Computational Knowledge for Toroidal Confinement Physics

C-S Chang

Korea Advanced Institute for Science and Technology &
Courant Institute of Mathematical Sciences, New York University

Scientific Discovery through Advanced Computing (SciDAC)
Prototype Fusion Simulation Project
**Center for Plasma Edge Simulation** (CPES)

Contents

- Introduction
- Primacy hierarchy in computational and laboratory studies
- Nonlinear, multi-scale, and self-organization
- Different codes for different purposes
- Different hardware architectures for different codes
- Toroidal plasma simulations at first principles level
- Verification and validation
- Summary

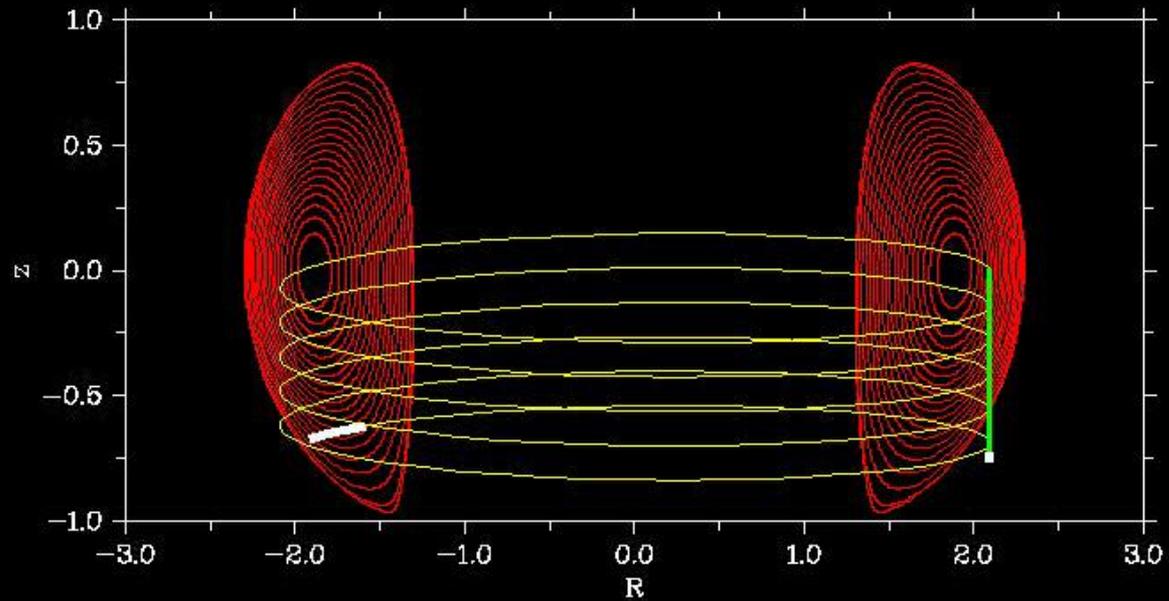References gathering will not be attempted in this lecture.

Simulation provides virtual laboratory
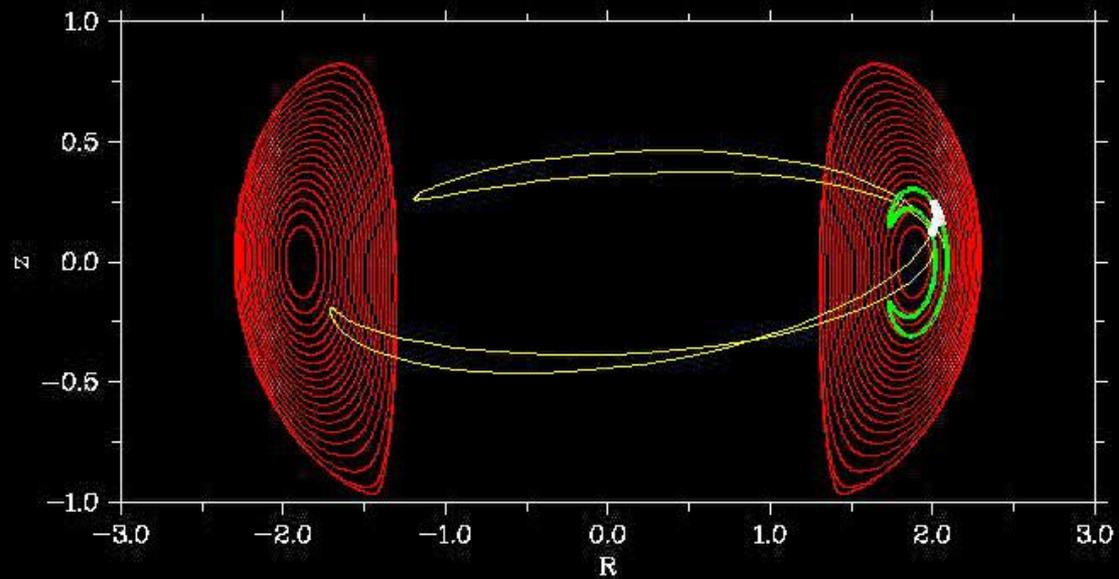## Particle diffusion in ExB convective cells
Hasegawa-Mima turbulence (G.Y. Park, low beta)
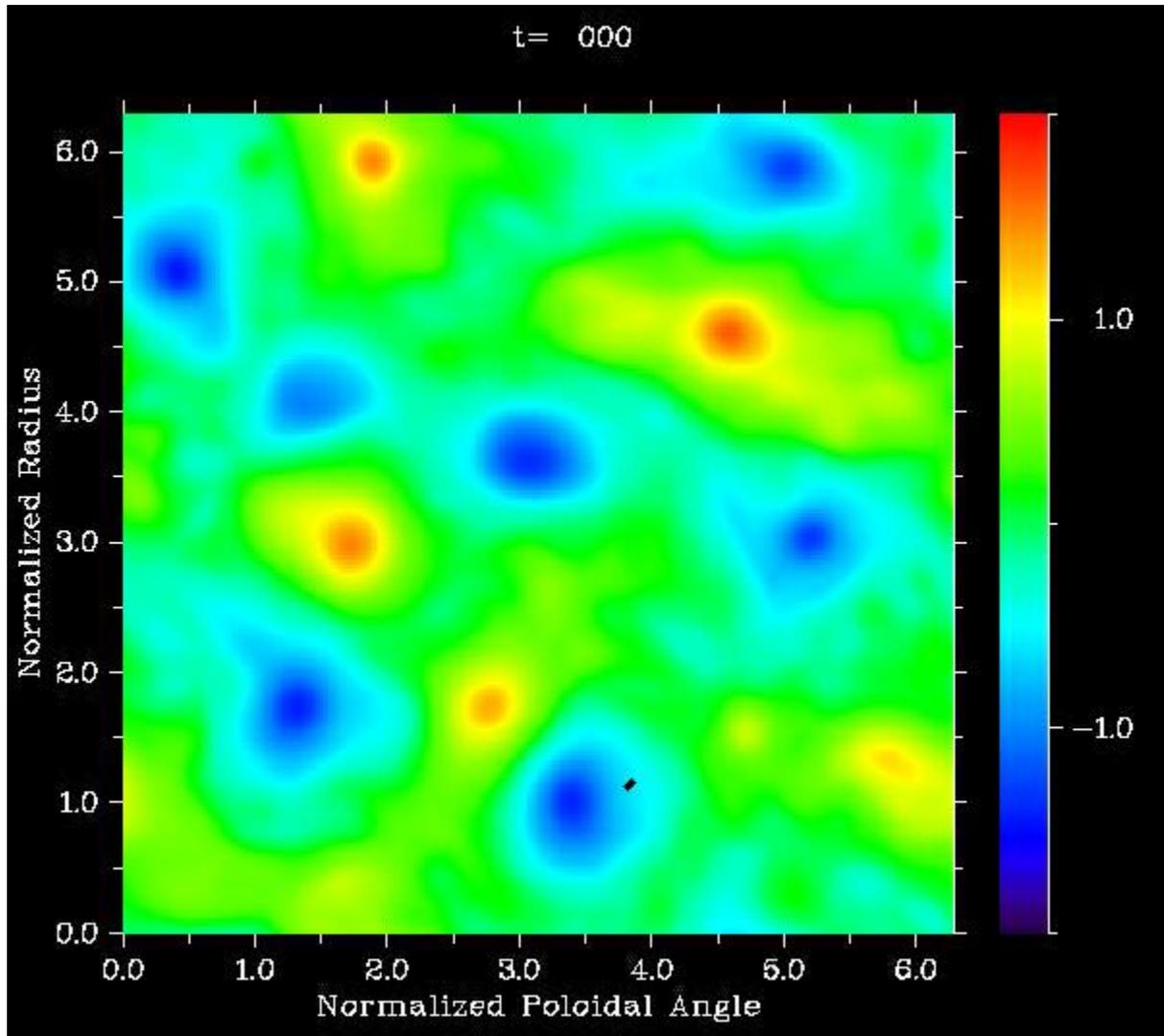
With no poloidal field, 100keV

Banana orbit, 100keV, S=0.

Introduction

- Computational physics is an exciting area for the future
  - US: 5 SciDACs (unit physics) and 3 proto-FSPs (integrated physics); FSP, Fusion Simulation Project for virtual tokamak, ~$25M/year (under planning)
  - EUFORIA (EU Fusion fOr ITER Applications)
  - Japan: Burning Plasma Simulation Initiative (BPSI)
  - China, India, Korea

- Upon analytic and experimental supports, the computational physics can bring breakthroughs.
- Shortage of qualified scientists in computational fusion physics is becoming a serious issue.

Statistical (primacy) hierarchy levels

What physical quantities are we trying to compute?
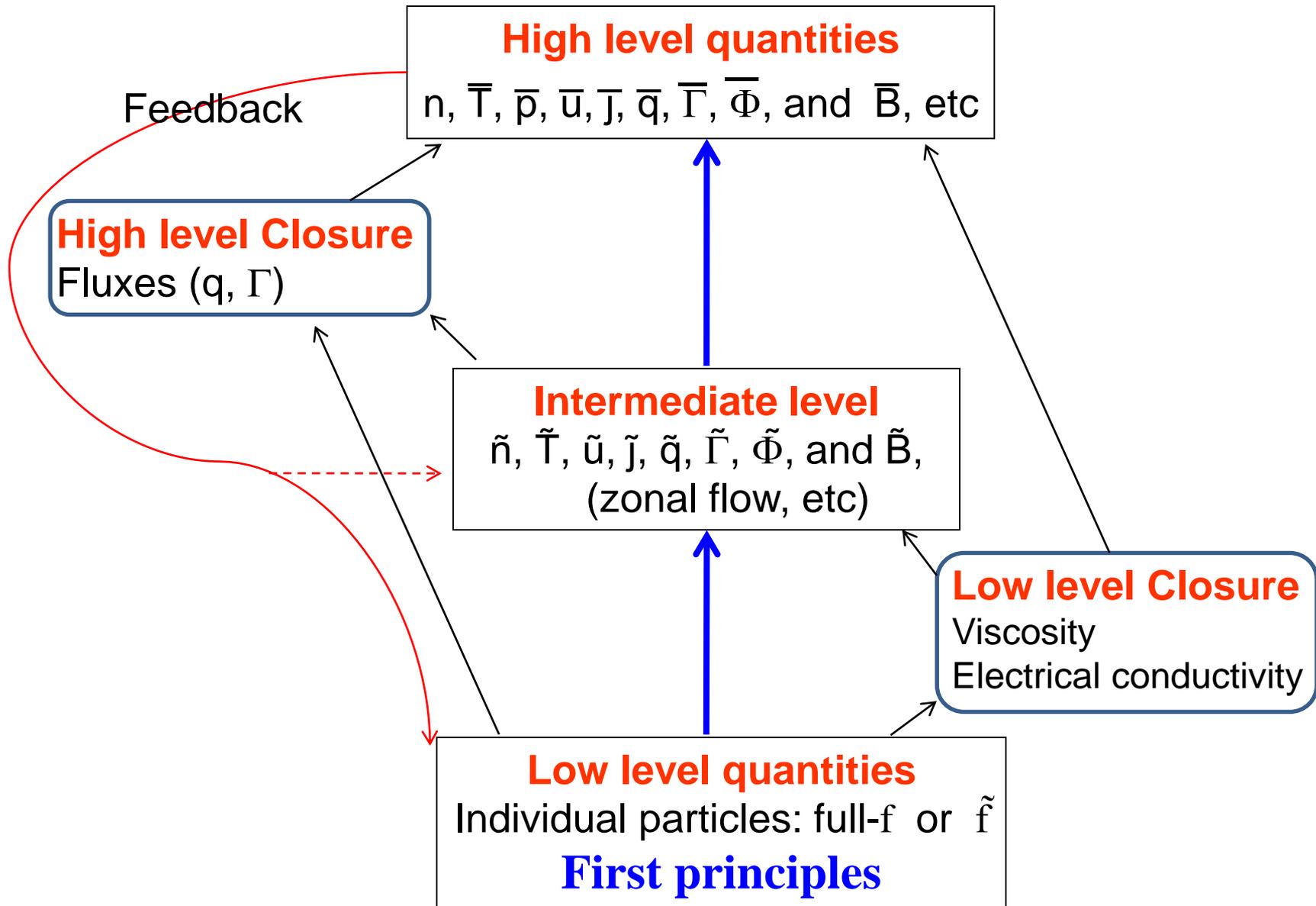
Nonlinear, multiscale, self-organization

- **The low level microscopic physics determine the spatio-temporal behavior of the high level macroscopic quantities through the hierarchy. At the same time, the macroscopic quantities affect the microscopic particle behaviors.**

- **All level physics phenomena need to be solved together self-consistently for a first principles simulation.**

- Individual behaviors (particles) are nonlinearly coupled to the whole, leading to a self-organized behavior which could not be understood from the individuals.
    - Elementary particle $\rightarrow$ atomic physics $\rightarrow$ molecules $\rightarrow$ cell $\rightarrow$ tree or dog
    - Individual particles $\rightarrow$ Super conductivity
    - Stock market, History, war etc

- The matter self-organizes to give us the "laws" or "principles," and the fundamental constants.

- New self-organized "laws" and "principles" are to be found from nonlinear-multiscale-complex study either by laboratory or virtual experiments.

Which level simulations do we need for ITER?

ITER may need simulation codes at all levels.

- Control room requires reduced transport model codes at the Highest level (reduced model), running on dedicated local CPUs: $\bar{n}$, $\bar{T}$, $\bar{p}$, $\bar{u}$, $\bar{j}$, $\bar{q}$, $\bar{\Gamma}$, $\bar{\Phi}$, and $\bar{B}$: Use closure information qualified by experiment or **first principles simulations.** **Key: how to implement self-organization?**

- Fluid and gyrofluid turbulence codes: reduced model codes, require closure and are less accurate, but are able to execute speedy evaluation of the intermediate level hierarchical quantities (fluid fluctuations).

- First principles codes evaluate the low level hierarchical quantities, obtain basic understandings from virtual experiments, and provide closure to higher level codes.

Expensive

High and intermediate level codes

These valuable codes have been developed with the limited computing resources in mind (fast results!).   They need support from the first principles codes.

- High level, reduced transport model codes: eg, **TRANSP**

$$\frac{\partial n}{\partial t} = - \frac{1}{r}\frac{\partial}{\partial r}\left(rnV\right) + S$$

$$\frac{3}{2}\frac{\partial}{\partial t}nT + \nabla\cdot\frac{5}{2}nT\vec{V} - \vec{V}\cdot\nabla nT + \pi_{\alpha\beta}\frac{\partial V\alpha}{\partial x_\beta} + \nabla\cdot\vec{q} = Q$$

$$\frac{\partial B_\theta}{\partial t} = \frac{\partial}{\partial r}\left(\frac{1}{r\mu_o\sigma(r,t)}\frac{\partial}{\partial r}rB_\theta\right),$$

- Intermediate level
  - Fluid, gyrofluid, and MHD: instability/turbulence codes

Gyrofluid equations

- G. Hammett -

- Gyrofluid equations derived by integrating full gyrokinetic equation over $v_{||}$ & $v_{\perp}$, to derive conservation laws for particles, parallel momentum, parallel energy, & average magnetic moment, etc.

- Reduces 5-D problem to 3-D, much faster

- Have to introduce closure approximations, *e.g.* Landau-damping and phase-mixing which are approximated by k-dependent parallel heat diffusion coefficients:

$$D_{||} \propto \frac{v_t^2}{\left|k_{||}\right| v_t + \nu}$$

- Collisionality is roughly handled by a 3-pole approximation to the Z-function for linear Landau damping [Hammett et al.]

- Still an approximation: qualitatively similar to full gyrokinetics in many cases but can have factor of ~2 errors. Used for physics studies like shear-flow suppression of turbulence.

- Improved closures by B. Scott is used for edge turbulence in GEM.

