Utilizing extreme scale HPC for full-f gyrokinetic particle simulation in realistic diverted tokamak geometry

S. Ku[a)], C.S. Chang [a)], E. D'Azevedo[b)], P. Worley[b)],
M. Adams[c)], S. Klasky[b)] and the CPES Team[d)]

[a)]Princeton Plasma Physics Laboratory
[b)]Oak Ridge National Laboratory
[c)]Columbia University
[d)]SciDAC Proto-FSP Center for Plasma Edge Simulation

PPPL PRINCETON PLASMA PHYSICS LABORATORY

CPES Center for Plasma Edge Simulation

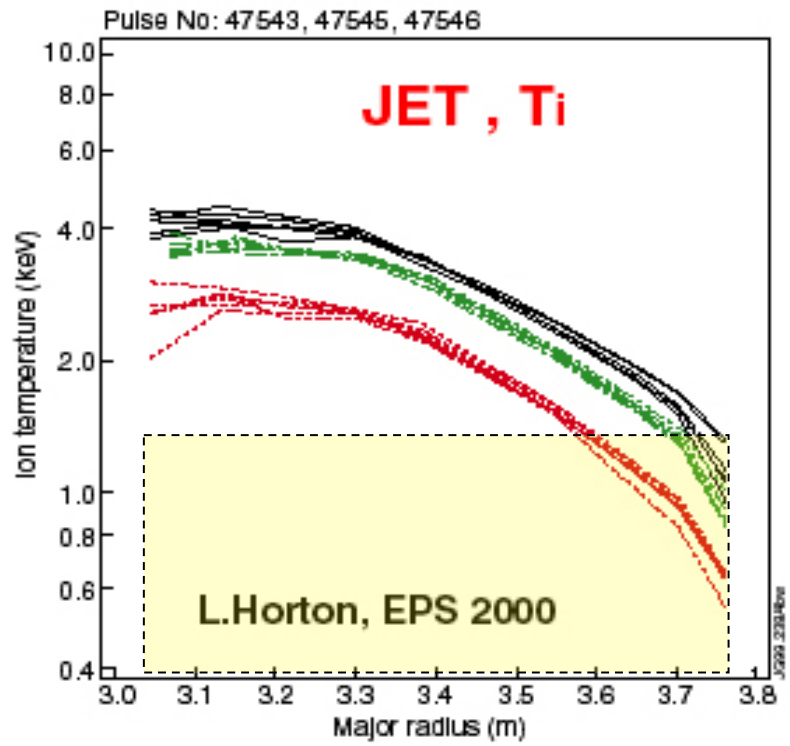
Outline

- Motivation
  - Importance of plasma edge for tokamak fusion
  - Non-locality of plasma transport
- XGC1
  - X-point included Gyrokinetic Code in realistic tokamak geometry across magnetic separatrix
  - Full-f method
- Utilizing extreme scale HPC
- Scientific findings
- Summary and On-going Development

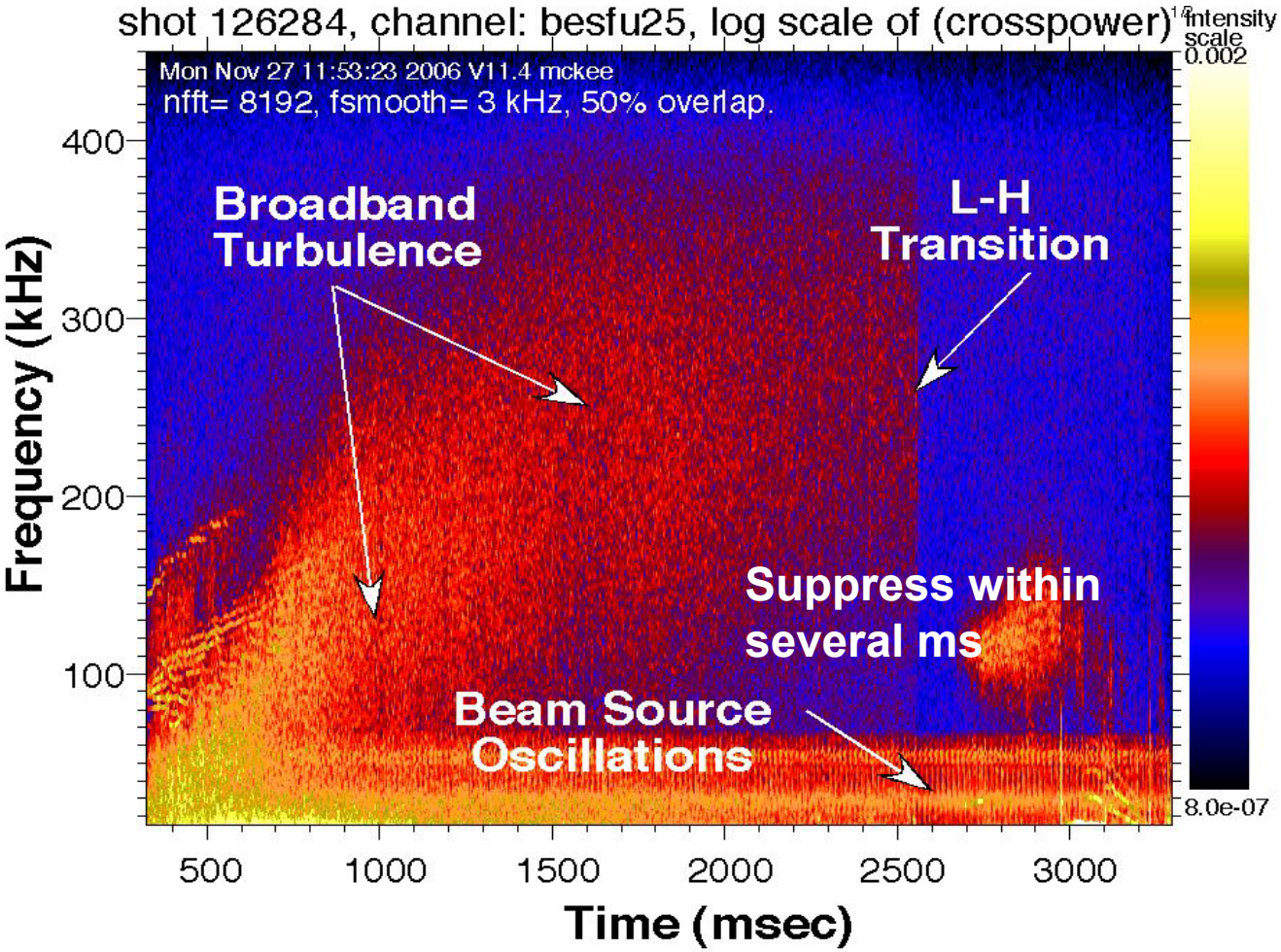
Edge physics is important for tokamak fusion

