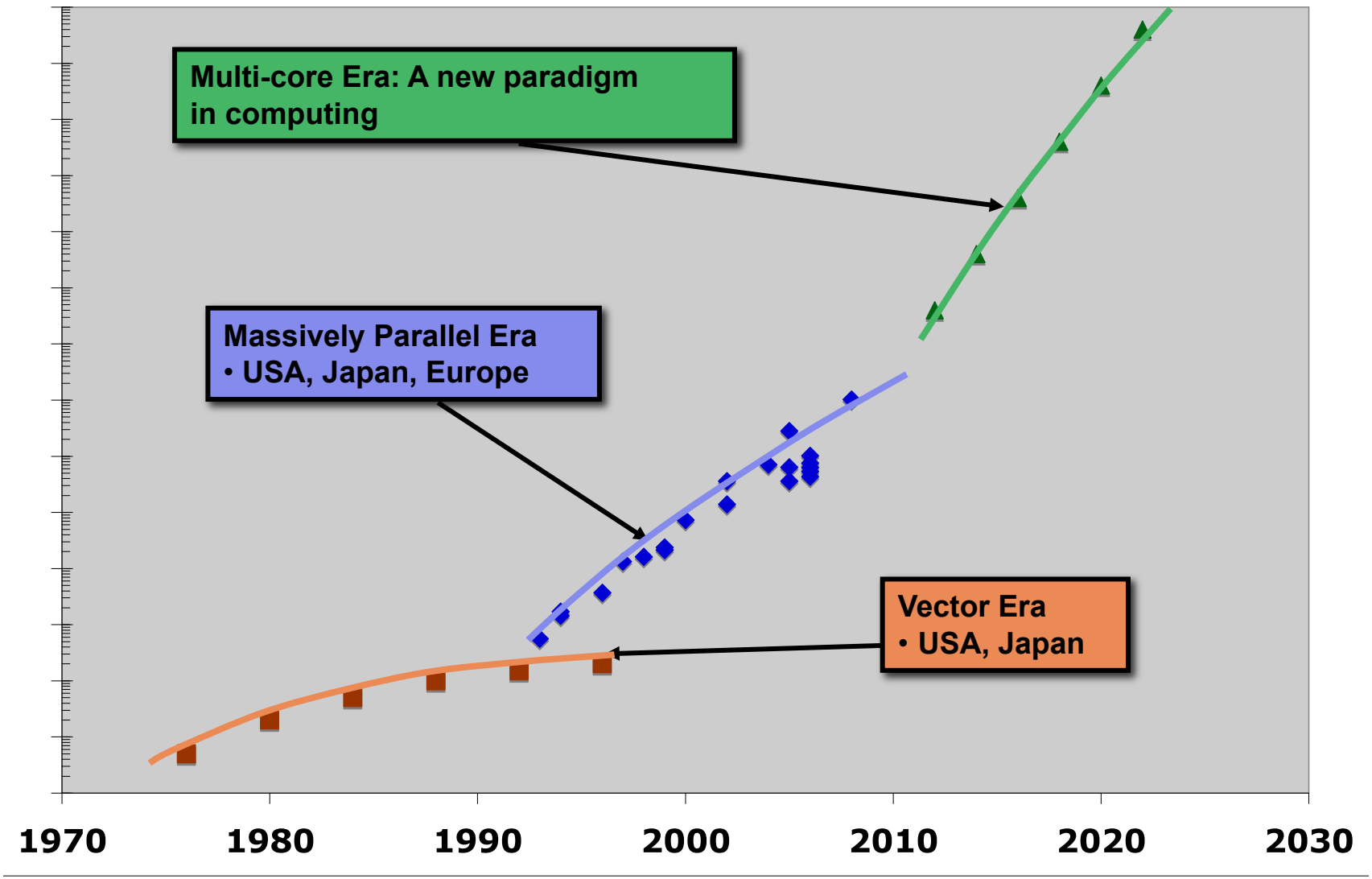
→ *Realistic HPC-enabled simulations required to cost-effectively plan, "steer," & harvest key information from expensive (~$1M/long-pulse) ITER shots*

ITER



FES Needs to be Prepared to Exploit *Local Concurrency* to Take Advantage of Most Powerful Supercomputing Systems in 21st Century

(e.g., U.S.'s Blue-Gene-Q & Titan, Japan's Fujitsu-K, China's Tianhe-1A, ….)



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 => OLCF's "Titan"), power management, memory, communications, 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 (MPI + threads, CUDA, etc. → rewriting code focused on data movement over arithmetic)

- References:

W. Tang, D. Keyes, et al., ***"Scientific Grand Challenges: Fusion Energy Sciences and the Role of Computing at the Extreme Scale,"*** PNNL-19404, 212pp (March, 2009).
http://www.er.doe.gov/ascr/ProgramDocuments/Docs/FusionReport.pdf

R. Rosner, et al., ***"Opportunities & Challenges of Exascale Computing"*** – DoE Advanced Scientific Computing Advisory Committee Report (November, 2010).