First principles delta-f gyrokinetic code

Gyrokinetic Vlasov equation df/dt=Lf=C(f)

$$L = \frac{\partial}{\partial t} + (v_{\parallel}\mathbf{b} + \mathbf{v}_d + \mathbf{v}_{E\times B}) \cdot \frac{\partial}{\partial \overline{\mathbf{R}}} - \mathbf{b}*\cdot\nabla(\mu B + <\Phi>_\alpha)\frac{\partial}{\partial v_{\parallel}} \qquad \mathbf{B}^* = \mathbf{B} + (Bv_{\parallel}/\Omega_s)\nabla\times\mathbf{b}$$

Gyrokinetic Poisson equation

$\nabla^2\Phi$

$$-\nabla^2\phi = \sum_s \left[4\pi e_s \int \overline{\delta f}_s \delta([\overline{\mathbf{R}} + \overline{\rho}_s] - \mathbf{x}) \mathcal{J}_s d^6\overline{Z} - \frac{1}{\lambda_{Ds}^2}\left(\phi - \langle\overline{\phi}\rangle_\alpha\right) \right]$$

- **Delta-f codes**: f=f_0 + δf, L= L_0 + δL, $L_0 f_0 = C f_0$

$$L_0 = \frac{\partial}{\partial t} + (v_{\parallel}\mathbf{b} + \mathbf{v}_d)\cdot\frac{\partial}{\partial\overline{\mathbf{R}}} - \mathbf{b}*\cdot\nabla(\mu B)\frac{\partial}{\partial v_{\parallel}}, \qquad \delta L = \mathbf{v}_{E\times B}\cdot\frac{\partial}{\partial\overline{\mathbf{R}}} - (\mathbf{b}*\cdot\nabla <\Phi>_\alpha)\frac{\partial}{\partial v_{\parallel}}$$

  – Start from f_0 and evaluate the deviation using particle weight δf=wf_0 ,

  dδf/dt = - df_0/dt +C(f) = - L_0 (f_0) + C(f_0) + δLf_0 +C(δf)

$$= \mathbf{v}_{E\times B}\cdot\frac{\partial}{\partial r}f_0 - (\mathbf{b}*\cdot\nabla <\Phi>_\alpha)\frac{\partial}{\partial v_{\parallel}}f_0 + C\delta f$$

  – Numerically easier and consume less computing resources than the full-f codes

Conventional full-f code df/dt=Lf=C(f)

$$L = \frac{\partial}{\partial t} + (v_{\parallel}\mathbf{b} + \mathbf{v}_d + \mathbf{v}_{E\times B}) \cdot \frac{\partial}{\partial \overline{\mathbf{R}}} - \mathbf{b}*\cdot\nabla(\mu B + <\Phi>_\alpha)\frac{\partial}{\partial v_{\parallel}}$$

$$\mathbf{B}^* = \mathbf{B} + (Bv_{\parallel}/\Omega_s)\nabla\times\mathbf{b}$$

$$-\nabla^2\phi = \sum_s \left[4\pi e_s \int \overline{\delta f}_s \delta([\overline{\mathbf{R}}+\overline{\rho}_s]-\mathbf{x})\mathscr{J}_s d^6\overline{Z} - \frac{1}{\lambda_{Ds}^2}\left(\phi - \langle\overline{\phi}\rangle_\alpha\right)\right]$$

$\nabla^2\Phi$

- Established in a simple geometry (Dawson, Birdsall, etc, 1968 →)
- Background and turbulent dynamics are simulated together.
- More difficult than delta-f in a toroidal geometry due to neoclassical equilibrium establishment and its interaction with turbulence
- Particle approach
  - Demands more computing time (>100) (particle). no growing weight.
- Continuum approach
  - More difficult numerically. Requires highly accurate conservations.

Common types of PDEs you want to pay attention

- ## Elliptic:
  - Poisson equation

$$\nabla^2 \Phi = \rho$$

- ## Parabolic
  - Diffusive phenomena: eg. Heat conduction

$$\frac{\partial \Phi}{\partial t} = D \nabla^2 \Phi$$

- ## Hyperbolic
  - Wave phenomena

$$\frac{\partial^2 \Phi}{\partial t^2} = c^2 \nabla^2 \Phi$$

PETSc solver library in SciDAC TOPS project (led by D. Keyes) has been set up for this purpose.

Different types of parallelization architecture for different types of problems

- A program process using shared memory
  - Parallelization is achieved through multiple threads (execution instructions)
  - OpenMP share of the memory
  - PDE dominant, mesh based codes (Eulerian continuum)

- Multiple processes using distributed memory
  - Parallelization is achieved through multiple processes on multiple processors (cores)
  - MPI communications between processors for information coupling between processes
  - PDE non-dominant (Lagrangian PIC)

- (Future) Sharing assisted Distributed memory architecture?
  - Many cores per processor node
  - Mixed MPI and OpenMP programming

Shared memory architecture

- **OpenMP** Application Program Interface (API)
  - **Open** standard for providing parallelization mechanisms on shared-memory **M**ulti-**P**rocessors

- Processors communicate through variables in a shared address space
- Easy to program: Referencing data stored in memory is similar to traditional single-processor programs

- Poor scalability: Increased bus traffic slows down the memory access time.

Serial Region

Shared Memory

Serial Region

Parallel for Region
with 4 Threads
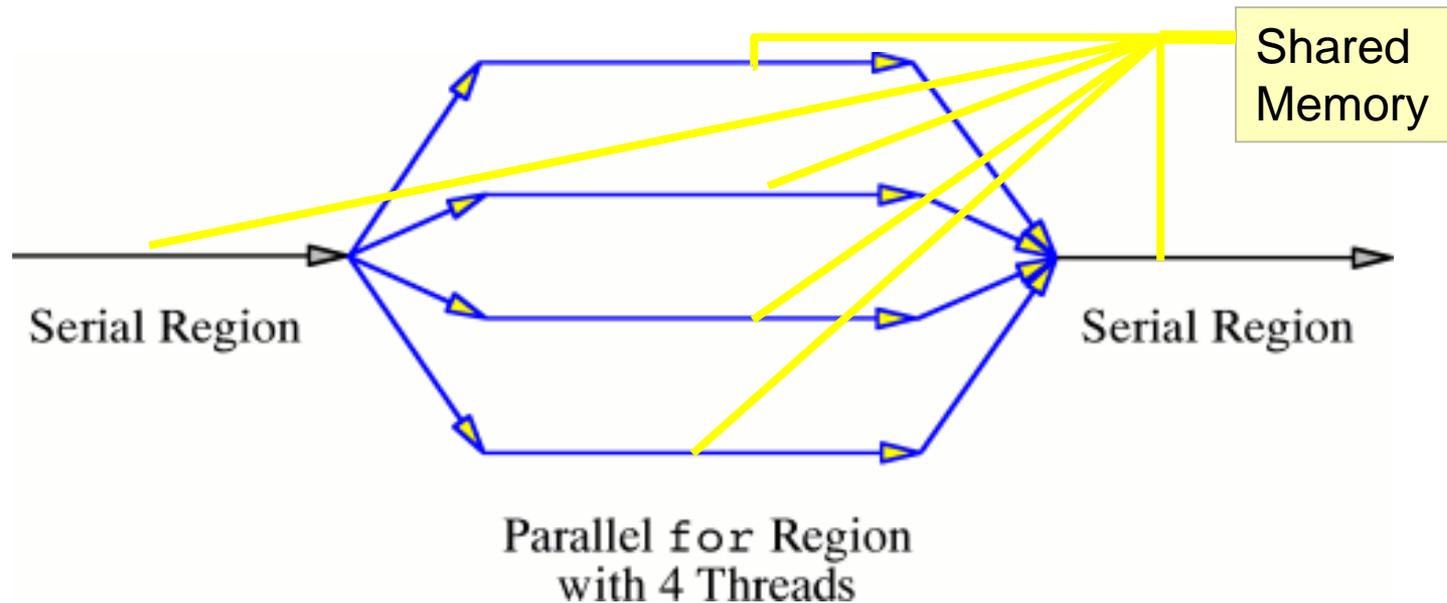
Distributed memory architecture

- **Message Passing Interface (MPI)** for inter-process communication on distributed-memory

- The program spawns into the number of processes
- Easy to build large, inexpensive MPI cluster computers

- Good scalability since each processor has a separate bus with access to its own memory. But requires a high-speed network between processors (infiniband).
- Key: How to distribute problems to multiple processors equipped with limited memory spaces while minimizing the inter-process communication.
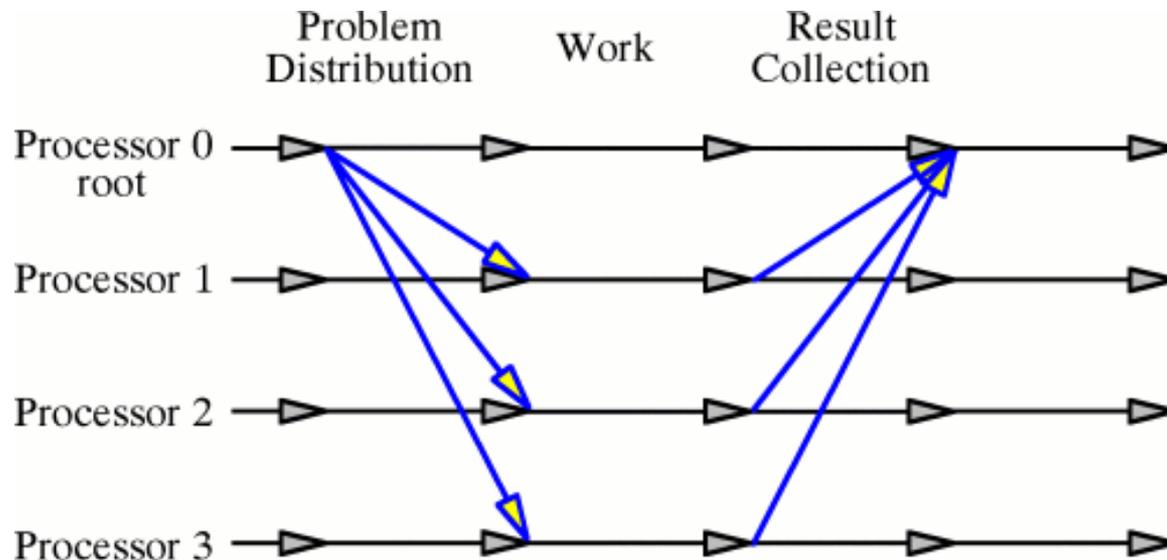
A simple MPI execution model

Excellent scalability of particle codes on distributed memory HPC

XGC1 on Jaguar Cray XT3 multi cores at ORNL
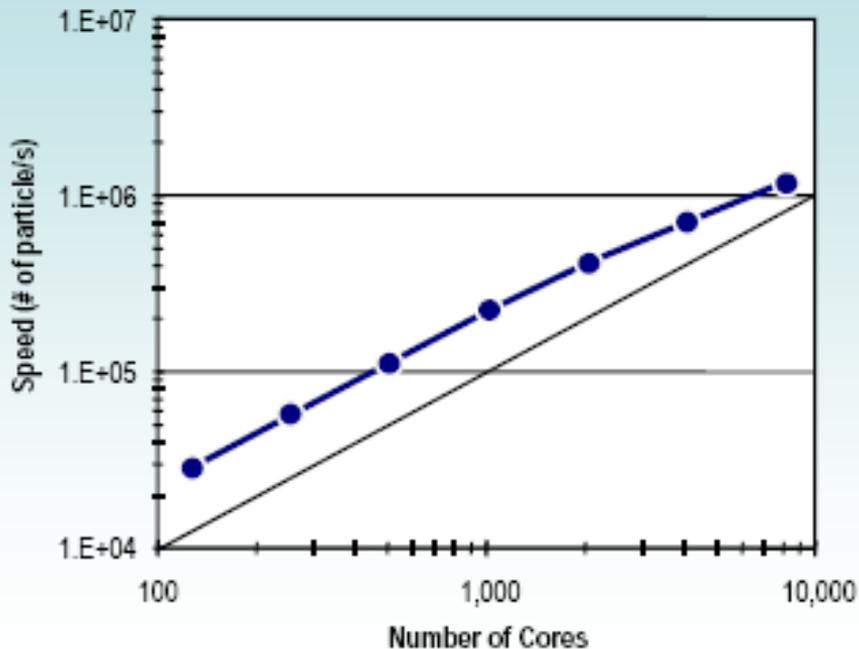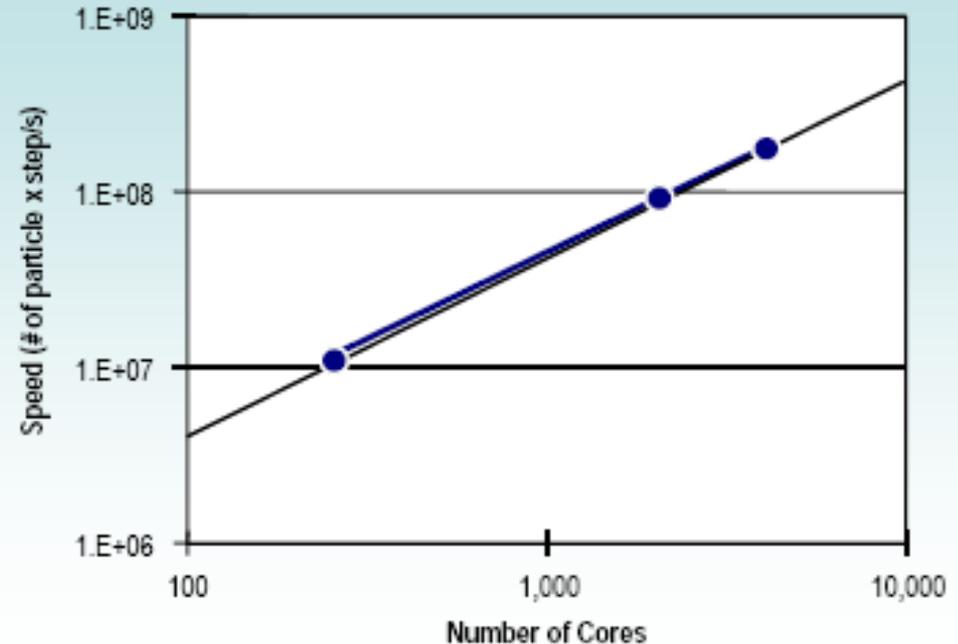
Spring 2007 (S. Ku)

Distribute problems to multiple processors
e.g., **Particle decomposition**

- Grid quantities must be replicated on each processors.
- The larger the grid number is, the more memory is needed.
$\Rightarrow$ Memory limit $\Rightarrow$ Domain decomposition

Domain decomposition
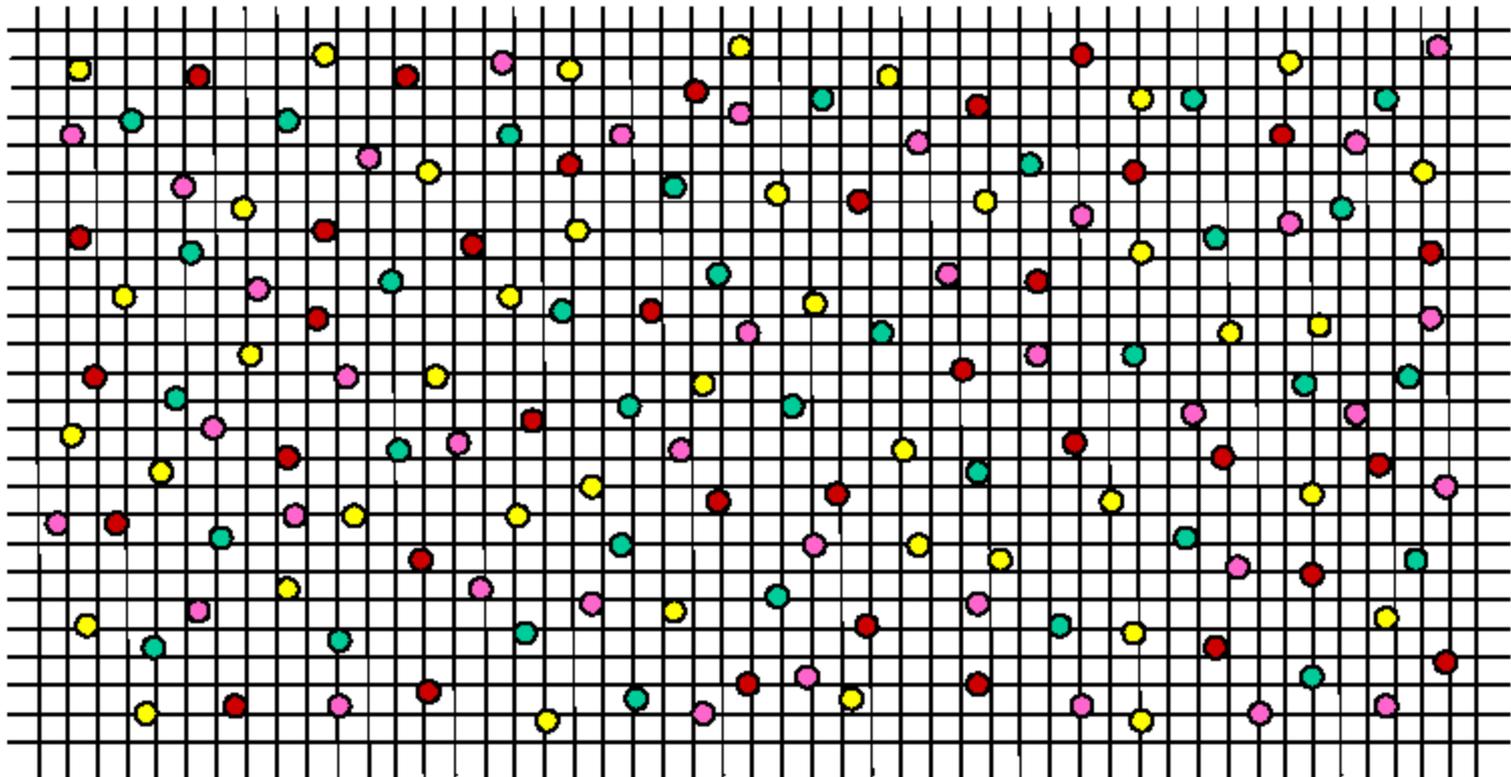
Processor 1   **Processor 2**
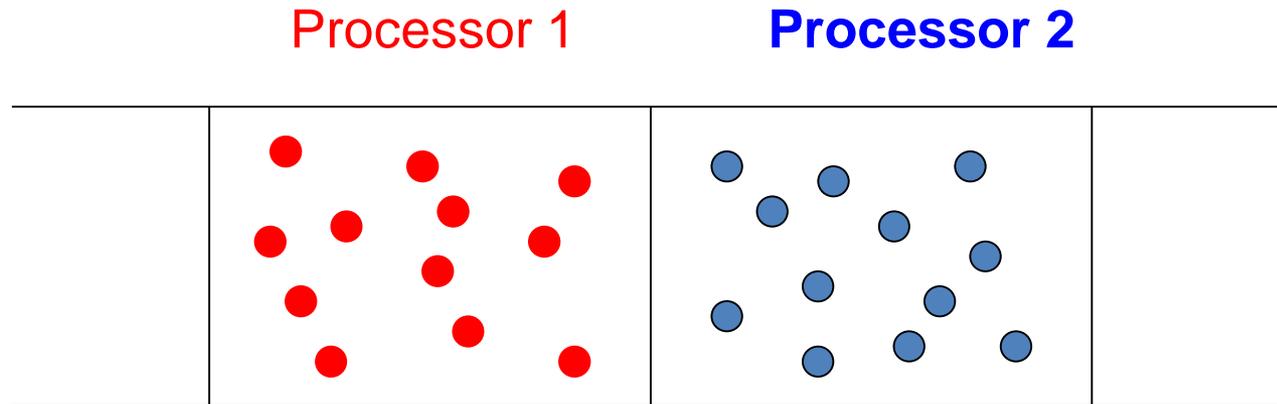
- Divide the physical simulation region into many domains utilizing the geometrical shapes.
- Each processor keeps memory of only one domain.
- Each particle is assigned to a processor according to its position.
- Communication between domains is achieved by MPI calls.
- As a particle moves to a different domain, it moves to the corresponding processor → some complication