- High T_i at the central core is needed for efficient fusion power reactor.
- T_i profile is stiff.
  - High edge T_i is critical for high core T_i.
  - T_i cannot be high near the material wall.

  → H-mode pedestal
- Edge determines the core fusion condition: The tail wags the dog.
- We need to understand these edge pedestal physics and the nonlocal core-edge interaction.



Edge condition affects core transport properties

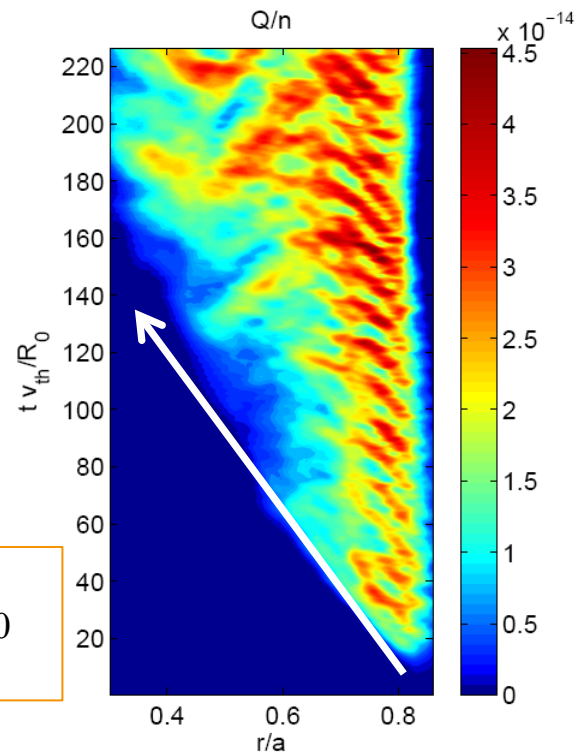


Density fluctuations at r/a = 0.65 G.R. McKee, et al. Plasma Fusion Res.(2007)

Non-local turbulence transport

- XGC1 finds that ITG turbulence can indeed invoke non-local core-edge interaction in several ms.
  - Turbulence onset and propagation
  - Cold pulse experiment
  - Avalanche process

- Full-f whole volume simulation is needed on extreme scale HPC
  - Realistic diverted geometry
  - Non-local turbulence and neoclassical transport together
  - T_i profile evolution