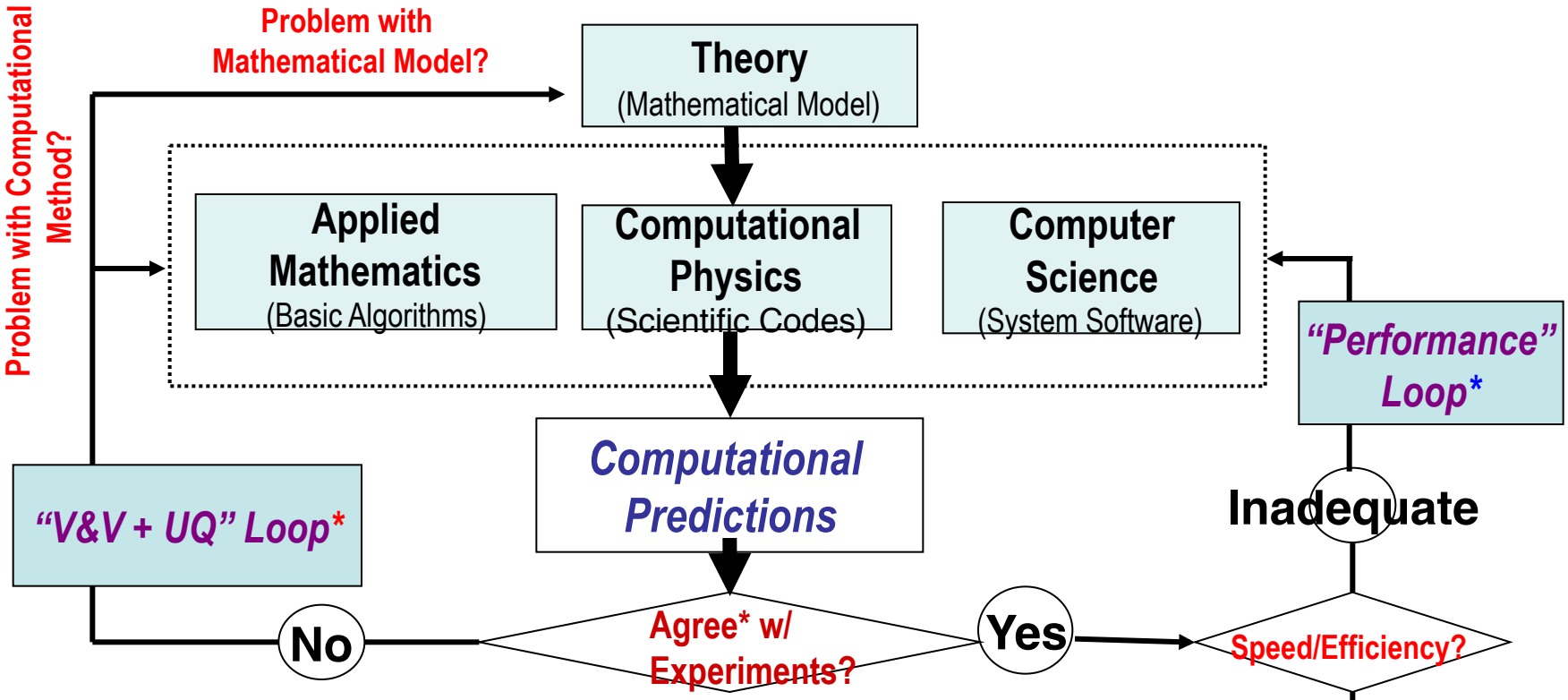
G8 Exascale Software Projects (2011- 2014)

- "Enabling Climate Simulation @ Extreme Scale" (ECS) – US, Japan, France, Canada, Spain
- "Climate Analytics on Distributed Exascale Data Archives" (ExArch) UK, US, France, Germany, Canada, Italy
- "Icosahedral-Grid Models for Exascale Earth System Simulations" (ICOMEX) – Japan, UK, France, Germany, Russia

→ "Nuclear Fusion Simulations @ Exascale" (NuFuSE) – UK, US, Germany, Japan, France, Russia

- "Modeling Earthquakes and Earth's Interior based on Exascale Simulations of Seismic Wave Propagation" (Seismic Imaging) – US, Canada, France
- "Using Next-Generation Computers & Algorithms for Modeling Dynamics of Large Bio-molecular Systems" (INGENIOUS) -- Japan, UK, France, Germany, Russia

Extreme Scale Programming Models for Applications --- *continue to follow interdisciplinary paradigm established by SciDAC Program*