Toroidal domain decomposition

- Each MPI process holds a toroidal section
- each particle is assigned to a processor according to its position
- Initial memory allocation is done locally on each processor to maximize efficiency
- Communication between domains is done with MPI calls (runs on most parallel computers)
- The particles move fast along the field lines in both directions, resulting in a natural load balance
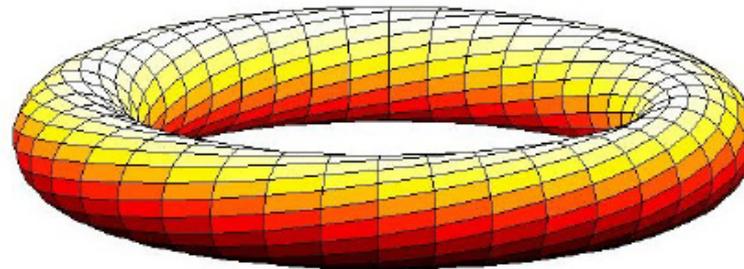
Further enhanced by radial domain decomposition.

Domain and particle mixed decomposition

- Each domain can have multiple processors working in it.
- Each processor holds a fraction of the total number of particles in that domain.
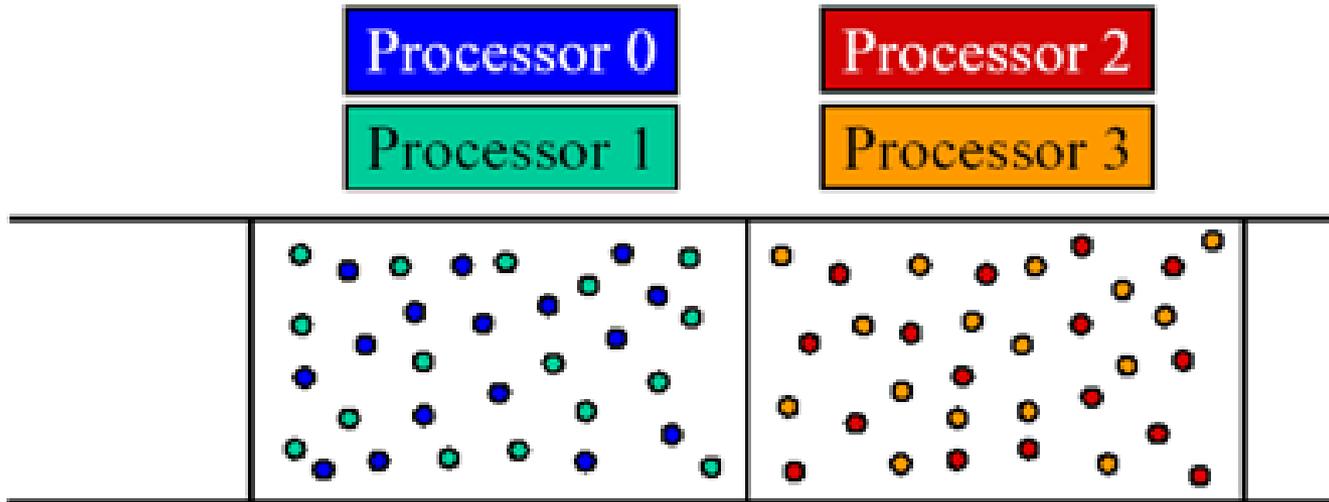
Toroidal plasma simulations at first principle level

- **Klimontovich system of particles**

  $N_s(\mathbf{x}, \mathbf{v}, t) = \Sigma_i \, \delta(\mathbf{x} - \mathbf{X}_i) \, \delta(\mathbf{v} - \mathbf{V}_i)$

  Take the time derivative

  $\partial N_s(\mathbf{x}, \mathbf{v}, t)/\partial t + \mathbf{v} \cdot \nabla_x N_s + \mathbf{a^m} \cdot \nabla_v N_s = 0$: Klimontovich equation

  $\mathbf{a^m} = (q_s/m_s)[\mathbf{E}^m + (\mathbf{v}/c) \times \mathbf{B}^m]$,

  $\mathbf{E}^m$ and $\mathbf{B}^m$ are the self-consistent micro fields from mutual (and external) interactions between all the particles.

  $\rightarrow$ Impossible to handle (10^{20} particles with 10^{20}! interactions)

$\Rightarrow$ Kinetic Theory in statistical mechanics

  Liouville eqn., BBGKY Hierarchy, …

- **Ensemble averaged Klimontovich equation**

Average volume must contain large number of particles, but smaller than Debye sphere.

$N_s(\mathbf{x},\mathbf{v},t) = f_s(\mathbf{x},\mathbf{v},t) + \delta N_s(\mathbf{x},\mathbf{v},t)$

$\mathbf{E^m} = \mathbf{E} + \delta\mathbf{E}, \ \mathbf{B^m} = \mathbf{B} + \delta\mathbf{B}, \ \mathbf{a^m} = \mathbf{a} + \delta\mathbf{a}$

$\partial f_s(\mathbf{x},\mathbf{v},t)/\partial t + \mathbf{v}\cdot\nabla_x f_s + \mathbf{a}\cdot\nabla_v f_s = -< \delta\mathbf{a}\cdot\nabla_v \ \delta N_s >$

↑   ↑

Individual particle statistics   Discrete interaction (collisions)

The collision operation is simplified: Lenard-Balescu eqn → Fokker-Planck operator (Landau and RMJ forms)

f_s , **E, and B** in the left hand side are statistical properties without the N-body interactions between the individual particles → approximation (sampling). Particle trajectories are subject to the macroscopic force only (collective).

⇒ Hopeful

Particle simulation

- Approximate the statistical properties of the Ensemble averaged Klimontivich distribution function using finite number of shielded "marker" particles (10^1 – 10^3 per grid node).

- Randomly sample the distribution function in the phase space

- Follow the Hamiltonian characteristics (equation of motion) of the marker particles.

- Use particle-in-cell (PIC) grid to evaluate the force.

# Velocity Space Resolution

[W. Lee]

Particle code: Random sampling: NxNp=4x5

Nv = 20
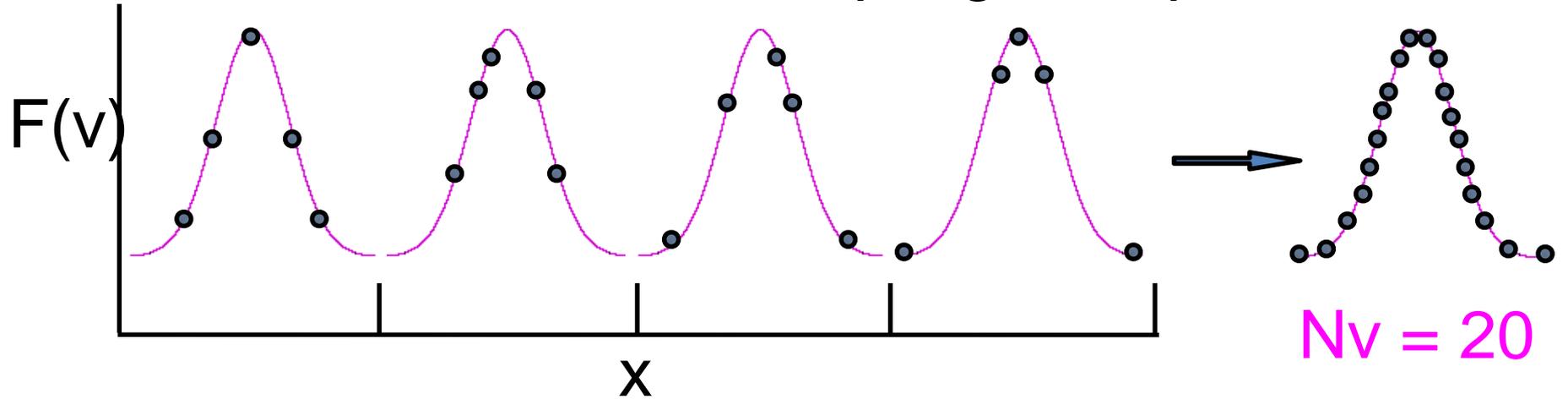
Continuum Code: Grid sampling: Nv=5

Nv = 5
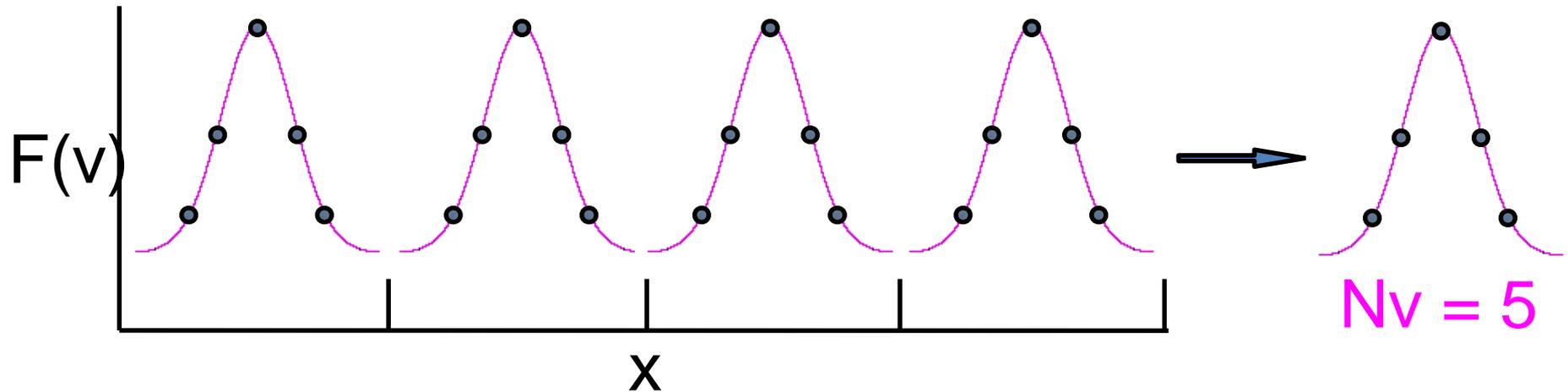
Particle-in-cell technique

- Push the marker particles along the Hamiltonian characteristics.
- Locate the cell
- Scatter particles to the grid
- Solve the force field on the grid (E&M, gravitational, etc)
- Find macroscopic information
- Gather the field at particle positions
- Push the particles
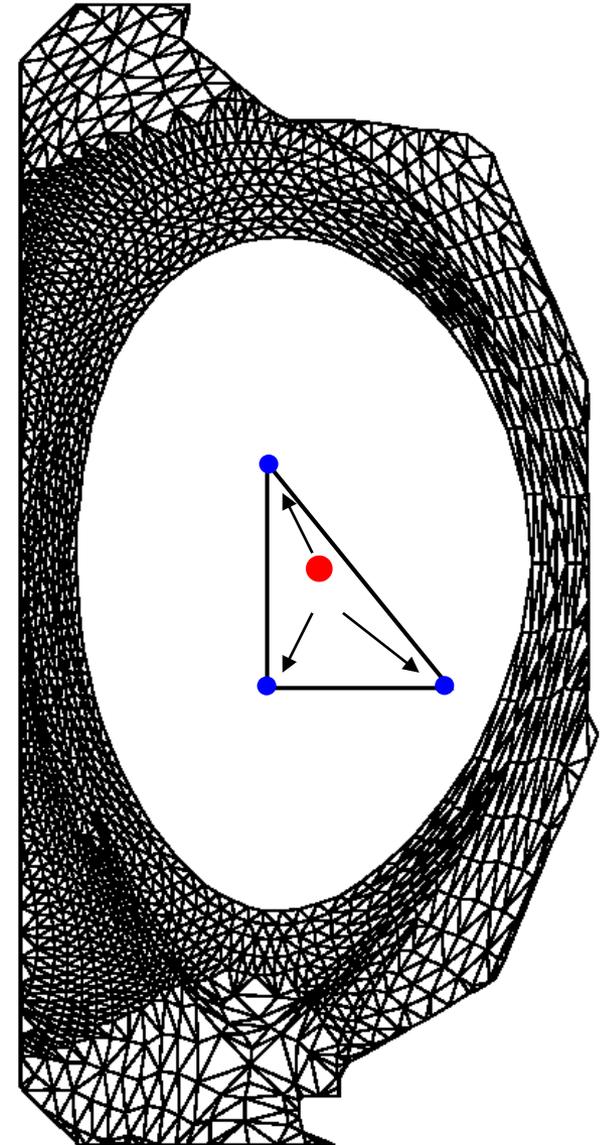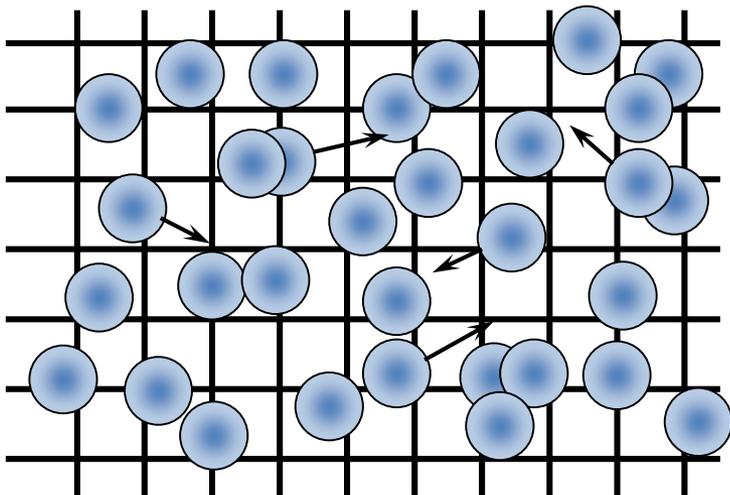
Four point averaged deposition to grid

Charge Deposition Step (SCATTER operation)