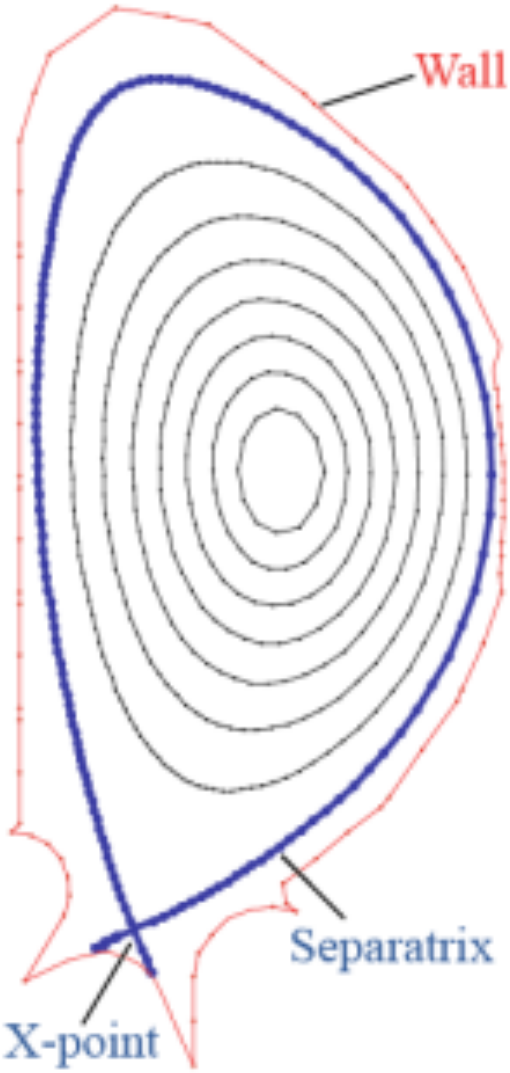
Start of edge cooling at t=0

XGC1

- Full-f, **X**-point included **G**yrokinetic **C**ode in realistic tokamak geometry across magnetic separatrix
- Particle in cell method
- Spatial simulation domain: whole tokamak plasma volume from magnetic axis to wall, with realistic tokamak edge geometry and Dirichlet wall boundary condition (grounded wall).
- Field-line following unstructured triangular grid
- Simulation size of production run :
  - ~150K - 200K cores, 1 day
  - Number of particles ~ 20 Billions
  - Number of cells ~ 2 Millions (DIII-D, C-mod)
  - Particle per cell (grid node) ~ 10,000
- Current capabilities: Electrostatic ion turbulence without scale-separation, with heat source and conserving linear and non-linear Coulomb collisions
- Developing capabilities: Electromagnetic turbulence with kinetic electrons.

Coordinate system for edge

- Magnetic separatrix is singular surface for core codes which use magnetic coordinate system.
- To handle separatrix and X-point, XGC1 uses Cartesian coordinate system for particle pushing.
- Equilibrium-B following, unstructured triangular grid for field solving.



Full-f vs δf method

- f is 5D phase space particle distribution function
- Marker particles can represent either full-f or δf
- $\delta f = f_{full} - f_0$:
  - calculates perturbed physics on fixed f_0
  - assumes conservative system
  - Efficiency computing.
  - Easier to simulate, but the self-consistent self-organization between mean and perturbed physics is missing.
  - Particle lost on the wall boundary is hardly described with δf method.

Full-f vs δf method

- Full-f:
  - Solve the 5D equation system without scale-separation between the mean and the perturbed physics.
  - More difficult to simulate.
  - HPC is a necessity: Much higher # of particles
  - Only a few production codes in the world
  - XGC1 may be the only production code in the world, which can handle realistic diverted geometry.
    - Edge plasma in contact with material wall must be full-f (non-equilibrium thermodynamics system).
    - Particle noise does not grow with time.

    → Simply increase # of particles to control particle noise

XGC1 for powerful HPC

- Mathematical model:
  - Particle time advance: ODE (Runge-Kutta and Predictor-Corrector)
  - Poisson equation: PDE (FEM, Algebraic Multi-Grid, HYPRE + PETSc)
- Hybrid parallelization: MPI/OpenMP
- Performance Optimization
  - Efficient particle-grid search
  - Dynamic load balancing of particles
  - Memory localization with multi-domain decomposition with nearest-neighbor communication
- Advanced Adaptive I/O (ADIOS)

XGC1 Wallclock Time: MPI vs. OpenMP

300K particles/thread, 12 cores per node (P. Worley and PERI)

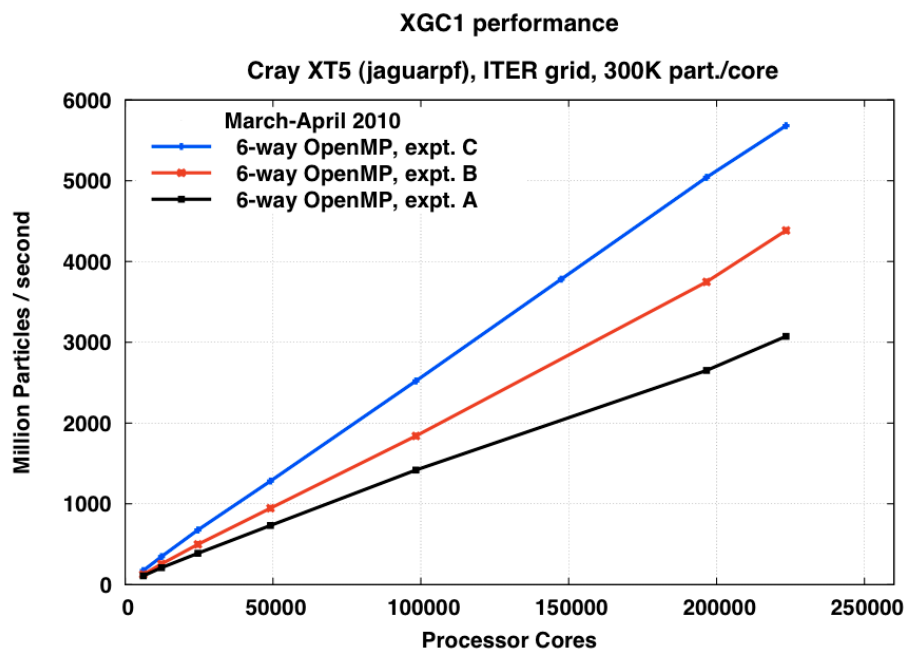
		Seconds for 10 timesteps (threads per process)			
Nodes	Cores	1	1x	6	12
512	6144	186	142	106	138
1024	12288	204	153	106	123
2048	24576	232	173	109	120
4096	49152	-	190	115	121
8192	98304	-	185	117	126
12288	147456	-	-	117	140
16384	196608	-	173	117	134
18624	223488	-	-	118	131