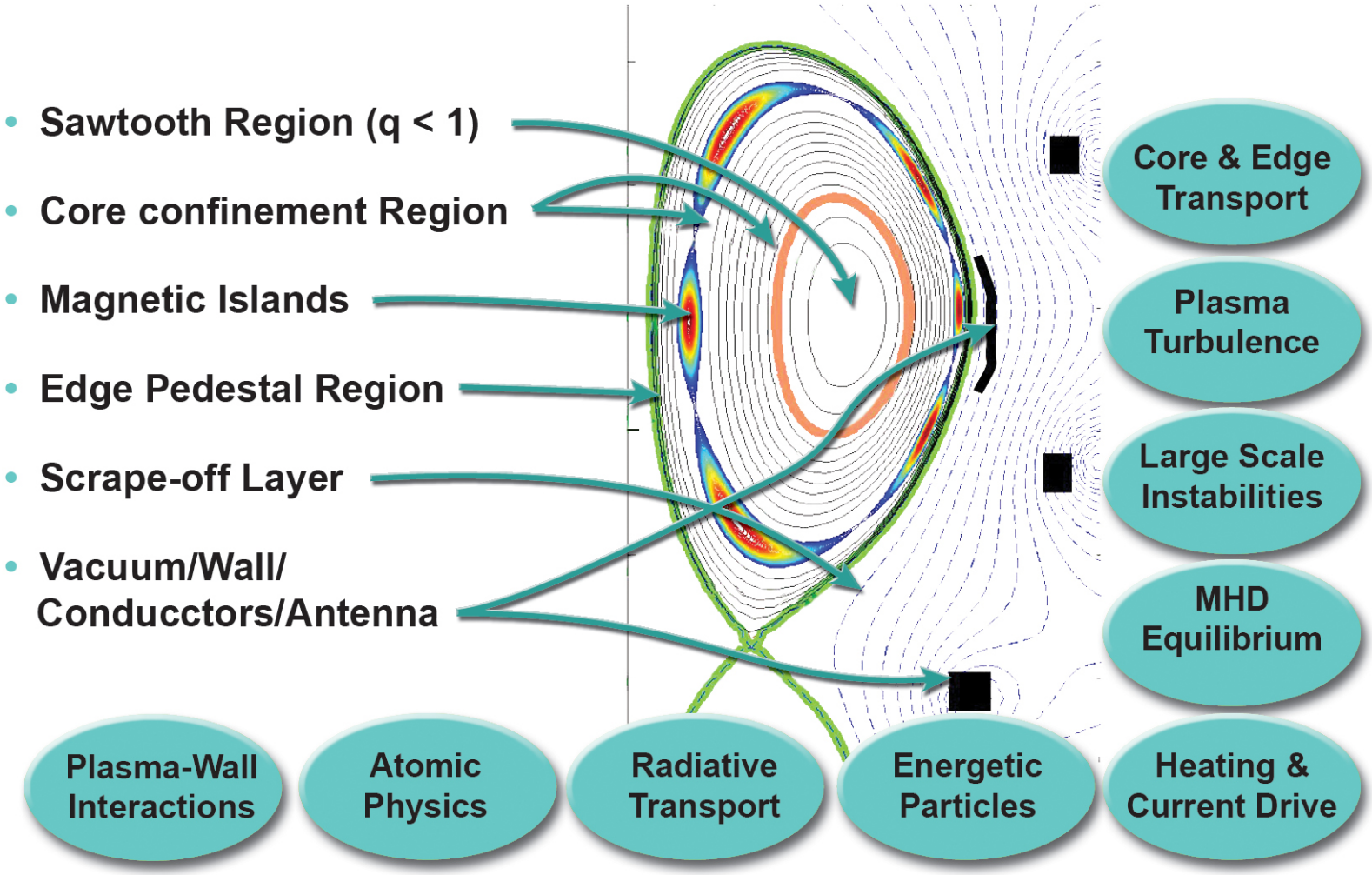
***Comparisons:** *empirical trends; sensitivity studies; detailed structure (spectra, correlation functions, …)*

Use the New Tool for Scientific Discovery

(Repeat cycle as new phenomena encountered)

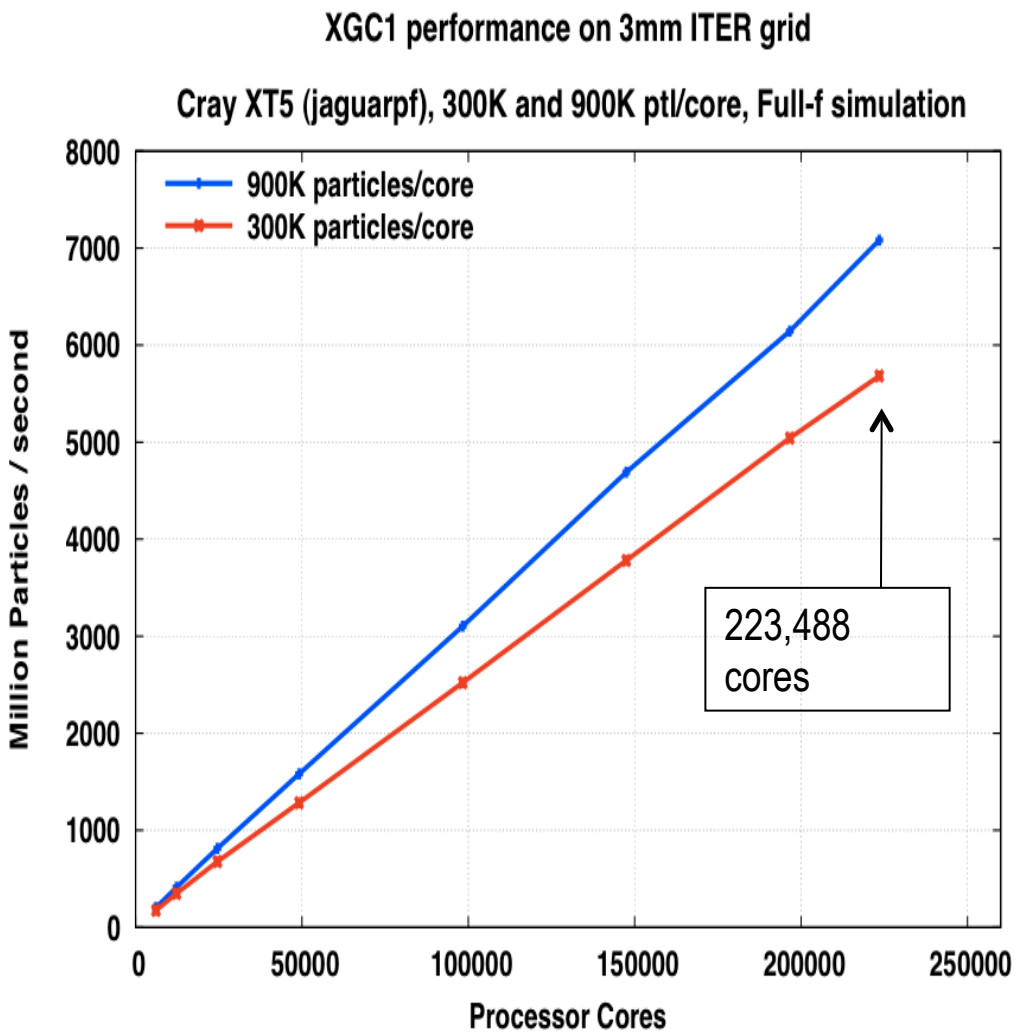
***Co-design Challenges:** *low memory/core; locality; latency; …..*