GTC
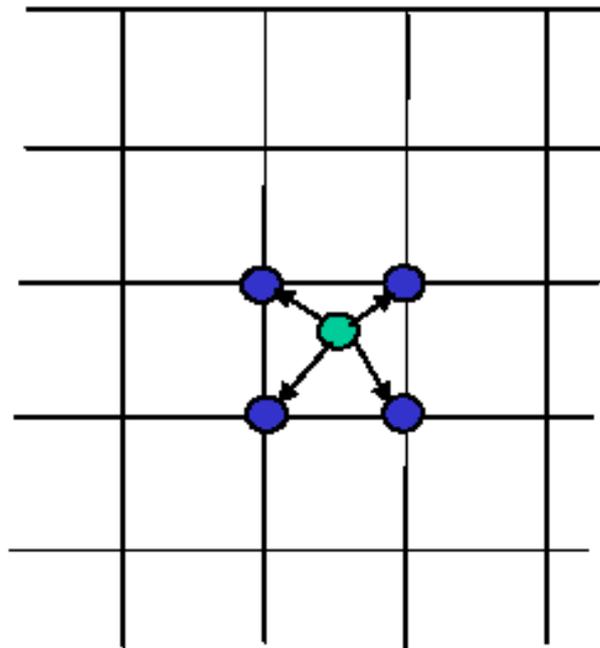
Classic PIC
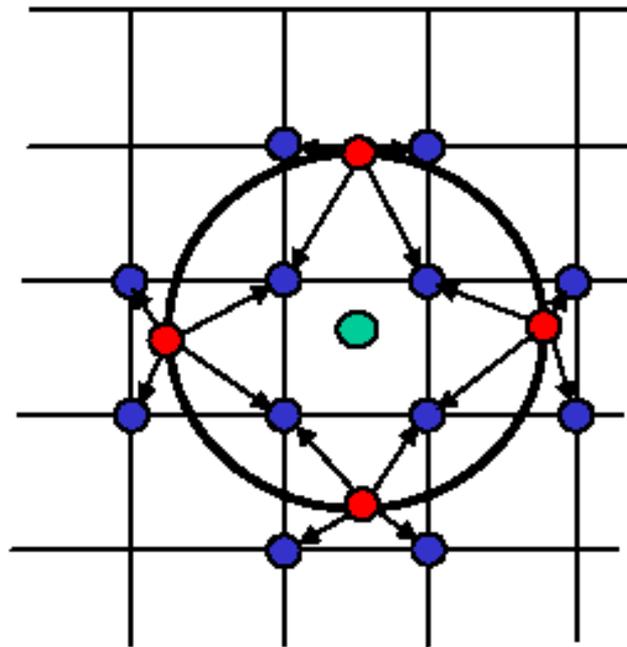
4-Point Average GK
(W.W. Lee)

Continuum kinetic simulation

- Eulerian approach
  - The GK eqns are solved as multi-dimensional partial differential eqns.
  - Particle distribution function is defined on 5D grids in phase space
  - Use the computational fluid dynamics schemes (finite difference, mainly)
- Semi-Lagrangian approach, along the Hamiltonian characteristics

$$df/dt=0, f(z_n, t+\Delta t)=f(z^*, t). \text{ Use the equation of motion to find } z^*$$

$$f(q,p,t)=f(q-(p/m)\ \Delta t, p+e\partial_q\Phi\ \Delta t)$$

  - Mesh grid is kept fixed in time in the phase space (Eulerian approach)
  - The Vlasov equation is integrated along the trajectories (Lagrangian approach) using the invariance of the distribution function along the characteristics.
  - Interpolation to the grid
  - Since the derivative terms are not explicitly evaluated as in a finite difference method (advection), relatively large time step is permitted (not limited by numerical instability but by numerical accuracy).
  - EU (GYSELA) and Japan (GT5D)

Pros and Cons of various gyrokinetic simulation types

Types	Pros	Cons
Radially and toroidally global	Large scale event Toroidal mode coupling	Computationally expensive
Radially global, toroidally wedge	Radial relations	Toroidal mode coupling? (Verifications exist)
Radially local ($\rho/a \to 0$)	Computationally cheaper	Large scale radial event?

Particle (Lagrangian)	Simpler and cheaper with good v-resolution, Easy parallelization	Particle noise
Continuum (Eulerian)	Particle noise is absent.	5D grid can be expensive, Grid dissipation, Small Δt due to Courant stability
Semi-Lagrangian	Particle noise is absent. Large Δt	Not well explored yet.

Pros and Cons of the full-f and delta-f methods

Methods	Pros	Cons
Full-f	Simulations of all scales. Non-equilibrium physics	Computationally expensive and difficult
Delta-f	Computationally cheaper and simpler	• 0th order scale is not simulated. • Difficult to implement sources and sinks. • Noise growth problem → long time simulation?
Mixed-f	May efficiently simulate large scale and/or non-equilibrium ion physics, and small scale electrons physics together.	Unexplored

Full-f ITG simulation in global cyclone geometry

Full-f gyrokinetic turbulence simulation in toroidal geometry is difficult due to strong neoclassical effects (no scale separation).

1. Grad-B drift creates an up-down asymmetric potential first.
2. Neoclassical equilibrium is obtained.
3. Streamers and zonal flows develop together (quasi-linear growth).
4. Streamers are torn apart.
5. Nonlinear stage proceeds.

To verify convergence, we ran with up to15B particles on 10,240 cores for ~20 hrs.
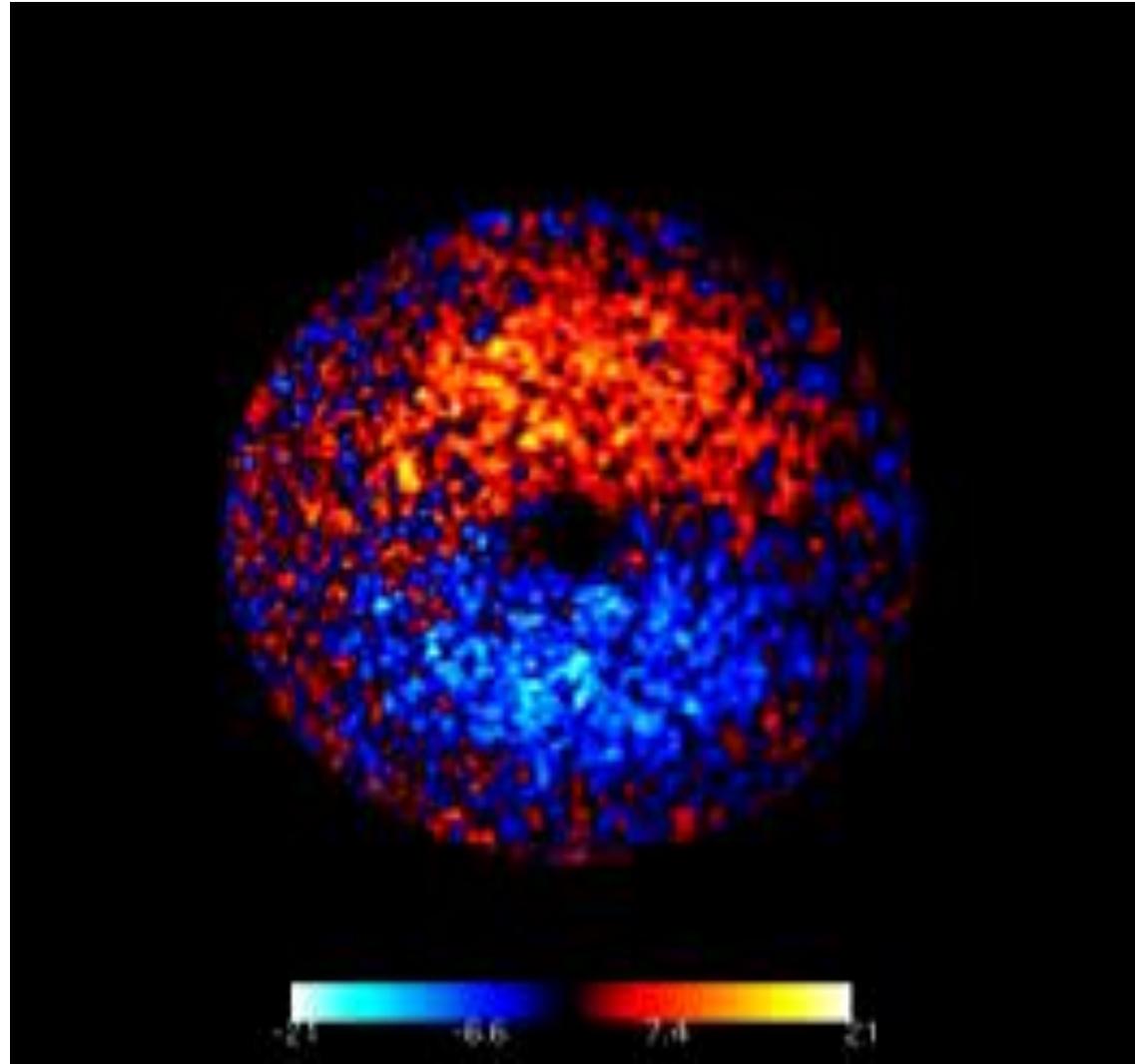
η_i reaches marginal stability during nonlinear ITG turbulence

Before ITG onset

After nonlinear Saturation

End of simulation
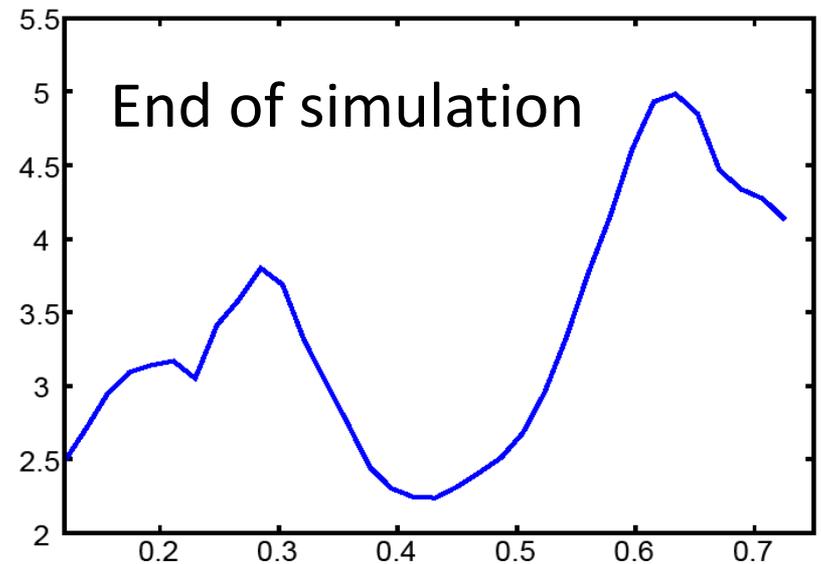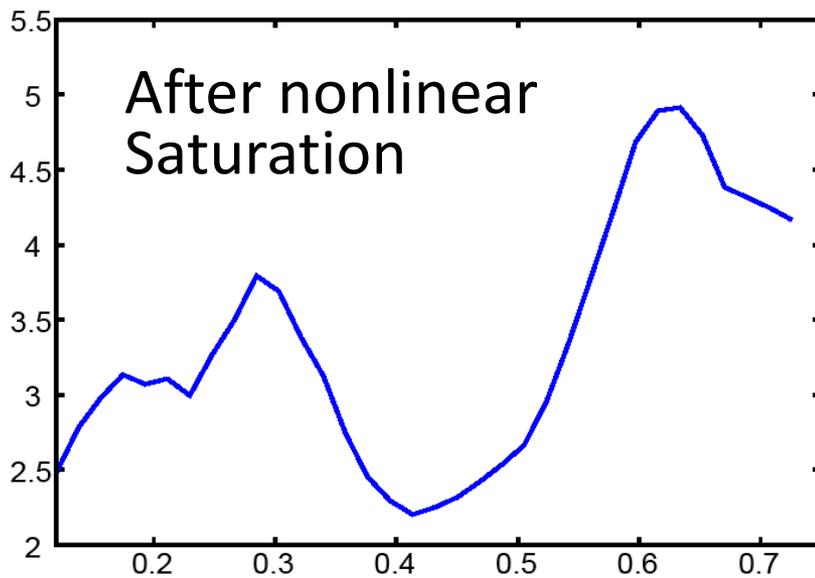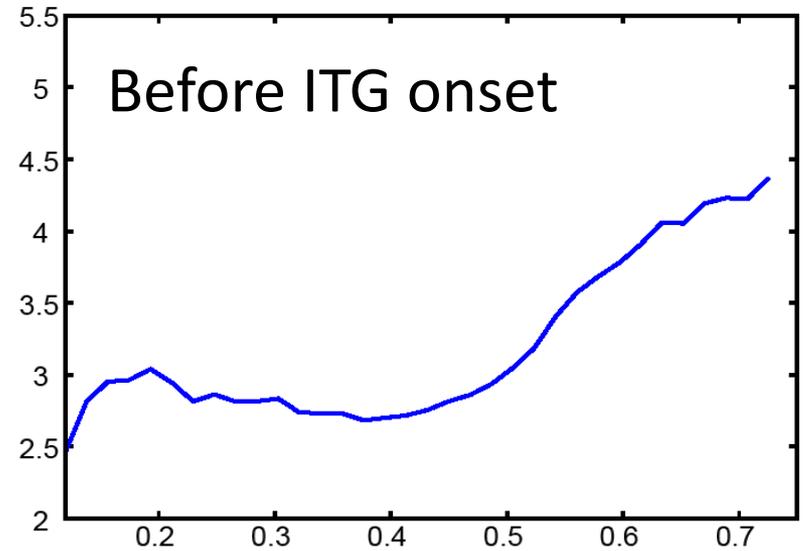
Full-f ITG turbulence across a pre-transition L-mode pedestal in the real geometry edge
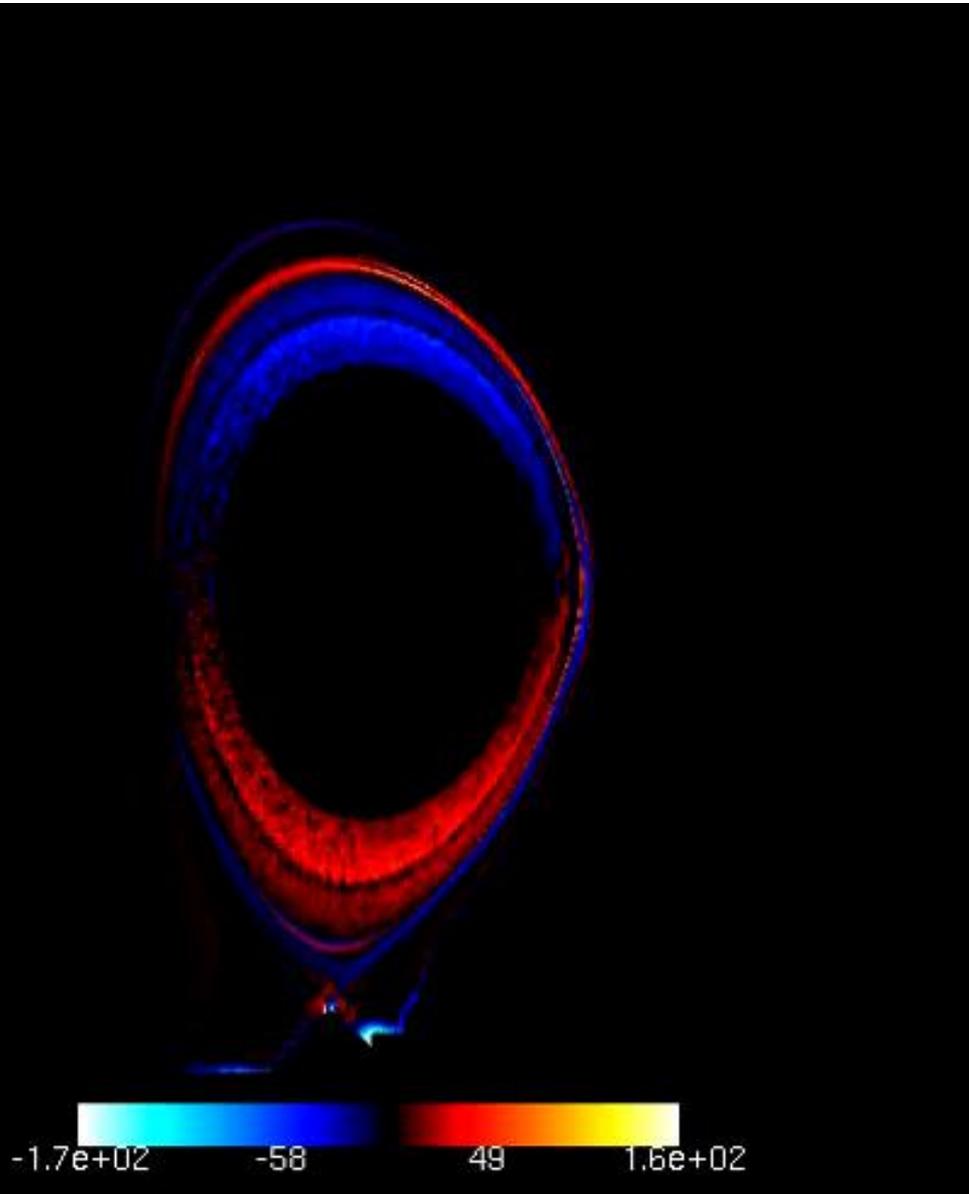
$\eta_i = d\,log\,T_i/d\,log\,n > 3$ at pedestal top (unstable), but $\eta_i < 2$ in the slope (stable)
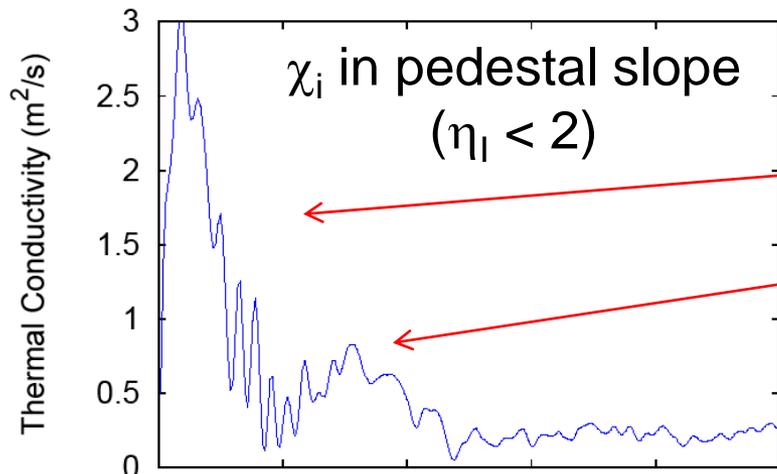
3.2B particles on 8,192 cores for ~10 hours