- "1x": using only 8 cores per node, so problem size only 0.67 that of other data.
- MPI-only not scaling well, and never competitive when using 6144 or more cores.
- 6-way OpenMP best performer in these experiments.

XGC1 performance has been improved efficiently at the maximal Jaguarpf capacity

300K particles/thread, 12 cores per node, 2 MPI processes per node

- The efficient scalability is preserved as we improve the algorithms toward more extreme scale computing.
- (E. D'Azevedo, M. Adams, P. Worley, S. Klasky, etc; and SciDAC PERI, TOPS, SDM)



Plotting number of particles processed per timestep per wallclock second.

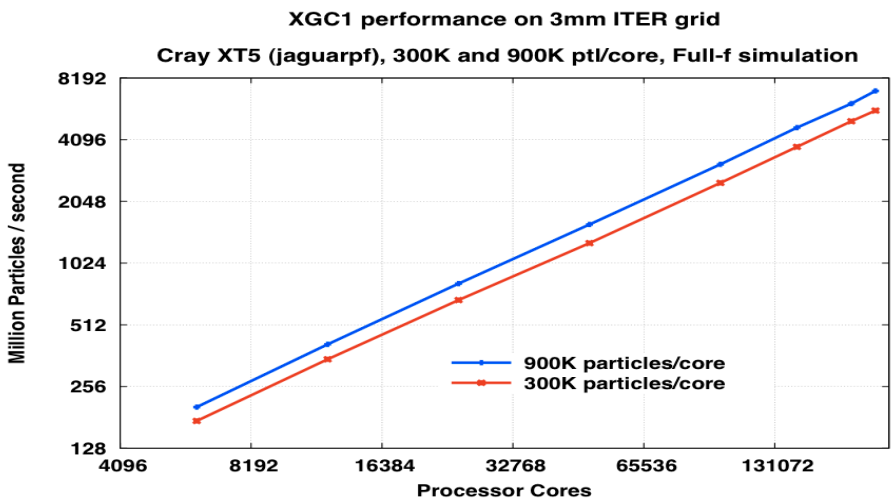
Expt. A: May, 2009 version of XGC1

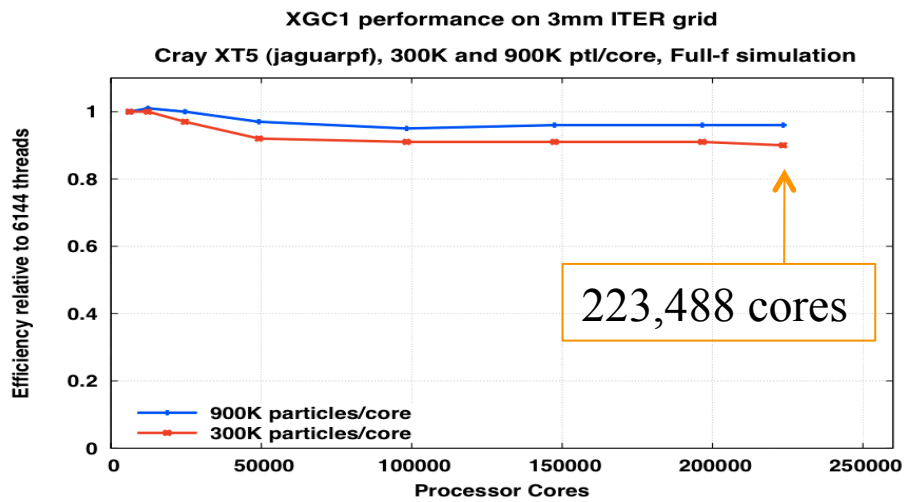
Expt. B: + a more efficient search technique for locating particle position in grid; removing array syntax within OpenMP-parallelized loops that was degrading OpenMP performance; thread-safe random number generation