Elements of an MFE Integrated Model ➔ *Complex Multi-scale, Multi-physics Processes*

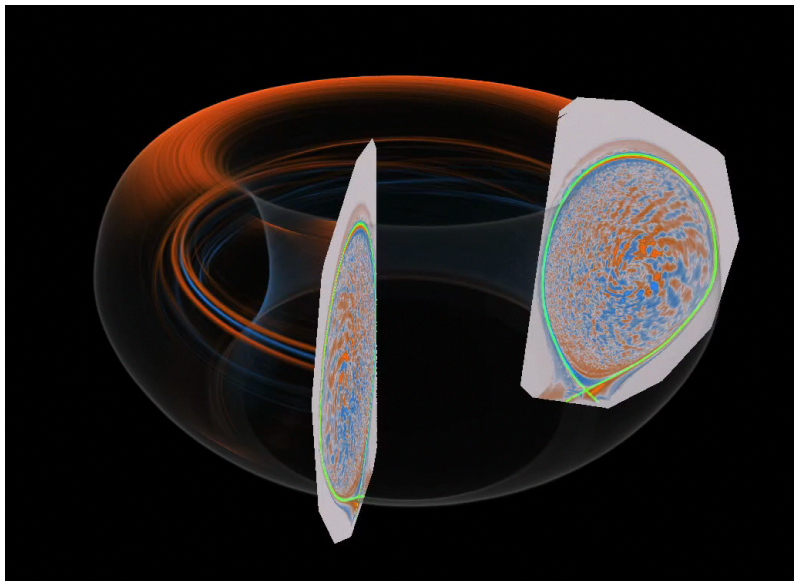


• W.Tang, D. Keyes, et al., ***"Scientific Grand Challenges: Fusion Energy Sciences and the Role of Computing at the Extreme Scale,"*** PNNL-19404, 212pp (March, 2009). http://www.er.doe.gov/ascr/ProgramDocuments/Docs/FusionReport.pdf

Integrated Plasma Edge-Core Petascale Studies on Jaguar



- C.S. Chang, et al., SciDAC-2 "CPES" Project: petascale-level production runs with XGC-1 require 24M CPU hours (100,000 cores × 240 hours)

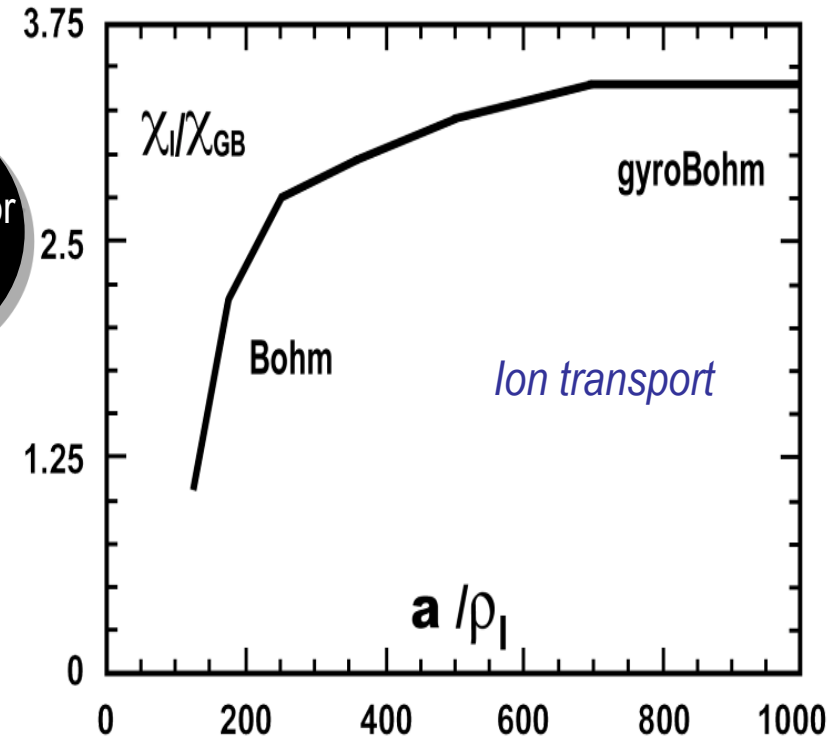


- XGC1 scales efficiently all the way to full Jaguar petaflop capability (with MPI+ OpenMP) & routinely uses >70% capability
- New SciDAC-3 "EPSi" Project: to address ***XGC1 conversion to GPU architecture of Titan***

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/ρ_i = 400 (JET, largest present lab experiment) through
  - a/ρ_i = 1000 (ITER, ignition experiment)

Good news for ITER!



- *Multi-TF simulations* using GTC global PIC code [e.g., Z. Lin, et al, Science, *281, 1835 (1998), PRL (2002)*] deployed a billion particles, 125M spatial grid points; 7000 time steps at NERSC ➔ *1st ITER-scale simulation with ion gyroradius resolution*

- ***Understanding*** physics of favorable plasma size scaling trend demands *much greater computational resources* + improved algorithms [radial domain decomposition, hybrid (MPI+Open MP) language, ..] & modern diagnositics