Plasma profiles
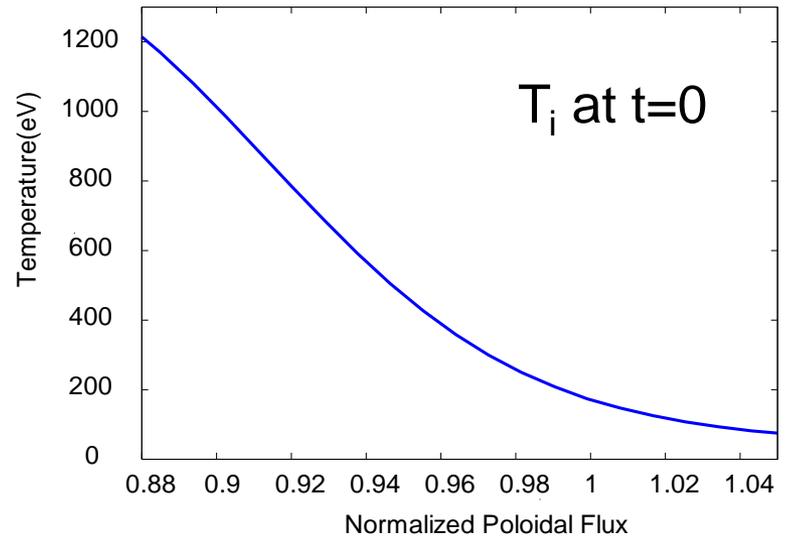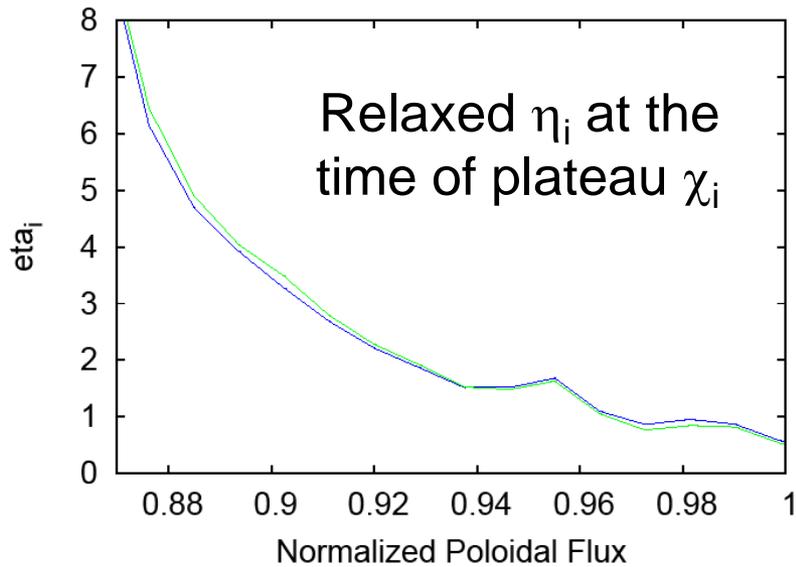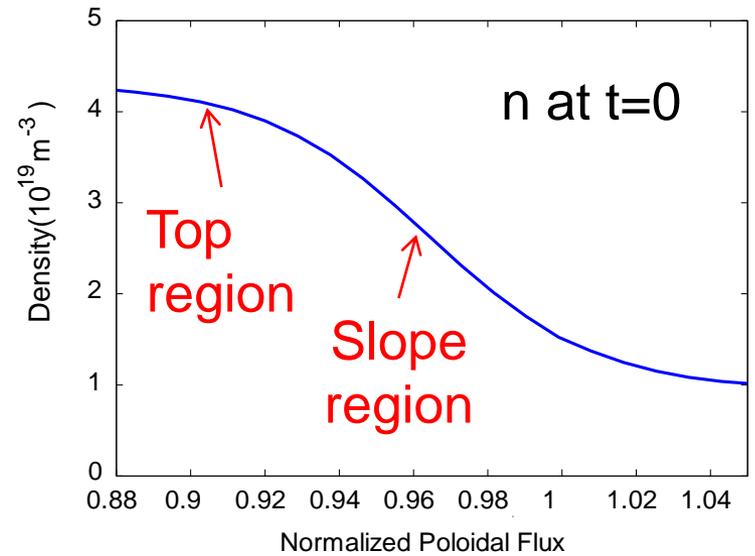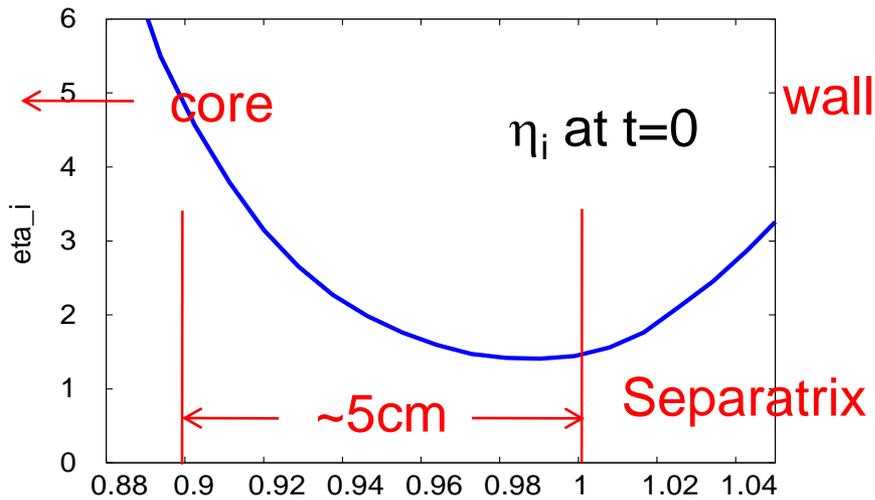
Ion thermal conductivity behavior in time

- Collisionless
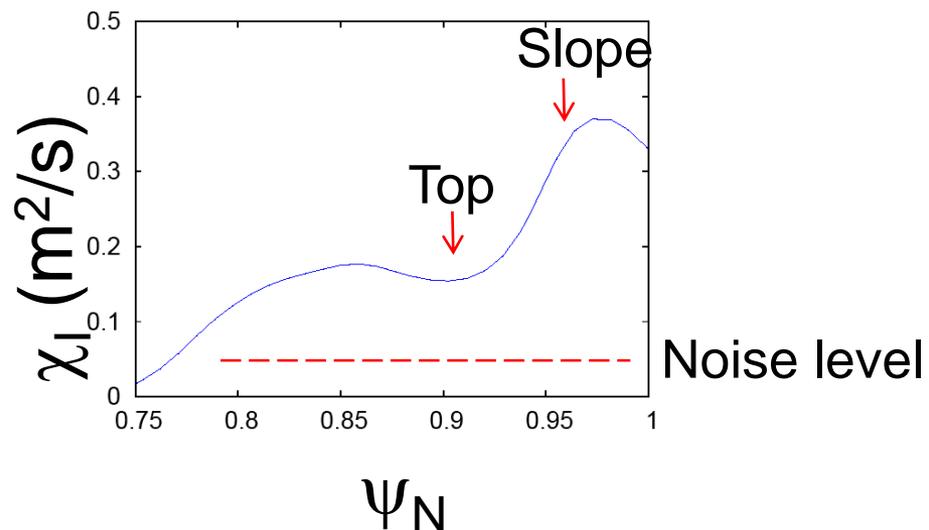- 3.2 billion particles
  (3,500 particles per node)

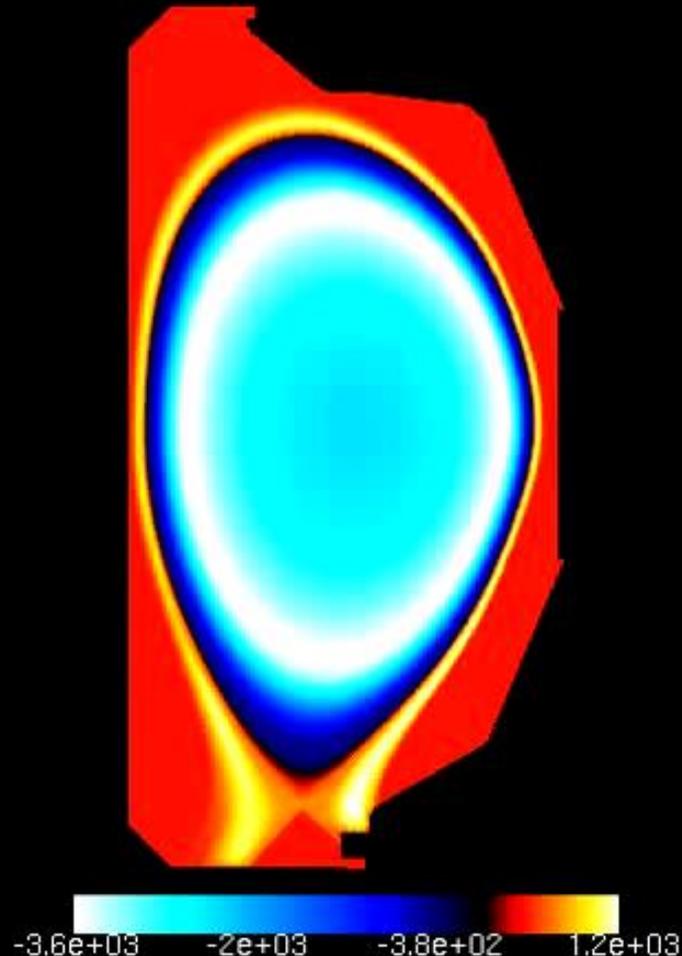Neoclassical settling down

ITG growth and settling down

χ_i in pedestal slope ($\eta_I < 2$)

χ_i at pedestal top ($\eta_I > 3$)

Slope

Top

Noise level
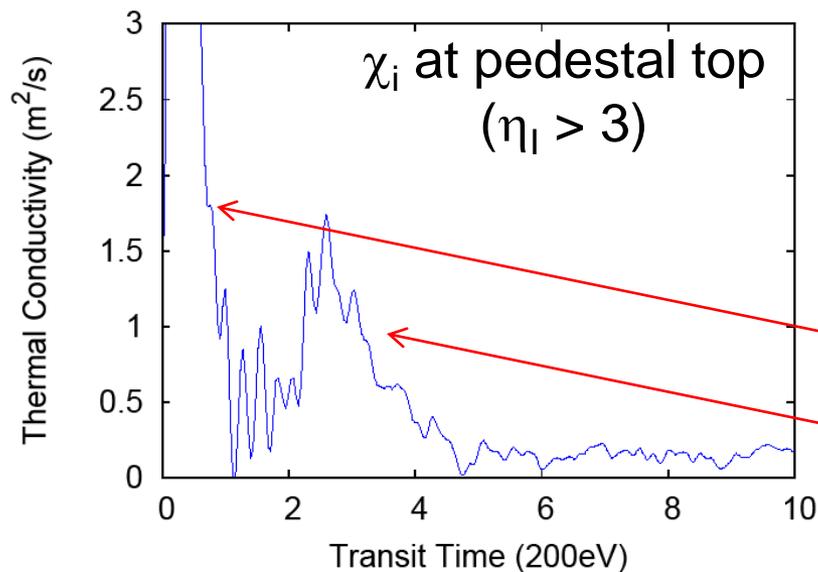
Neoclassical settling down

ITG growth and settling down

<Φ> oscillations in the edge pedestal

-3.6e+03   -2e+03   -3.8e+02   1.2e+03

- Nonlocal GAM oscillations over the whole pedestal
- $\omega_* \sim \omega_{GAM}$
- Nonlocal interaction of turbulent energy through GAM

Movie of axisymmetric, flux-surface-averaged potential dynamics in pedestal

**EXB particle motions
in the poloidal
plane (Steep pedestal)**
→ large scale convective
ExB particle motions in the
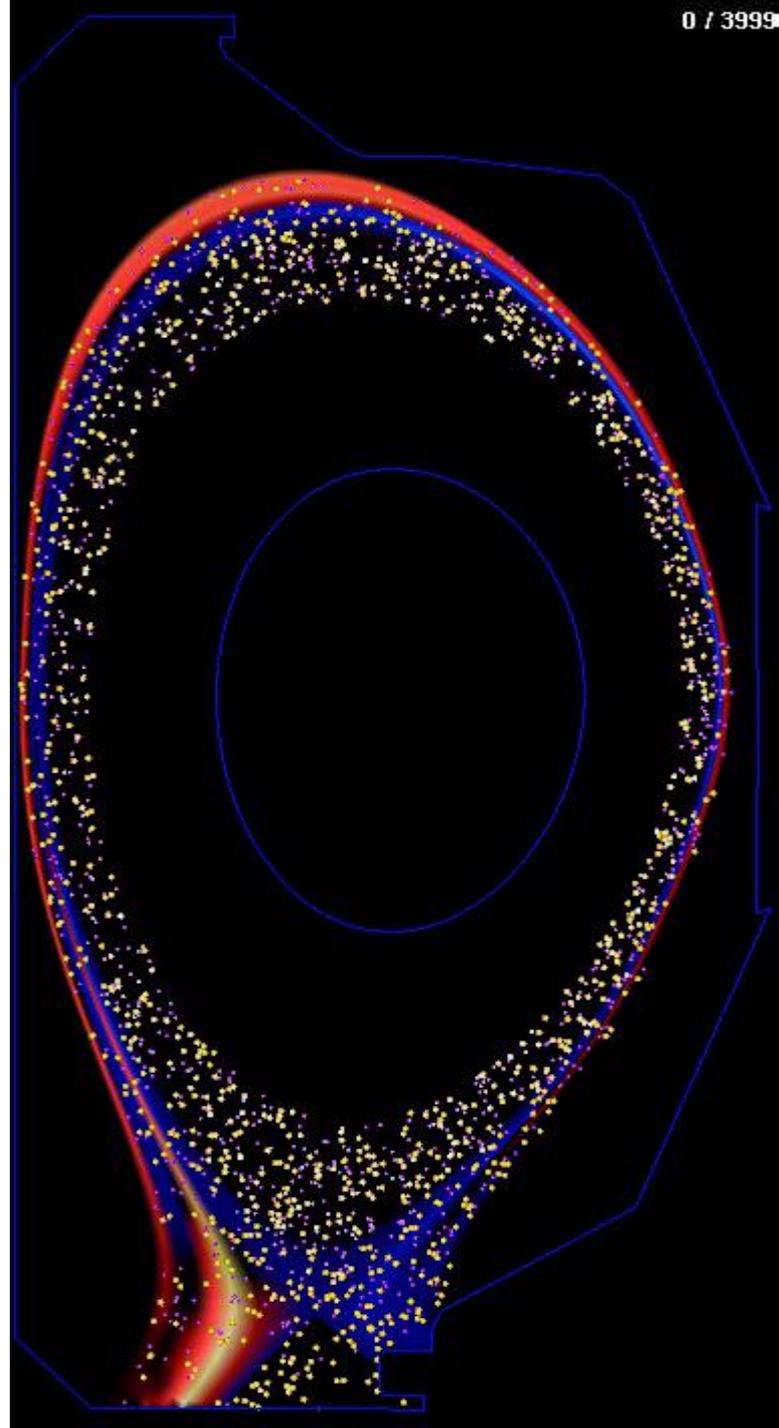scrape-off layer