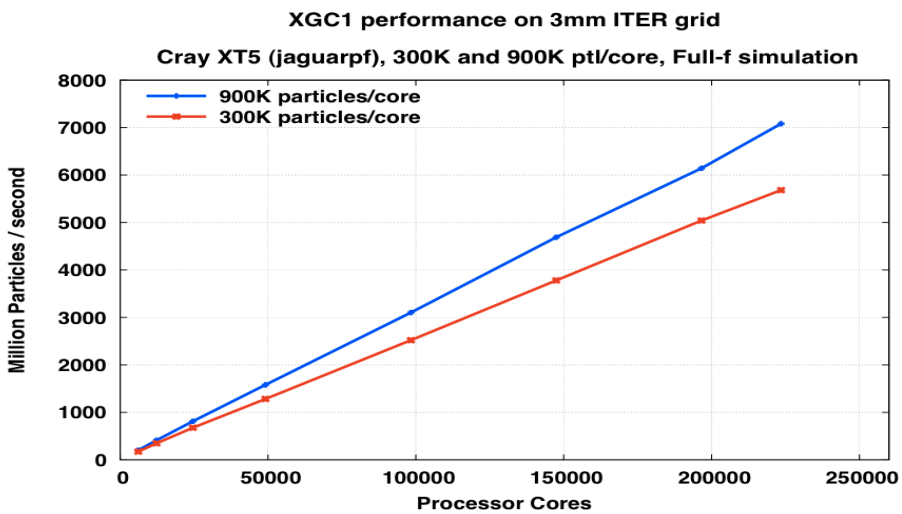
Expt. C: + spline interpolation optimizations → factor of two improvement

XGC1 scalability and Efficiency vs computational intensity

12 cores per node, 2 MPI processes per node





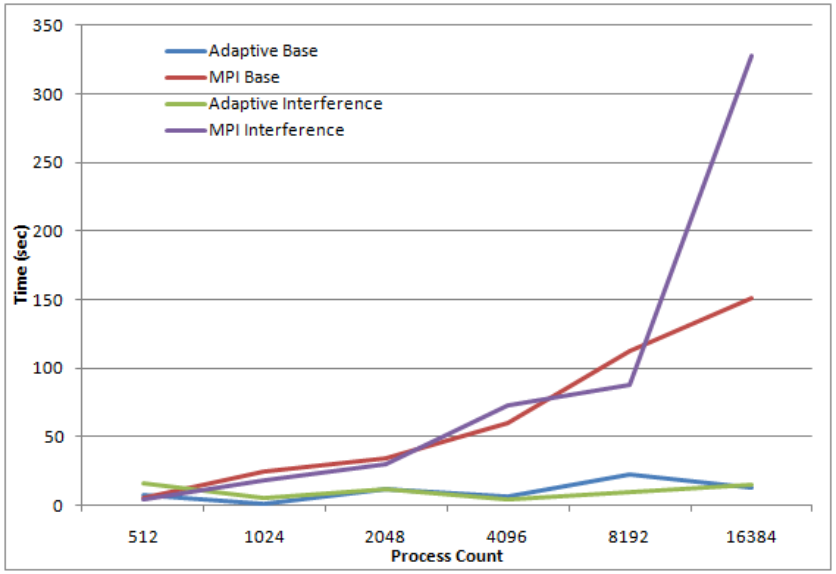


- 900K particles per thread problem is more computationally intensive than 300K problem, which leads to a somewhat higher particle push rate (approx. 20%).
- Performance scaling is excellent for both problems.

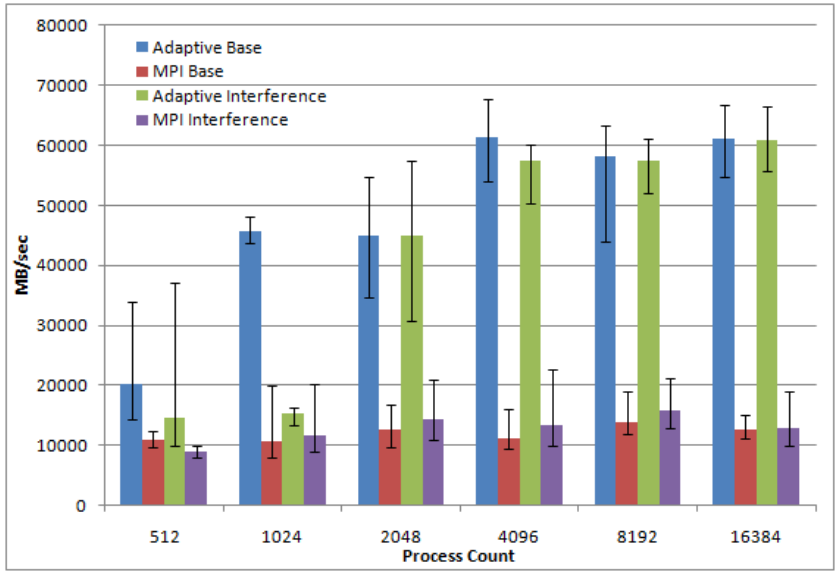
Fast I/O is a necessity
XGC1 with ADIOS 1.2 Adaptive I/O

- Restart file size of XGC1 >~ 2 TB.
- Without Adios (parallel HDF5), 2TB restart file was taking > 1hour for every hour of run!
  - Shift from slower to faster storage targets to manage external interference
  - New methods (blue/Green): >50GB/s: takes ~ 1m for 2Tb restart file