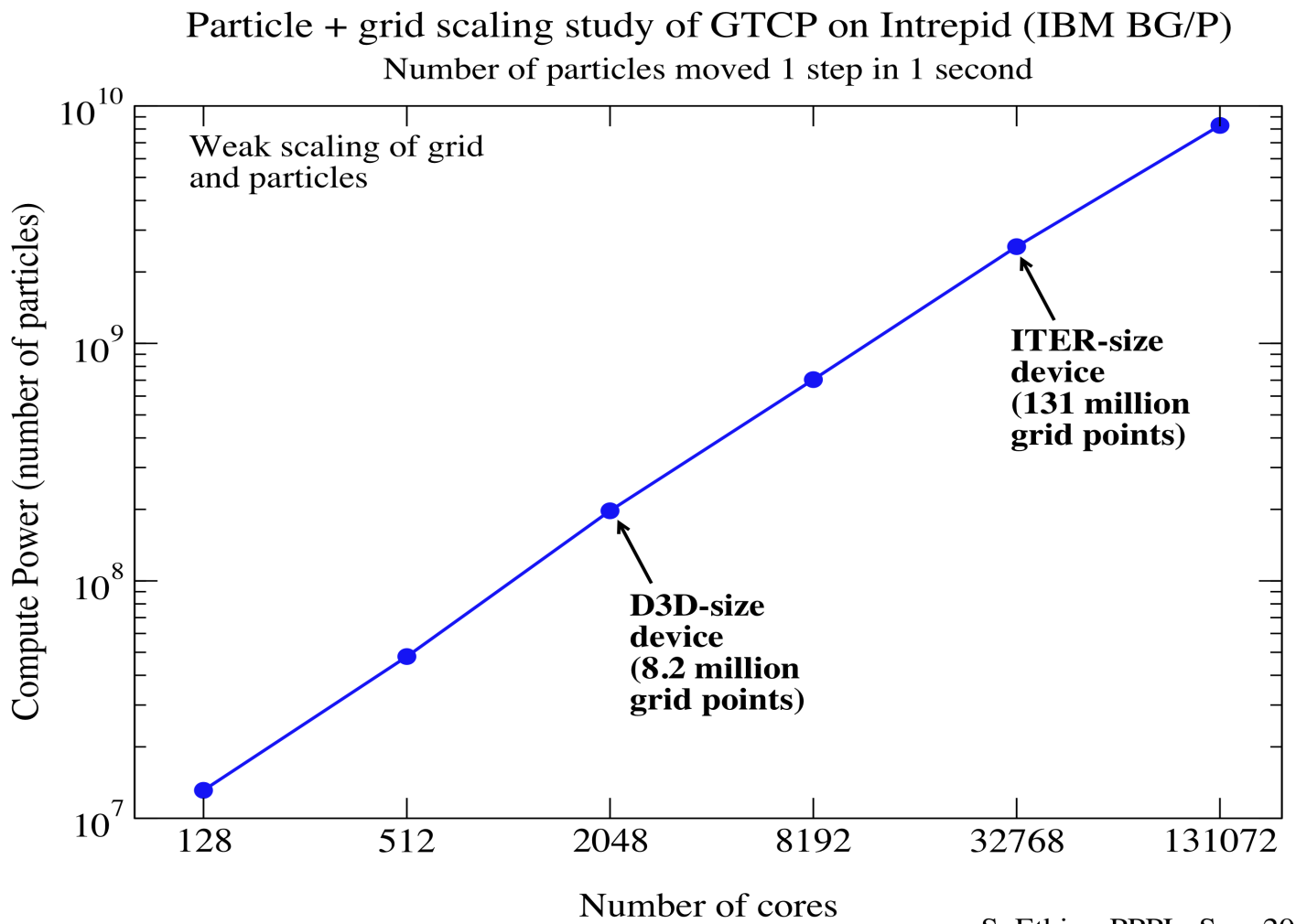
  -- *current Early Science Applications (ESA) GTC-P project on ALCF*

➔ ***Excellent Scalability of Global PIC Codes on LCF's enables advanced physics simulations to improve understanding***

- Global PIC code development ***for GPU and other low memory/core environments*** actively pursued

[e.g. -- ***SC2011 Paper on GPU version of GTC; 2011 Beijing Exascale CoDesign Workshop → GTC on Tianhe-1A, China***]