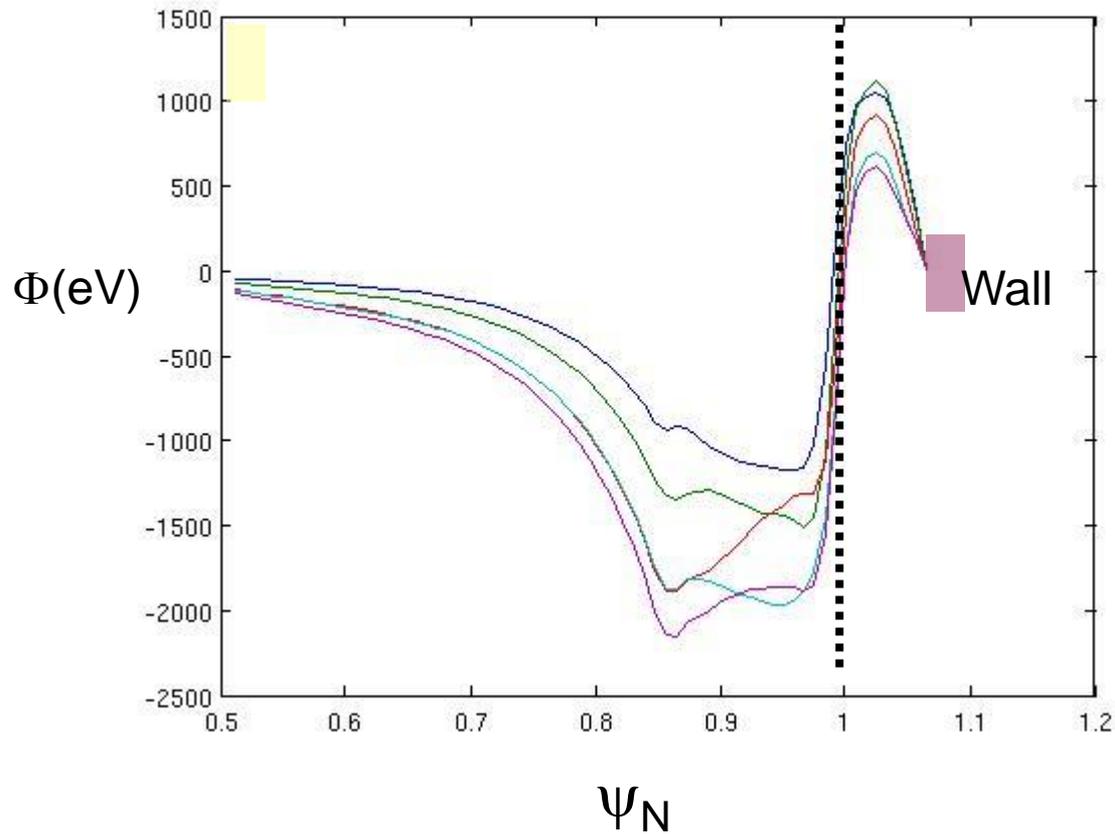ions

electrons

$V_{\parallel} > 0$

$V_{\parallel} < 0$

S. Ku, C.S. Chang, et al., J. Physics:
Conference Series 46, 87 (2006)
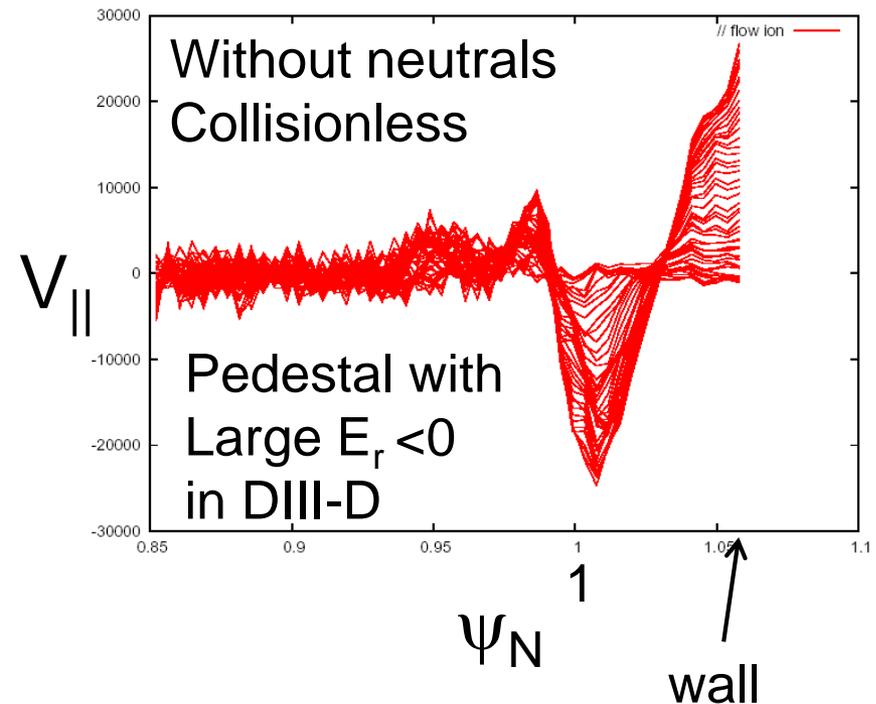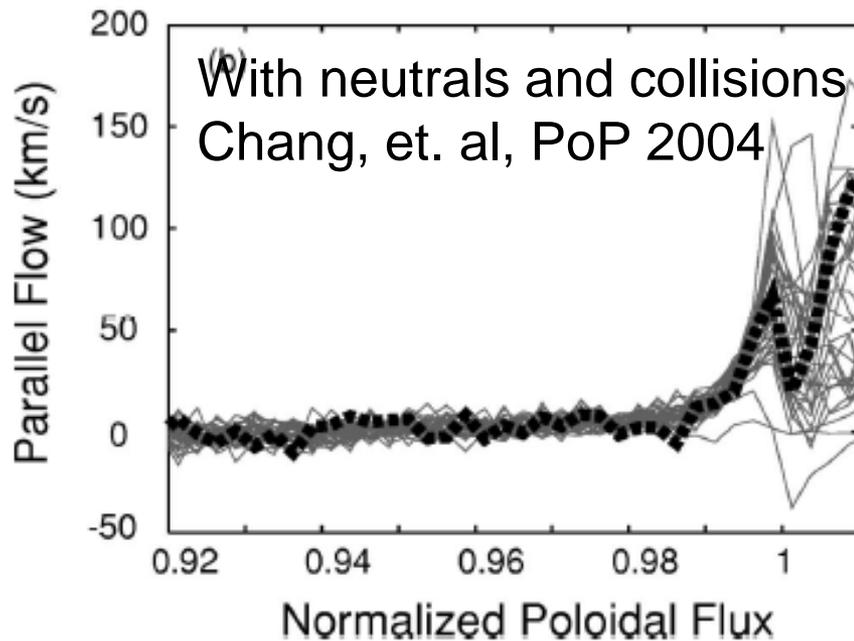
Strongly sheared mean ExB profile in the entire edge

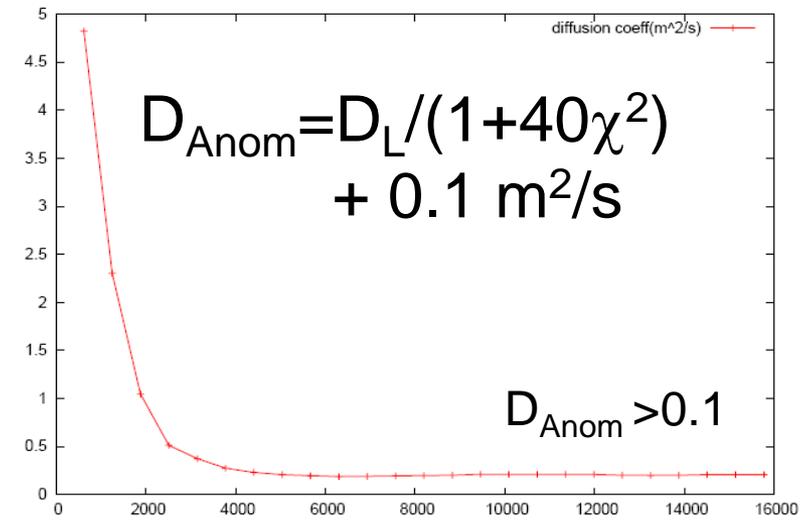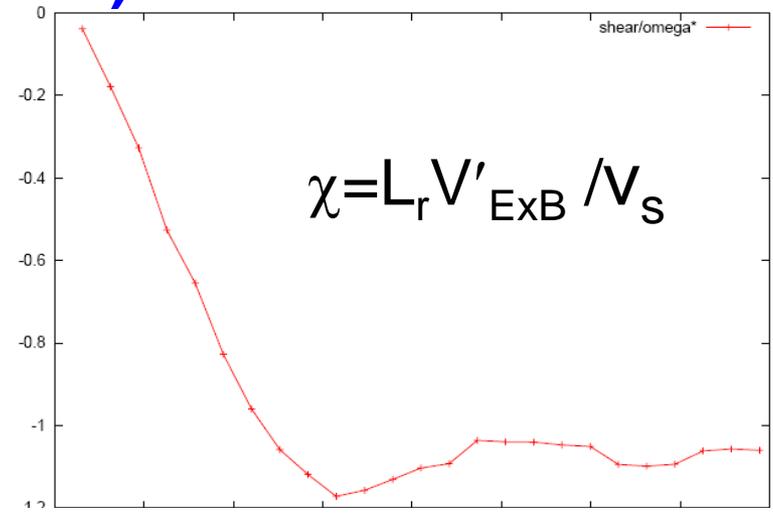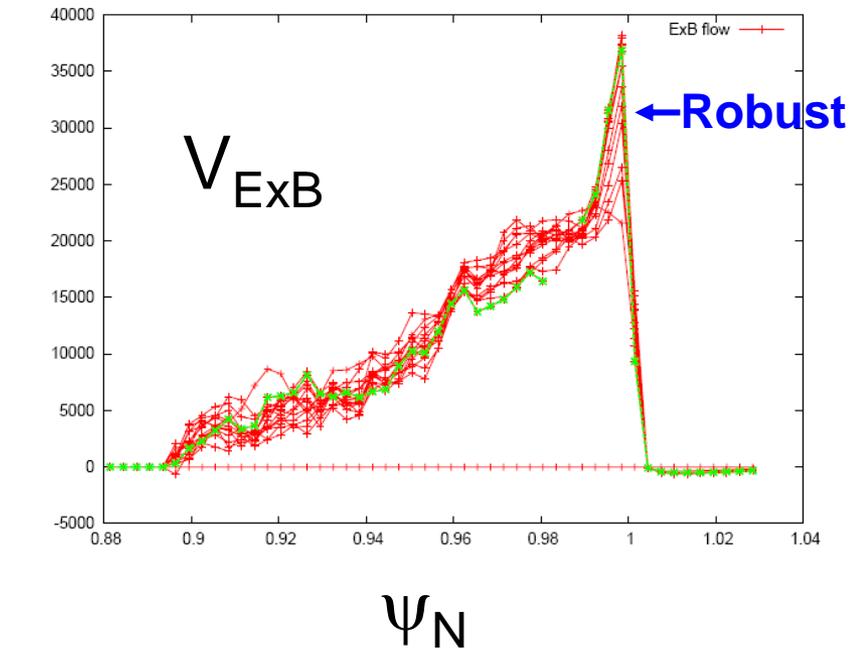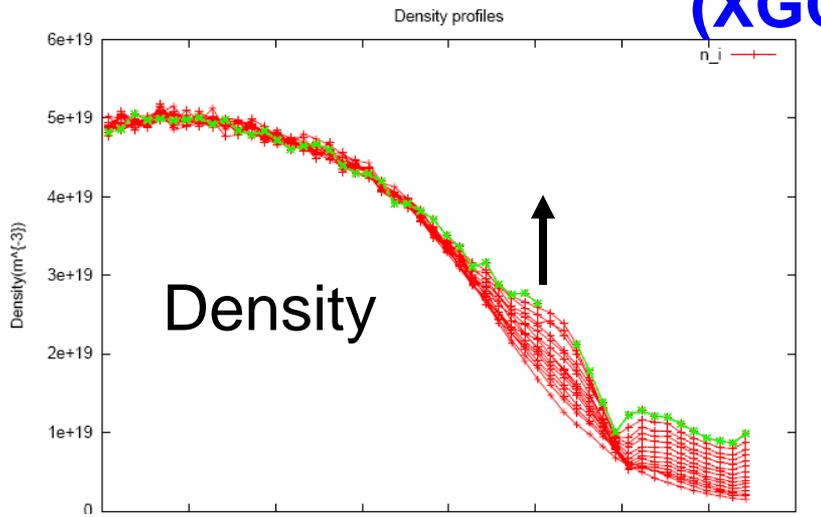# Spontaneous toroidal rotation in the edge

Chang, et. al, PoP 2008

Rotation eqn in axisymmetric core is not valid in edge

$$u_{i\parallel} = (cT_i/eB_p)[k\,d\log T_i/dr - d\log p_i/dr - (e/T_i)d\phi/dr]$$

With neutrals and collisions
Chang, et. al, PoP 2004

Without neutrals
Collisionless

$V_\parallel$

Pedestal with
Large $E_r < 0$
in DIII-D

// flow ion

ψ_N

1

wall

Neutrals fill up the local valley

Reduced model kinetic codes can be used to study L-H transition in self-consistent mean plasma dynamics. (XGC0, 2006)
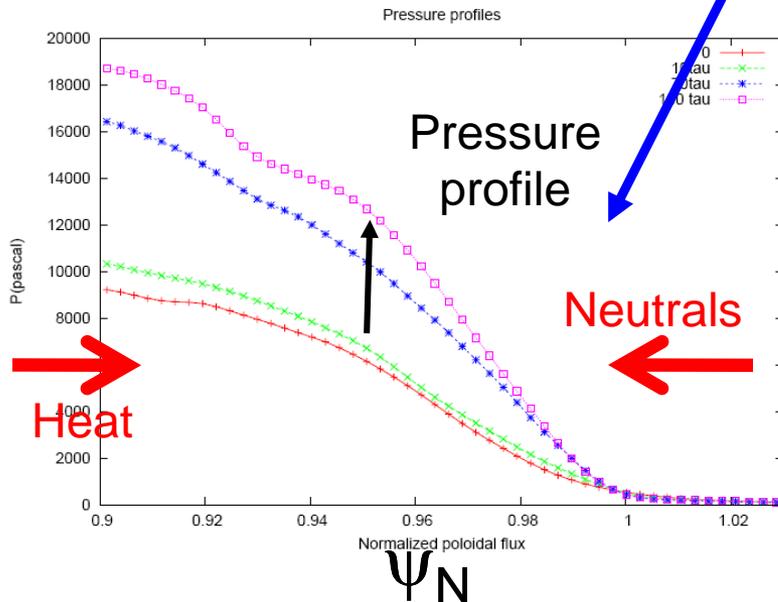

Density

V_{ExB}

←Robust

$\chi = L_r V'_{ExB} / v_s$

$D_{Anom} = D_L / (1 + 40\chi^2) + 0.1 \ m^2/s$

$D_{Anom} > 0.1$

ψ_N

Simulation time steps
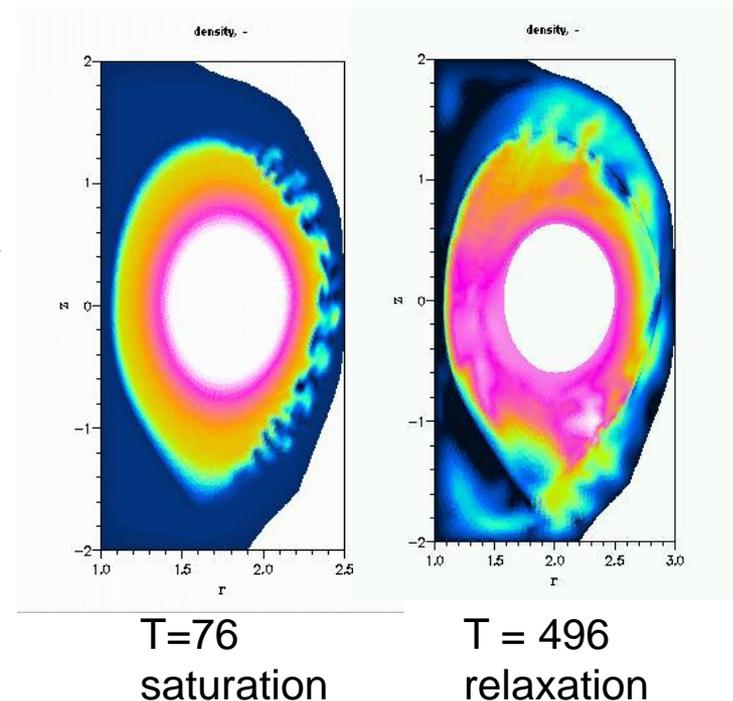
An example of Integrated Simulation
Integrated XGC0/M3D-OMP/Elite/M3D-MPP simulation of ELM cycle in automated EFFIS

Linear stability check (**Elite**)