I/O variability in new Adios

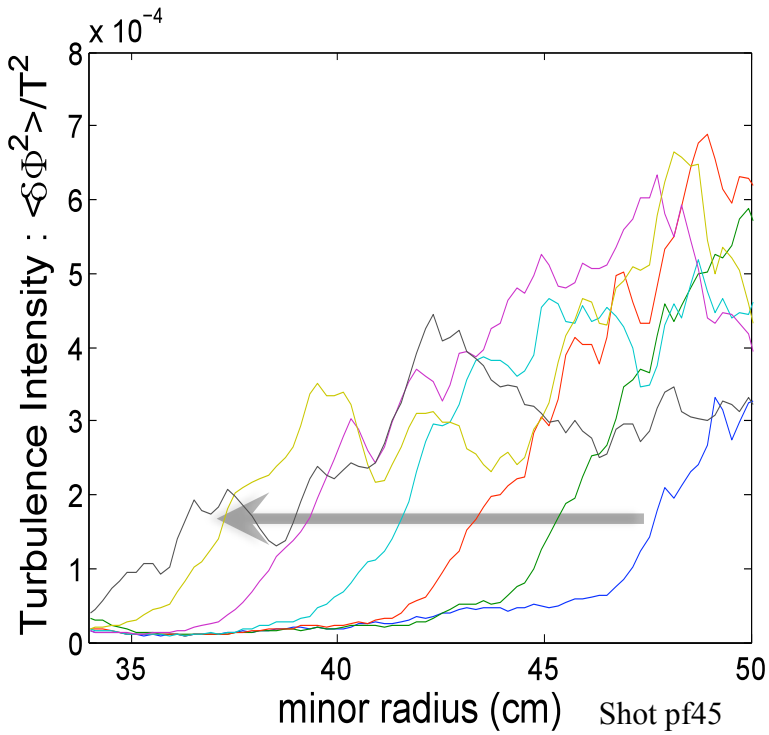


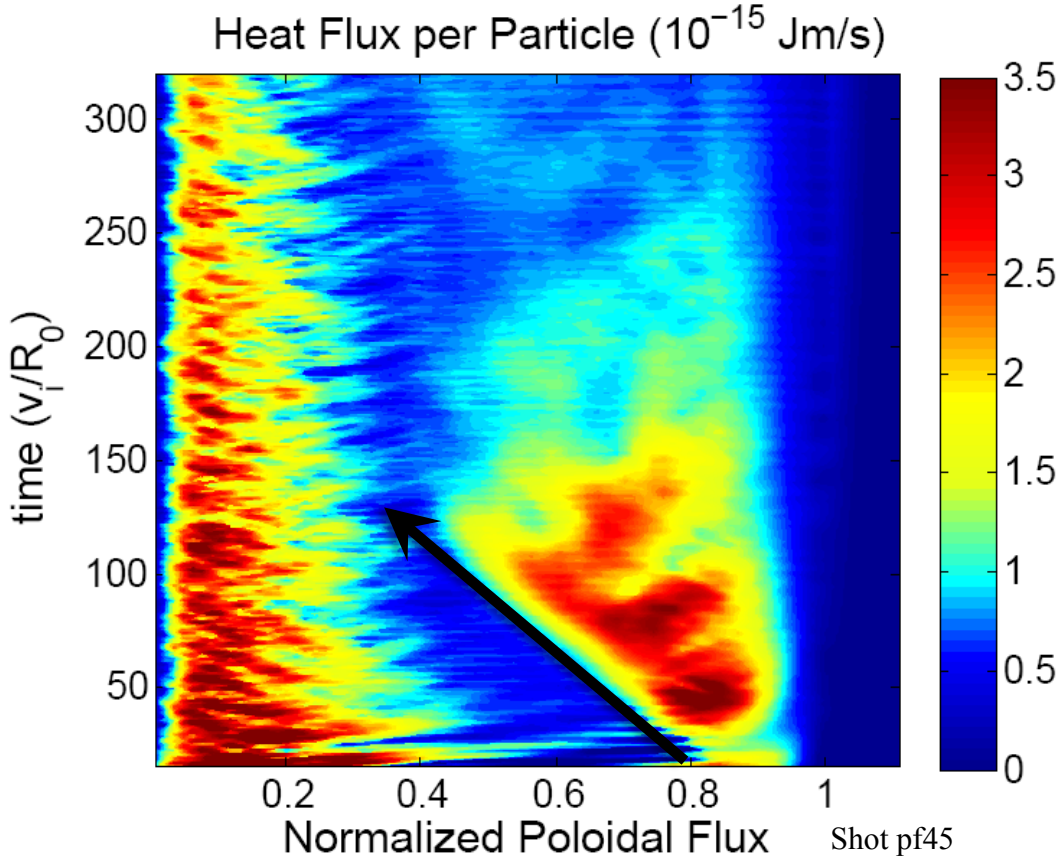
XGC1 I/O performance on a busy system
196,608 cores