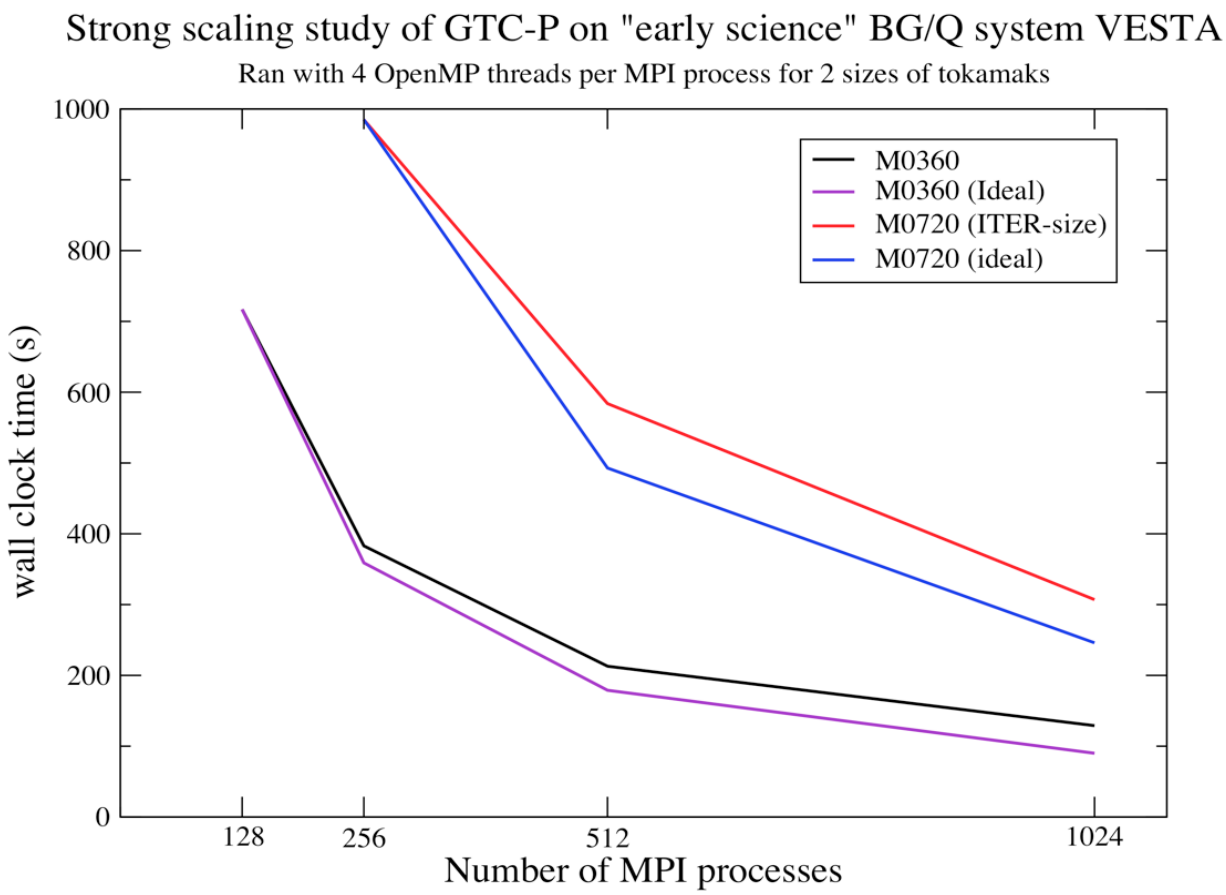
Weak Scaling Study GTC-P on IBM BG-P at ALCF



S. Ethier, PPPL, Sep. 2009

- **Excellent scalability demonstrated [both grid size and # of particles increased proportionally with # of cores] → (*also on 294,912 cores, 72 racks @ JSC in Germany*)**
- **Plans in place for similar weak scaling collaborative studies on Fujitsu-K Machine in Japan**

Strong Scaling Study of GTC-P in "Early Science" Project on Single-Rack IBM BG/Q "Vesta" System at ALCF



Excellent performance demonstrated –> ***recent results from Early Science ALCF Project show ~ order of magnitude improvement on new (multi-petaflop) IBM BG-Q ("Mira")***

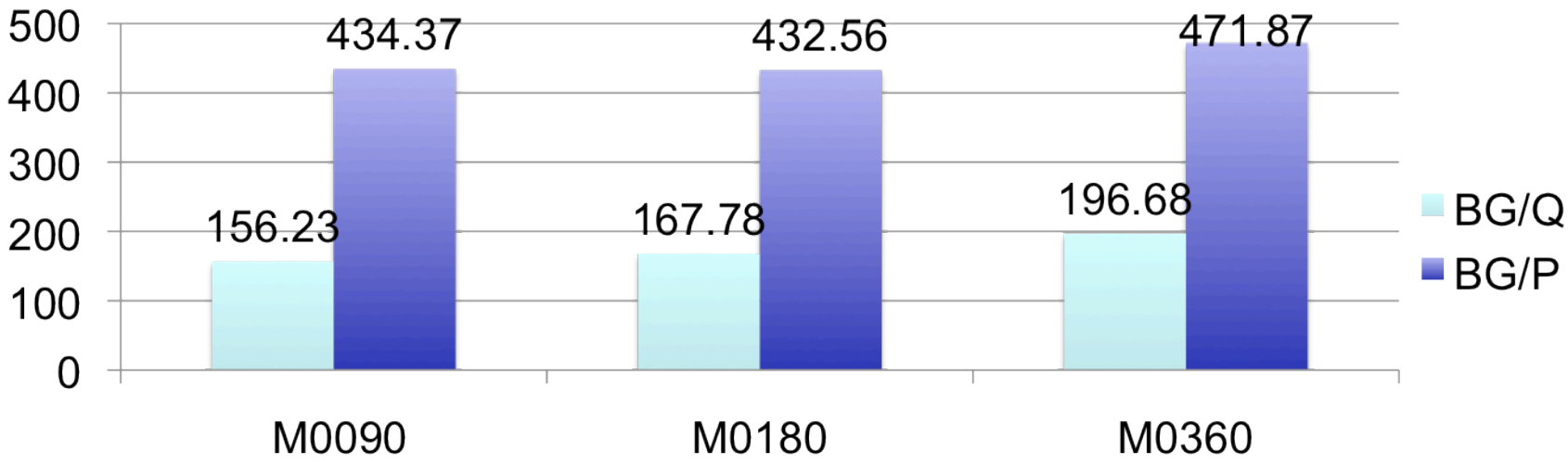


Figure: GTC-P performance comparison (seconds) on BG/P and BG/Q for each case with particle per cell (ppc)=100 for 100 steps [Note: *90, 180, 360 → problem size # of grid points in radial direction*]

M0090	Total # of nodes	Total # of cores	# of cores/ node	# of threads/ core	Time (s) for 100 steps	Speed up per core	Speed up Per node
BG/P	128	512	4	1	434.37	1.0	1.0
BG/Q	32	512	16	4	156.23	2.78	11.12
M0180							
BG/P	512	2048	4	1	432.56	1.0	1.0
BG/Q	128	2048	16	4	167.78	2.58	10.31
M0360							
BG/P	2048	8192	4	1	471.87	1.0	1.0
BG/Q	512	8192	16	4	196.68	2.40	9.60