Pedestal buildup in **XGC0** with B-reconstruction by **M3D-OMP**

Pressure profile

Neutrals

Heat

ψ_N

ELM density crash to relaxation in resistive **M3D-MPP**

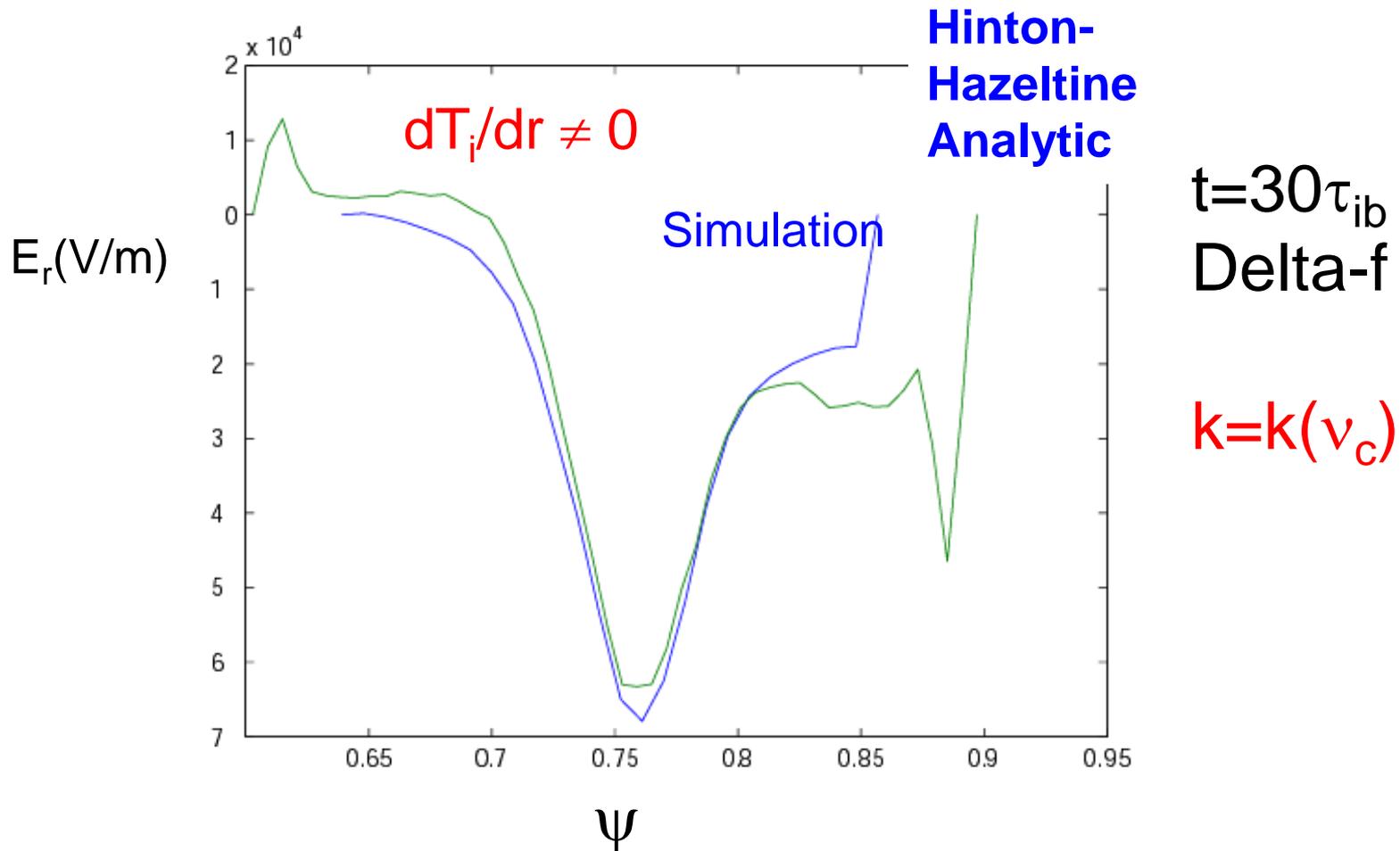T=76 saturation

T = 496 relaxation

Code (solution) verification

Are the computational model equations solved correctly and accurately? Verification deals with mathematics.

1. Numerical studies of convergence rates
2. Monitoring of physically conserved quantities
3. Benchmarking with other codes
4. Comparing with analytical solutions
5. Method of manufactured solution
   1. $L(f)=0$
   2. Manufacture an analytic solution by introducing a special source S, $L(h)=S$
   3. Obtain exact (semi) analytic solution to the equation $L(f)=L(h)$ and check how true f=h is from the code.
   4. We can choose S to test various numerical properties of the code.
   5. This method can provide systematic documentation of the actual rates of convergence, estimates of the computational error, and resolution requirements for different code properties.
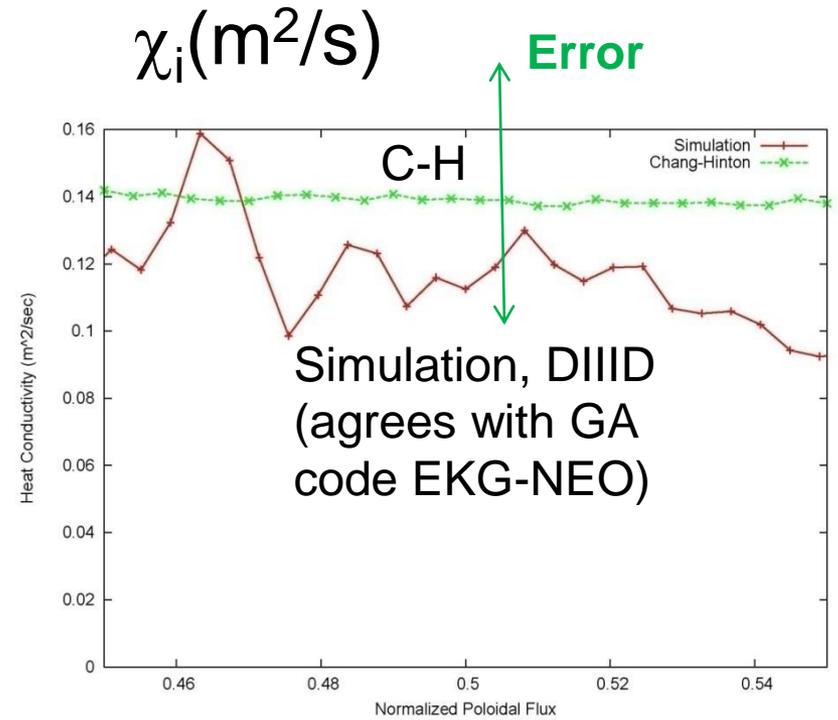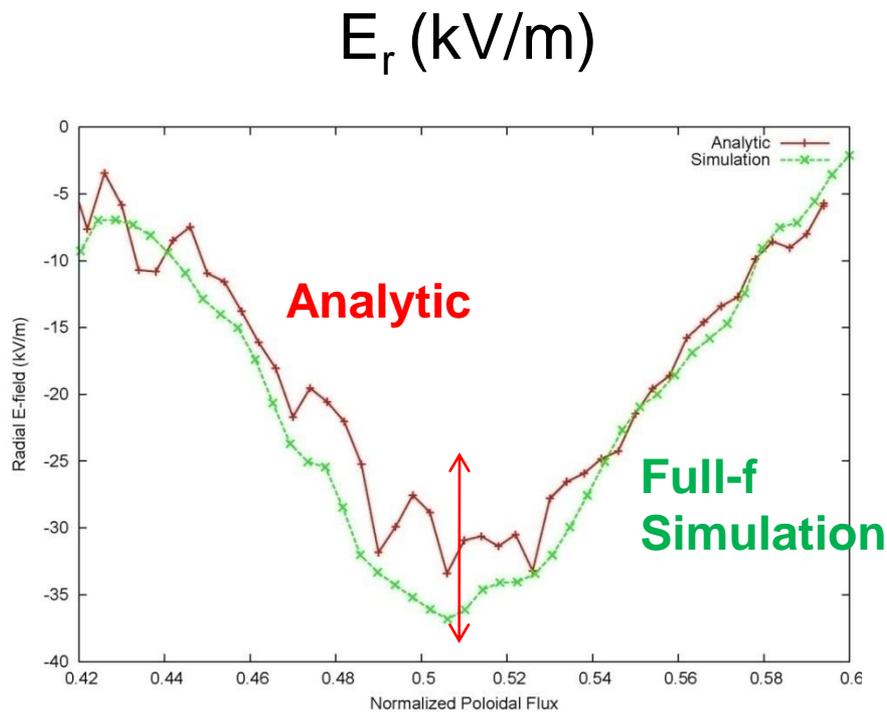
XGC1 verification against analytic neoclassical poloidal rotation in core

$$u_{i\parallel}= (cT_i/eB_p)[k\ \mathrm{d}log\mathrm{T}_i/\mathrm{dr} -\mathrm{d}log\ \mathrm{p}_i/\mathrm{dr}-(e/T_i)\mathrm{d}\phi/\mathrm{dr}]$$

$\mathrm{dT}_i/\mathrm{dr} \neq 0$

Hinton-Hazeltine Analytic

Simulation

$E_r(V/m)$

ψ

$t=30\tau_{ib}$
Delta-f

$k=k(\nu_c)$

Full-f XGC1 neoclassical verification

[Chang and Ku, TTF/ECC 2007]

E_r (kV/m)

χ_i (m^2/s) **Error**

Analytic

Full-f Simulation

C-H

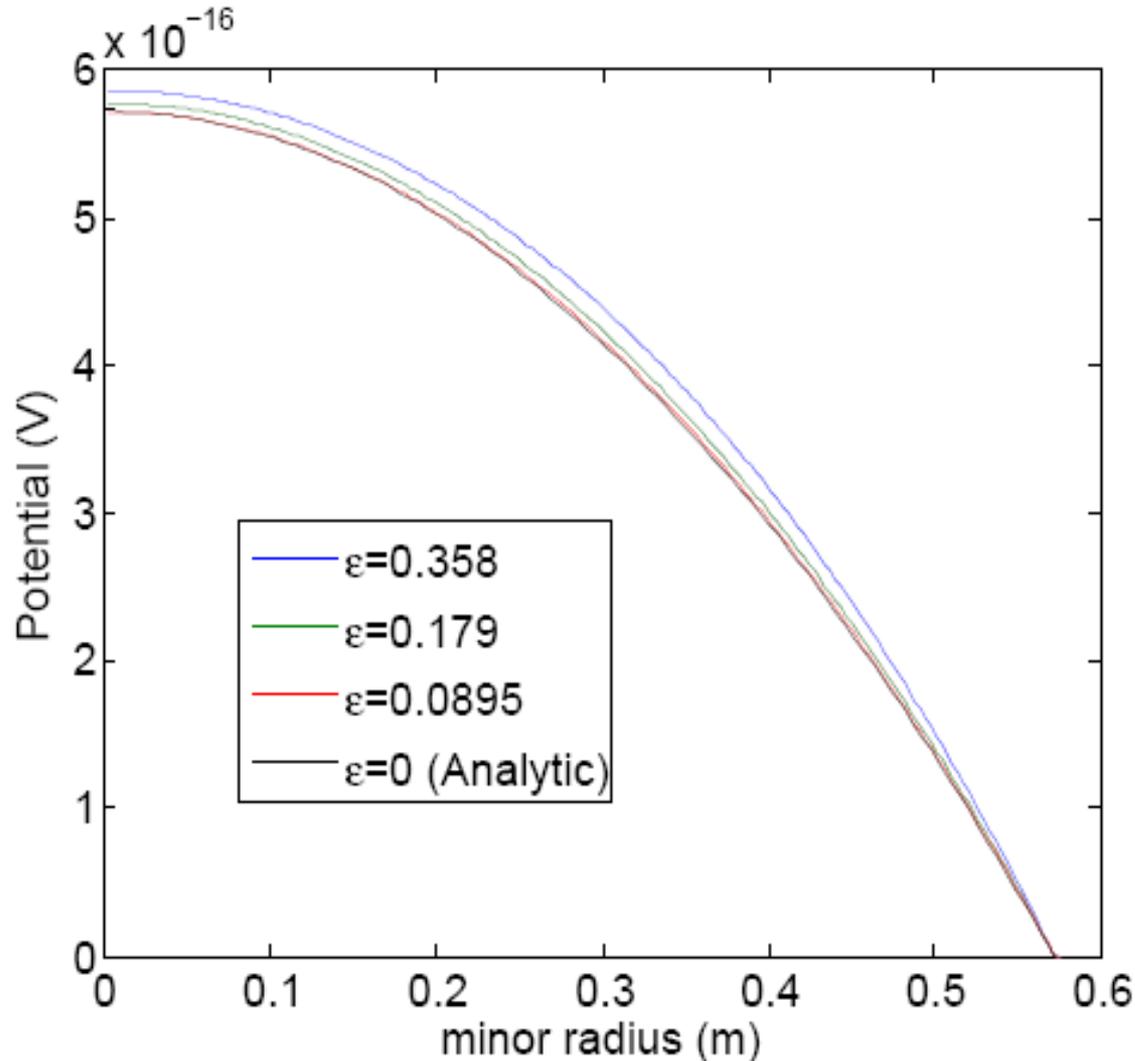Simulation, DIIID (agrees with GA code EKG-NEO)

ψ_N

ψ_N

Banana   Plateau   collisional

Example of the manufactured solution method

Verification of the FEM Poisson solver in XGC1 gyrokinetic code

- Circular concentric tokamak geometry
- m=n=0 mode
- $(\rho_i^2/T) \nabla_\perp^2 \Phi = -S$
  S=constant
  Φ_A is an exact analytic solution at $\varepsilon=r/R_0=0$.
- $(\rho_i^2/T) \nabla_\perp^2 \Phi = -S$ from code should approach Φ_A as ε becomes small with $\mathcal{O}(\varepsilon^2/2)$ error.

Code (model) validation

Are the "models" accurate representation of the real world? Validation deals with physics experiments.
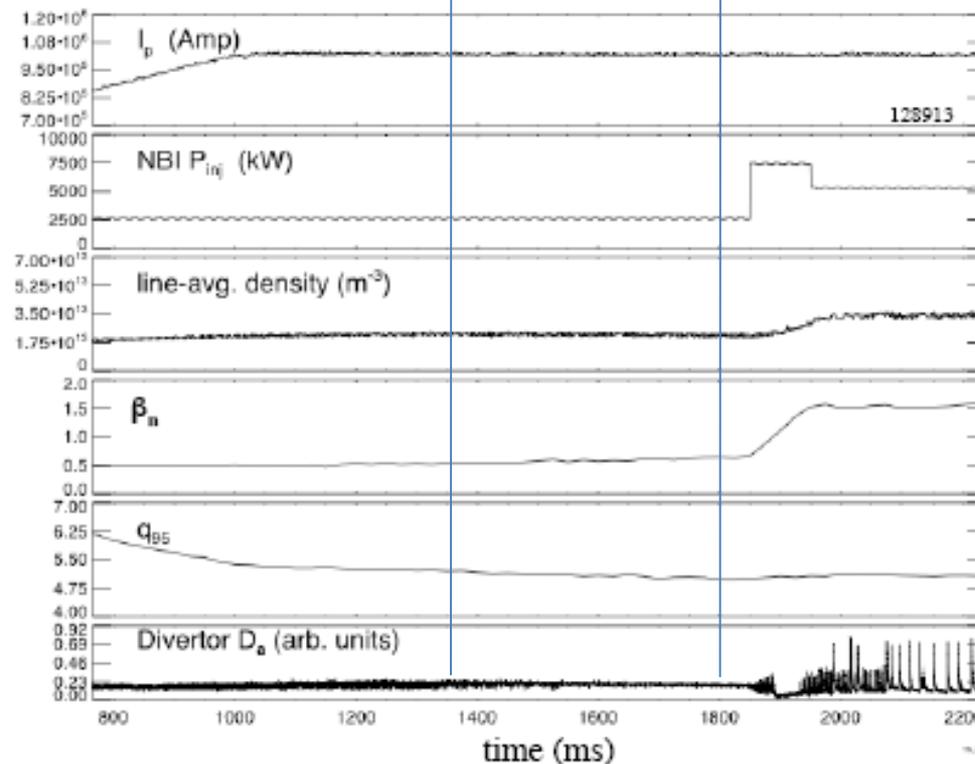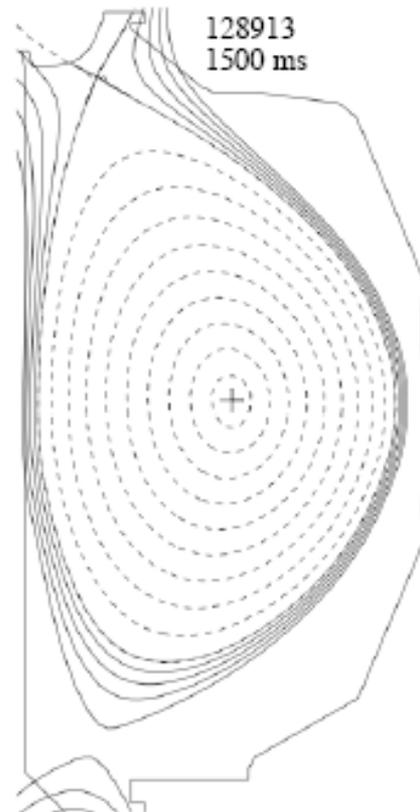
1. The "models" include the equations and the solving conditions.
2. More meaningful after verification
3. Should include the observables at all hierarchical level, if possible.
4. New experiments may need to be designed.
5. Synthetic diagnostics is another issue for meaningful validation
   - What does the experimental diagnostics measure in space and time?
   - What are the uncertainties, how do we minimize them?
6. Validation metric may be set up to quantify the success of a model
   "A formula for objectively quantifying a comparison between a simulation result and experimental data. The metric may take into account errors and uncertainties in both sets of data as well as other factors such as the primacy of the quantities being compared."