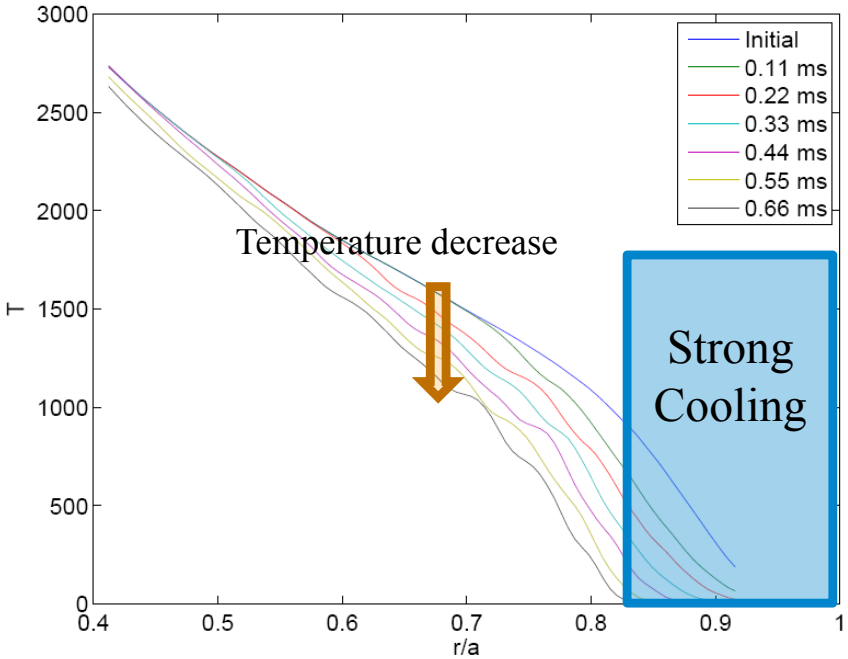
"Tail wags the dog" Phenomena

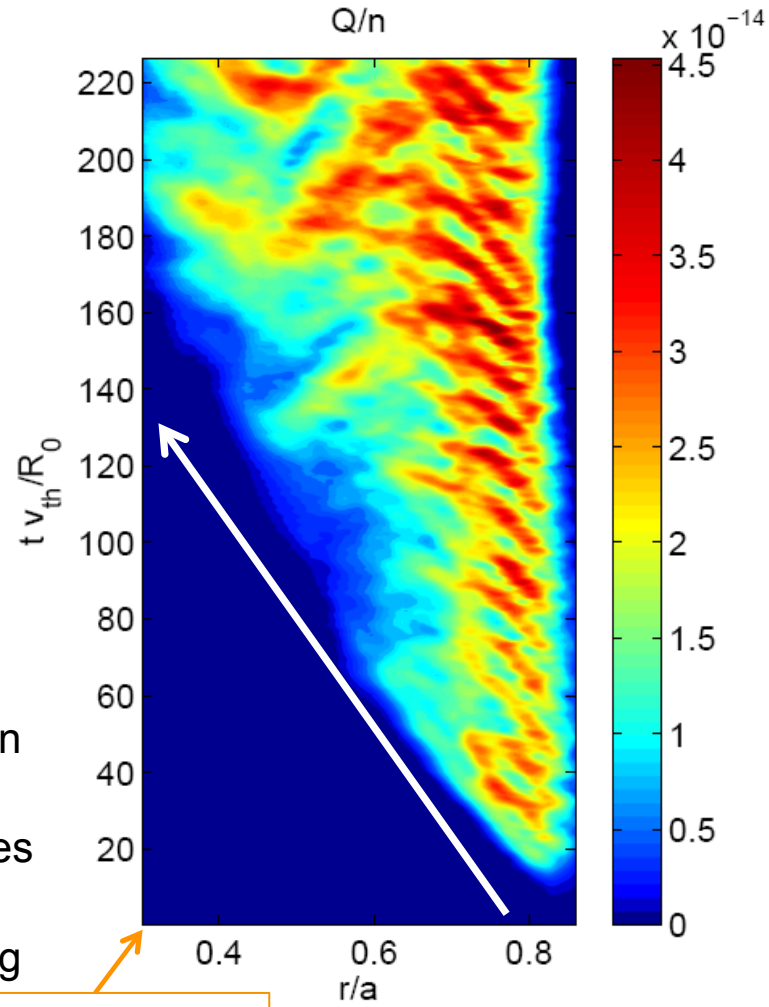
- Turbulence is generated at pedestal top and propagates into core plasma.
- Ballistic inward propagation of turbulence from pedestal top at speed ~ 0.3 km/s ~ 1.7 $\rho_{(i,0.6)} v_{(i,0.6)} / R_0$