Tables: test settings and performance results on BG/P and BG/Q

Performance Speed Up Comparison Results

M0180 ppc=100	**Our test**	**ANL**	**IBM**
Speed up per node (Q/P ratio)	10.31	10.7	11.2

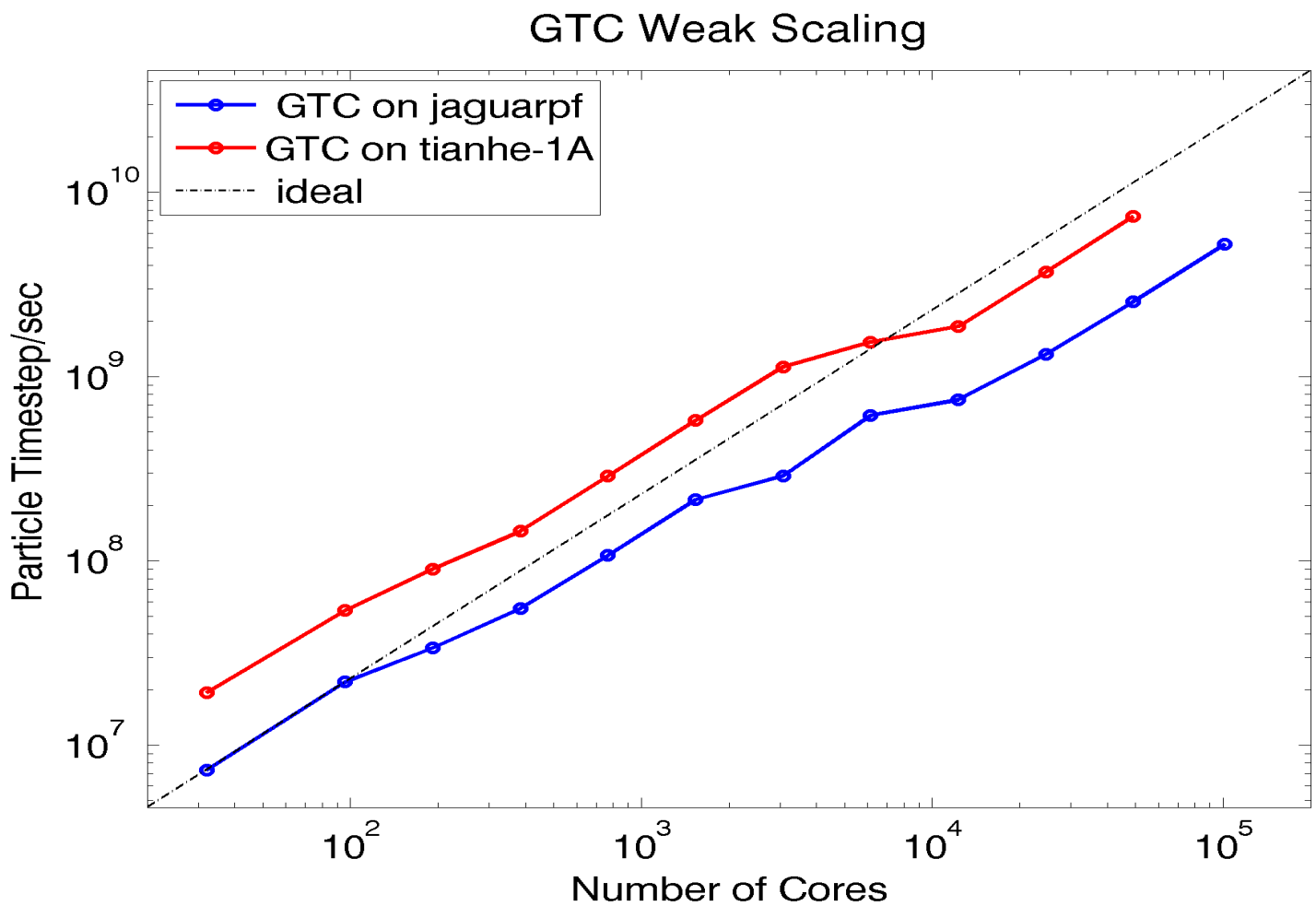
Table 2: Speed up per node ***comparison with ALCF and IBM results*** for M0180 problem size → 180 grid points in radial direction with particles/cell (ppc) =100 for 100 time-steps

GTC ON TIANHE-1A ➔ Particle-in-cell global kinetic turbulence code (GTC) running on CPU's only in scaling case study (Z. Lin, et al.)

Observations on improved performance:

- Tianhe-1A (8 core nodes) & Jaguarpf (12 core nodes) ➔ improvement actually ~ 1.7
- Improvement due primarily to Intel processor & compiler performance on Tianhe-1A
- GTC's relative insensitivity to communication time ➔ little benefit from Tianhe-1A's better network

--



New GTC-GPU Code *(K. Ibrahim, LBNL; B. Wang, Princeton U; et al.)*

Introduced at SC2011:

K. Madduri, *K. Ibrahim*, S. Williams, E.J.Im, S. Ethier, J. Shalf, L. Oliker, *"Gyrokinetic Toroidal Simulations on Leading Multi- and Manycore HPC Systems"*

- Physics content in GTC-GPU code *is the same* as in the GTC-P code
- Challenge: massive fine-grained parallelism and explicit memory transfers between multiple memory spaces within a compute node

- Approach: consider 3 main computational phases: *charge deposition, particle push and particle shift*

-- integrates three programming models [nVidia, Cuda, & OpenMP] within a node, and MPI between nodes

-- demonstrated excellent scaling behavior on NERSC Dirac test-bed (displayed in SC'11 paper)

-- explored breaking the limit of Amdhal's law on speedup by parallelizing - using atomics - the charge deposition phase, which has iterations with loop-carried dependency → *Memory locality improves performance of most routines but degrades performance for atomics because of access conflicts*

➔ *Conflicting requirements for locality and conflict avoidance* make optimizing the performance on GPUs both interesting and challenging.