[P. Terry *et al., "Validation in Fusion Research: Towards Guidelines and Best Practices*," US-EU TTF 2007, to appear in Phys. Plasmas.]
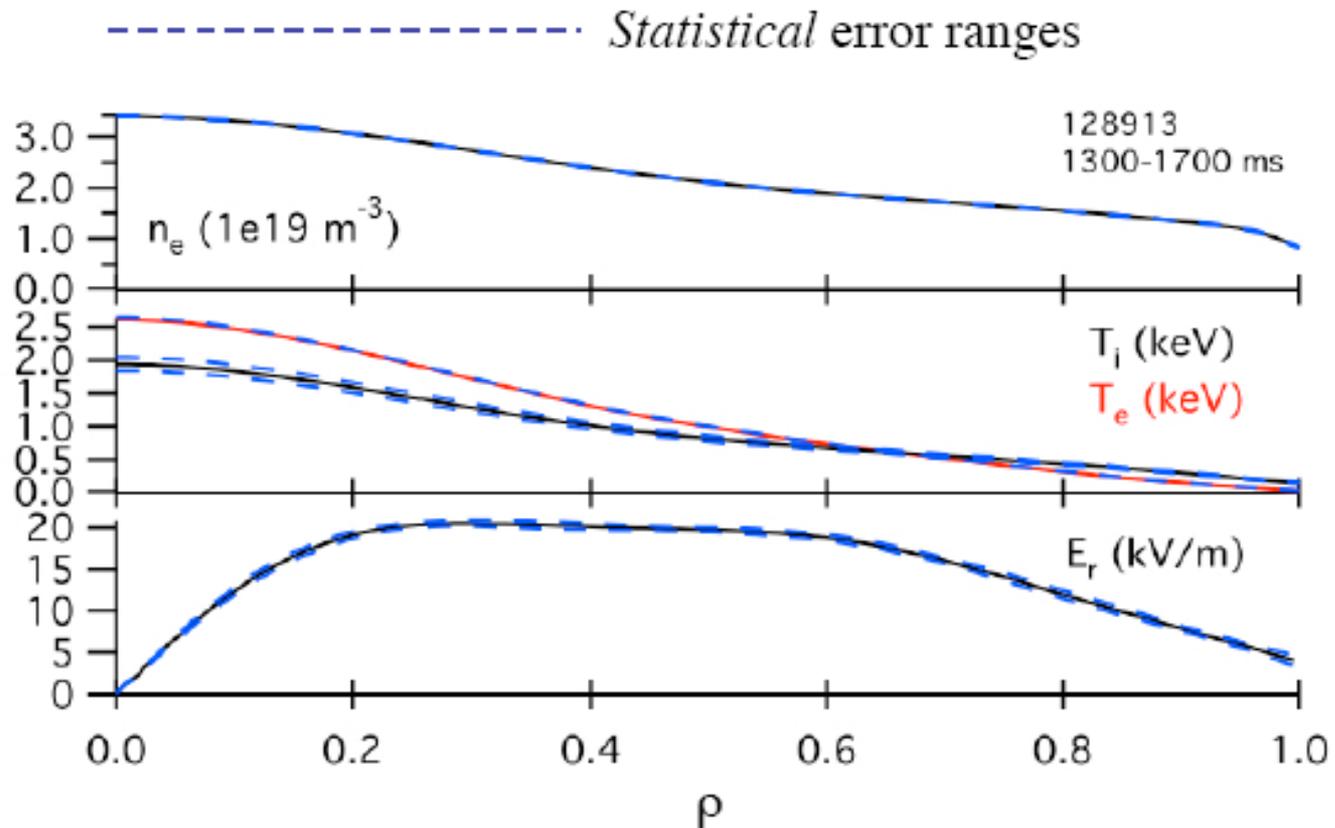
How do we minimize the uncertainties?
Is it a steady state code?

Use 400 ms of Steady L-mode Data for This Exercise

128913
1500 ms

I_p (Amp)

128913

NBI P_{inj} (kW)

line-avg. density (m^{-3})

β_n

q_{95}

Divertor D_a (arb. units)

time (ms)

(Courtesy of C. Holland)

Statistical error matters in the validation

Use of Steady Discharge Allows Minimization of *Statistical* Uncertainty

(Courtesy of C. Holland)
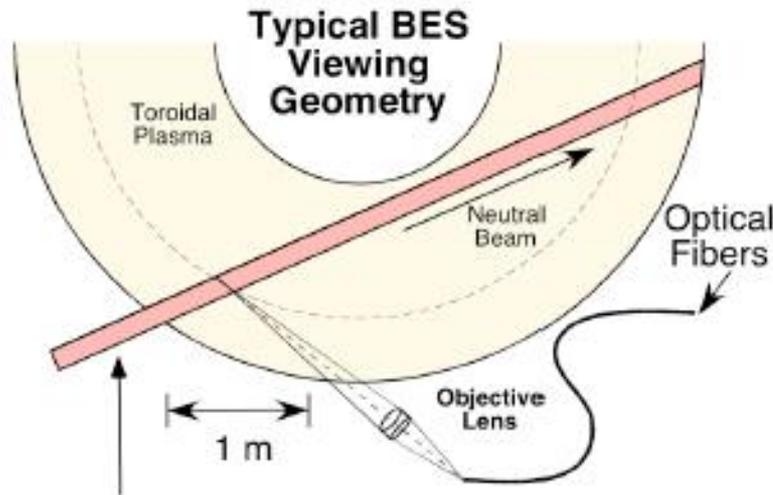
BEAM EMISSION SPECTROSCOPY MEASURES SPATIALLY LOCALIZED, LONG-WAVELENGTH ($k_\perp \rho_I < 1$) DENSITY FLUCTUATIONS

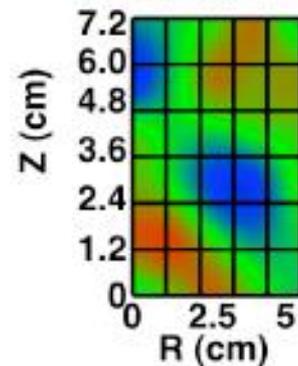Collisionally-excited, Doppler-shifted neutral beam fluorescence

$$D^o + e,i \Rightarrow \left(D^o\right)^* \Rightarrow D^o + \gamma(n = 3 \to 2, \lambda_o = 656.1 \text{ nm})$$
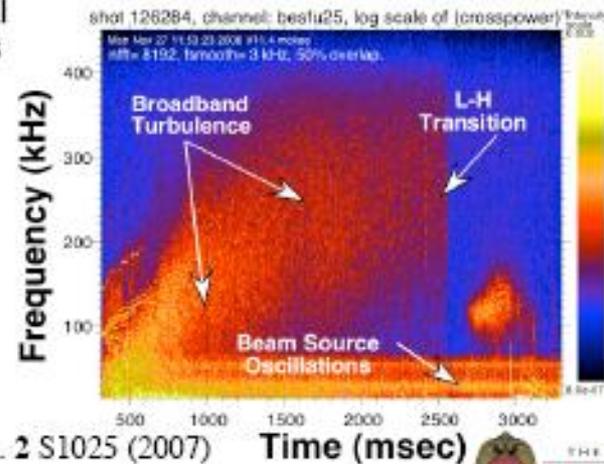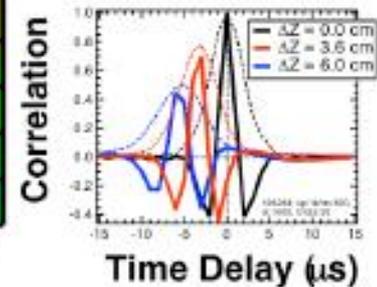
Emission intensity proportional to local density

Typical BES Viewing Geometry

Toroidal Plasma

Neutral Beam

Optical Fibers

Objective Lens

1 m

75 KeV D^o Neutral Beam

Imaging Array

Z (cm) — 7.2, 6.0, 4.8, 3.6, 2.4, 1.2, 0 ; R (cm) — 0, 2.5, 5

Poloidal Cross Correlations

Correlation — Time Delay (μs)

ΔZ = 0.0 cm
ΔZ = 3.6 cm
ΔZ = 6.0 cm

shot 126284, channel: besfu25, log scale of [crosspower]

Frequency (kHz)

Broadband Turbulence

L-H Transition

Beam Source Oscillations

Time (msec)

G. R. McKee et al., Plasma Fusion Res. 2 S1025 (2007)

DIII-D NATIONAL FUSION FACILITY

THE UNIVERSITY of WISCONSIN MADISON

Comparisons of Local Simulations Undertaken at ρ = 0.5 and 0.75

- Primary focus of work to date has been comparisons of local simulations against experiment at ρ = 0.5 and 0.75

- Locations represent a trade-off between ease of simulation and signal-to-noise ratio of fluctuation measurements
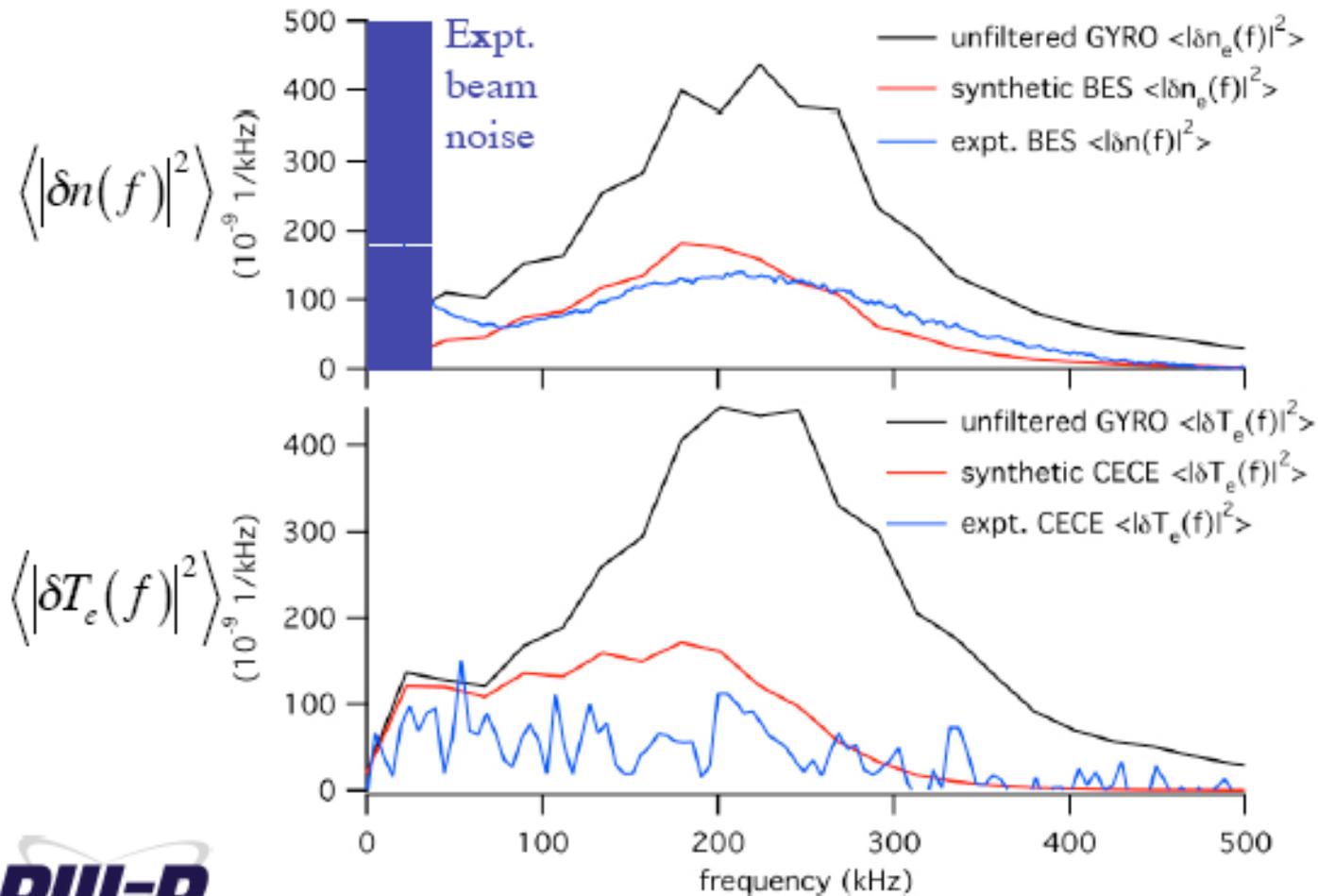
(Courtesy of C. Holland)
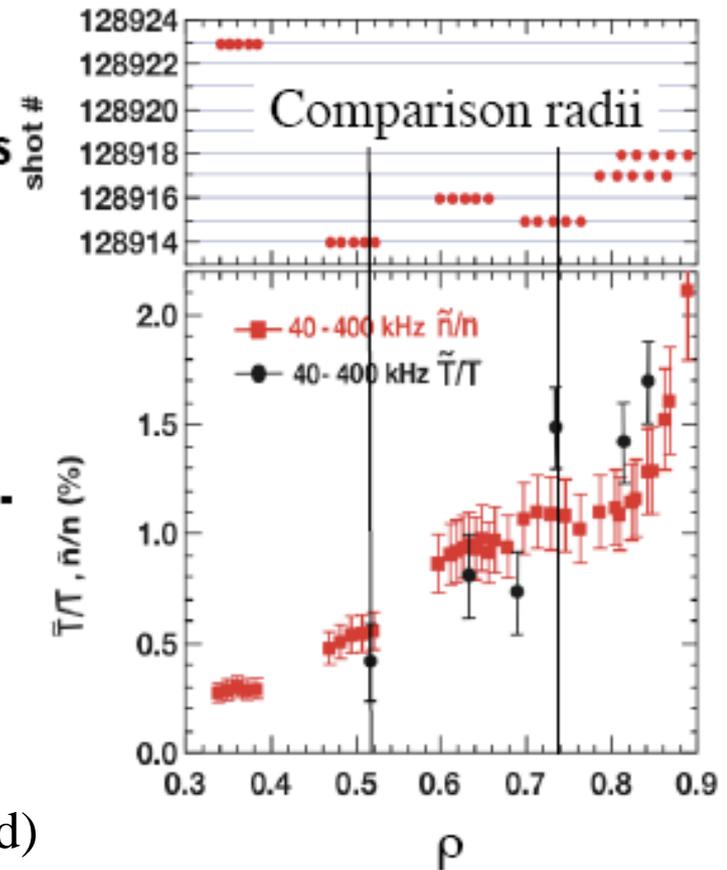
Ex: Applying BES PSF to GYRO Simulation Data

- IDL post processing tool written to generate synthetic BES array; PSF form taken from calculation by M. Shafer

- Tool first interpolates PSF data (generated on a regularly spaced (R,Z) grid) onto a grid compatible with GYRO data (which uses a field-line following (r,θ,α) coordinate system)

- At each time point of interest, record
  - Synthetic signal defined as

$$\delta n_{synthetic}(x,y,t) = \frac{\int d^2x' \psi^{PSF}(x-x',y-y')\delta n_e^{GYRO}(x',y',t)}{\int d^2x' \psi^{PSF}(x-x',y-y')}$$

δn_e, t = 350

channel "location"

  - GYRO signal at gridpoint closest to nominal BES location (term this signal the unfiltered GYRO signal in this poster)

- Because GYRO calculates fluctuations in co-rotating reference frame, must transform data back into the lab reference frame. Linear interpolation is used to increase the effective time resolution (equivalent to sampling rate) of the GYRO data, preventing aliasing due to the introduction of the equilibrium Doppler shift.

(Courtesy of C. Holland)

DIII–D NATIONAL FUSION FACILITY SAN DIEGO

UCSD
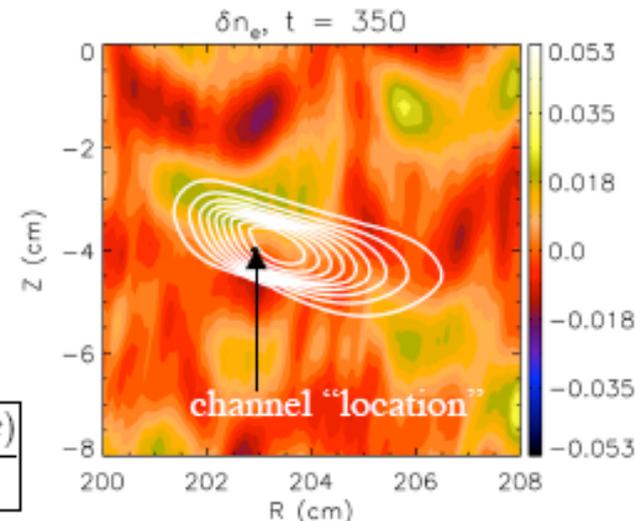
Example of a GYRO validation activity (C. Holland)

(Courtesy of C. Holland)

Summary

- Fusion simulation can provide an exciting career
  - One of the front runners in multiscale nonlinear self-organization science and integrated simulation
  - Innovative contributions are needed for the reduced model development
  - Contribution to the possible solution to the energy problem (semi-infinite time scale)
- Growing area
- Shortage of qualified your scientists is becoming a problem.
- Be ready for collaborative work with the applied mathematicians and computer scientists