Cold pulse simulation

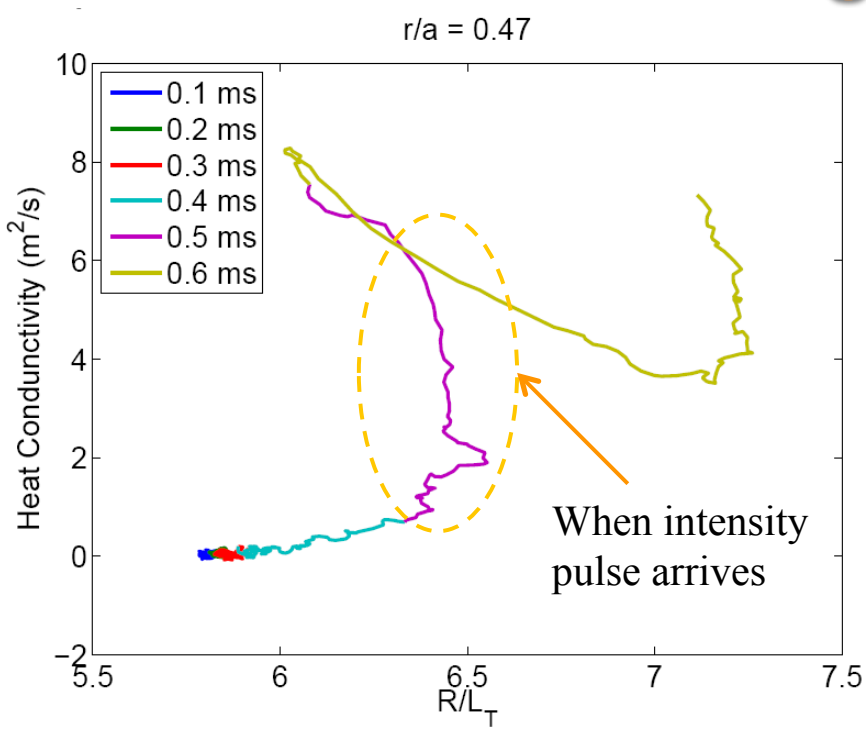




Start of edge cooling at t=0

- ITG turbulence in XGC1p (simple geometry version of XGC1)
- Strong cooling at the edge after the plasma reaches quasi steady state.
- Intensified turbulence starts from the strong cooling layer.
- Intensified turbulence is propagating inward.

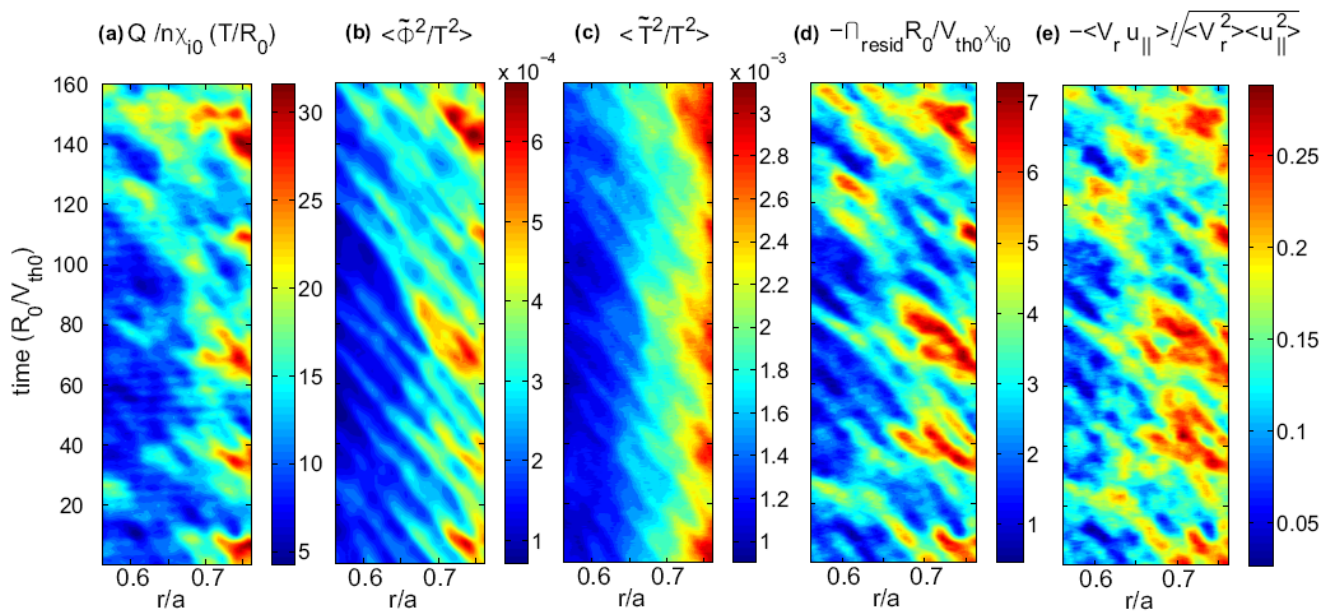
Local stiffness reflects incoming cold pulse



- Local stiffness model $\chi = \chi_{base} + \chi_s \left(L_{Tc}/L_T - 1\right)^{\gamma} H(L_{Tc}/L_T - 1)$
- Local stiffness reflects the incoming cold pulse turbulence.
- Higher heat conductivity due to incoming intensity pulse.
- Speed of cold pulse ~= speed of intensity propagation
- Nonlocal turbulence phenomena

Study of momentum transport

- Numerical "Solomon Experiment:" Add torque to cancel intrinsic rotation
  - Investigate property of ITG-driven residual stress (non-rotation driven)
  - Inward intensity pulse drives residual stress as well as heat flux.
  - Non-local transport phenomena for momentum transport



Summary and On-going development

- XGC1 is full-f particle-in-cell code on realistic tokamak geometry including separatrix and wall.
- Scales efficiently to the maximal number of Jaguar cores for scientific discovery runs.
- Core-edge Non-local phenomena of turbulence transport are observed in experiment and XGC1 simulation.
- Near future capability:
  - Electromagnetic turbulence using full-f kinetic electrons in edge.
- Performance improvement:
  - GPGPU – to get ready for next generation HPC.