GTC-GPU Optimization Progress

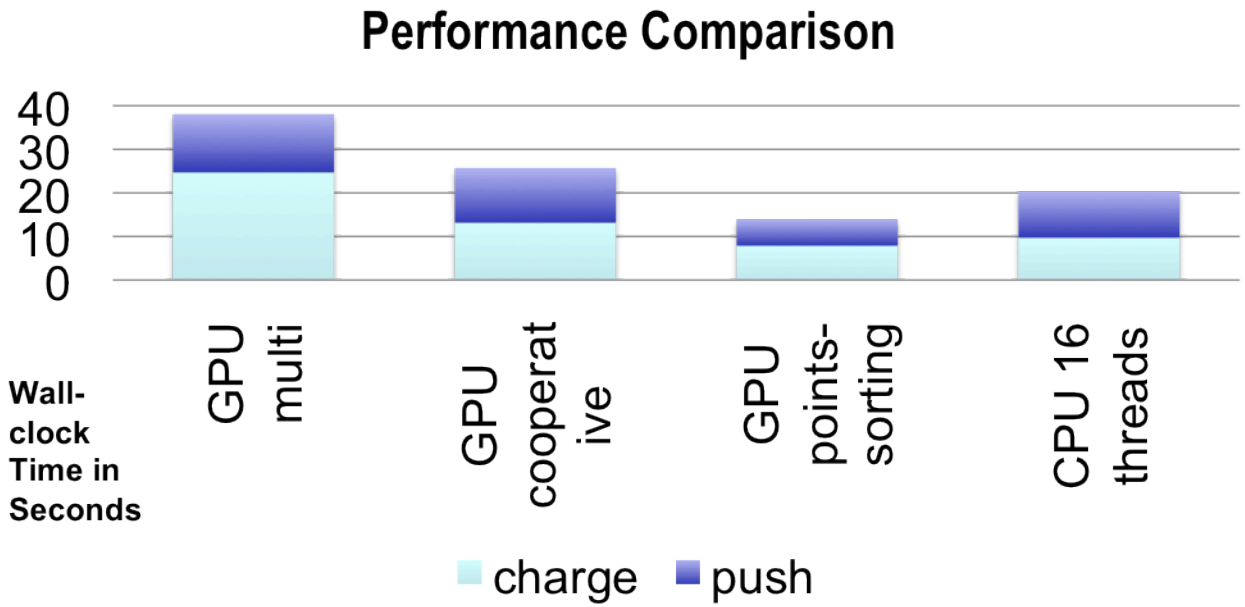
- Gather and scatter operations are key computational components in a PIC method ➔ *account for 80% of the total computational time in GTC*
- Challenge: Achieving *highly efficient parallelism* while dealing with

  (i) random access that makes poor use of caches; and (ii) potential fine-grain data hazards that serialize the computation
- Approach: Improve locality by sorting the particles by deposition grid points

  -- for gyrokinetic PIC method (where each finitie-sized particle is represented by four points on a ring) requires sorting these points instead of particles

  → *Sorting is an important pre-processing step in PIC method when dealing with GPU architecture (see next page)*

GTC-GPU Charge & Push Subroutine



- "GPU multi": the original GPU version
- "GPU cooperative": an improved GPU version that uses shared memory to achieve coalesced global memory access (for improved memory bandwidth)
- "GPU points-sorting": the up-to-date GPU version that uses shared memory to: (i) achieve coalesced global memory access; and (ii) reduce global memory access through points sorting
- "CPU 16 threads": the best optimized CPU version with 16 threads OpenMP

Note: *All tests illustrated here were carried out on Dirac (NERSC) with 10 ppc for 192 grid-points in radial dimension with 100 steps*

Problem Size Studies

Grid Size	B	C	D
mpsi	192	384	768
mthetamax	1408	2816	5632
mgrid	151161	602695	2406883

Problem Settings:

- mpsi: number of grid points in radial dimension
- mthetamax: number of grid points in poloidal dimension at the largest ring
- mgrid: total number of grid points in each plane

Problem size C corresponds to JET size tokamak
Problem size D corresponds to ITER size tokamak

The largest problem we can run on Dirac (a single 3GB Fermi GPU on each node) is C20 (JET size tokamak with 20 ppc).

• We expect to be able to run D20 (ITER size tokamak with 20 ppc) on Titan (with a single 6GB Kepler GPU on each node).

Summary: Programming Model Challenges in Moving toward Exascale

- **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!***

  -- scientific codes often use data structures that are easy to implement quickly but limit flexibility and scalability in the long run

- **Latency**: *Need to explore* ***highly multi-threaded algorithms*** *to address memory latency*
- **Flops vs. Memory**: *Need to utilize Flops (cheap) to better utilize Memory (limited & expensive to access)*

- **Advanced Architectures:** *Need to deploy innovative algorithms within modern science codes on* ***low memory per node architectures*** *–* ***(e.g, BG/Q, Fujitsu-K, Tianhe-1A, & Titan)***

  -- multi-threading within nodes, maximizing locality while minimizing communications

  -- large future simulations *(e.g., PIC ➔ need to likely work with >10 billion grid points and over 100 trillion particles!!)*

  → ***Significant Progress achieved with GTC-P on BG/Q ("Mira") & GTC-GPU on hybrid CPU-GPU systems***