NEPTUNE-Particles

Ben McMillan, Tom Goffrey, Tony Arber, Keith Bennett
CFSA, Physics, University of Warwick UK

Particle project.

- Why are particle codes useful for in plasma edge modelling?
- Particles/Neptune: work to date.
- Co-design: needs going forward.

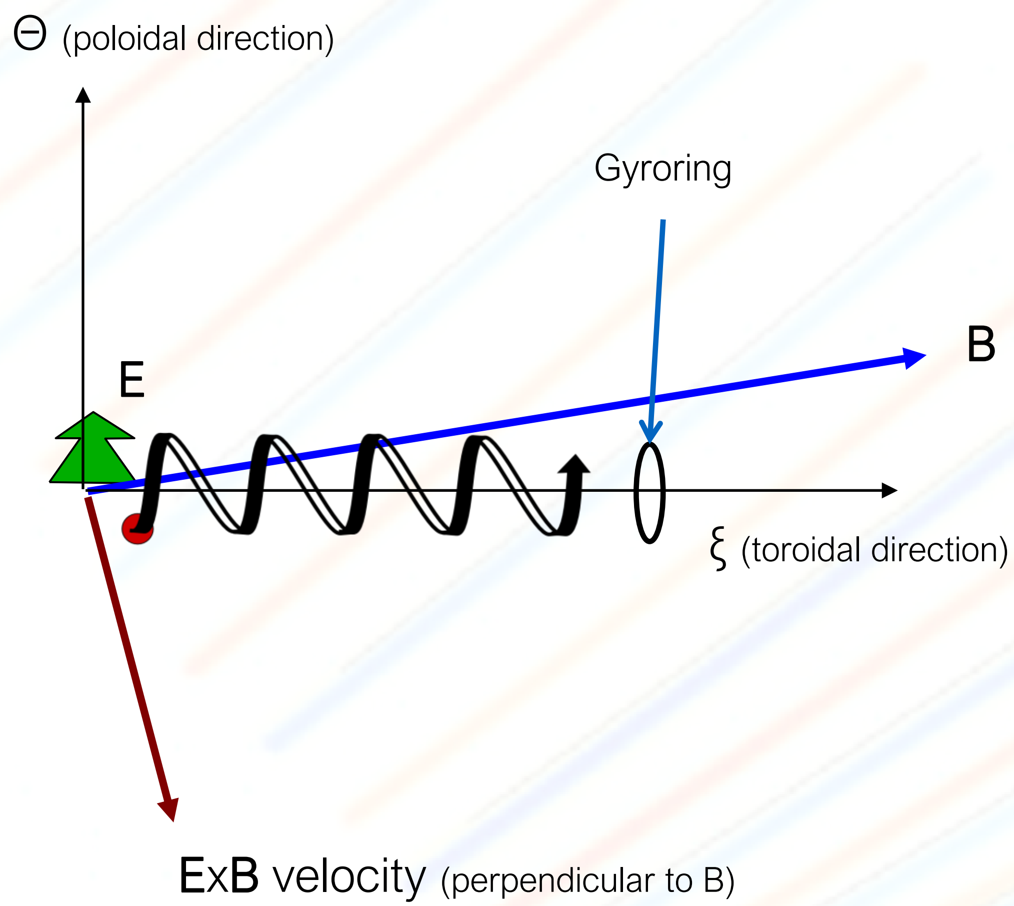
Need for particle methods.

- Low collisionality: fluid assumption often invalid. Especially for neutrals and hot plasma ejection in transient events.
- Describe distribution in velocity space. Several methods possible, but Lagrangian kinetics tend to be numerically favorable when:
  - 3D in velocity.
  - Collision operator cannot easily be written in PDE form.
  - Distribution function multi-component (multiple beams/temperatures).
  - Spatial grid is non-optimal. (Particle methods less sensitive)
- Ideally: flexible choice of fluid/Eulerian kinetic/Lagrangian kinetic based on location/species.

“Disruptive PIC methods”

- Current particle methods:
  - Limited timestep due to explicit fields. M4
  - Limited timestep to explicitly follow gyromotion. M4
  - Noisy: low noise methods only work for small fluctuations M5
  - Don’t integrate well with fluid methods M5.
- Edge plasmas can be handled with PIC, but massive improvements possible.

Gyrokinetic PIC (M4/M6)



- Gyrokinetics: Instead of solving the full particle orbits, consider the gyrating particles as rings of charge.
- Allows us to avoid fast timescales.
- PIC gyrokinetics is standard: e.g ORB5 code (Warwick, SPC Switzerland, IPP Germany)
- Basic drift-kinetic particles equations and currents now added to the EPOCH-miniapp.
- Pathway to advanced EM algorithms, mixed schemes.

More minor improvements. (M4)

- Refactoring code to split out timestepping: allow integration with external EM field solvers (BOUT++, Nektar++ ?)
- Continuous integration.
- Initial implementation of particle sub-stepping. (particle solve timestep decoupled from field update).

Fluid/Particle interaction (M5/M6)

- In core tokamak codes (e.g. ORB5) ”control variates” used $f = f_0 + g$ to reduce noise.
- Moments of f_0 known analytically, moments of g use Monte-Carlo: reduced variance (“noise”).
- Adaptive f_0 based on fluid equations:
  - Allows large fluctuations (edge relevant).
  - Setting g=0 allows fallback to fluid limit.

Fluid/PIC (M5/M6)

$$f_0(\mathbf{Z},t) = \frac{n(\mathbf{x},t)}{[2\pi T(\mathbf{x},t)]^{3/2}} \exp\left[\frac{m\{\mathbf{v}-\mathbf{v}_0(\mathbf{x},t)\}^2/2}{kT(\mathbf{x},t)}\right]$$

Requiring that first 3 moments of g are zero:

$$\frac{\partial n}{\partial t} + \nabla.[n\mathbf{v}] = \int d\mathbf{v}S, \tag{4}$$

$$\frac{\partial nv}{\partial t} + \nabla.[n\mathbf{v}^2 + 3nkT] - \mathbf{F} = \int d\mathbf{v}vS, \tag{5}$$

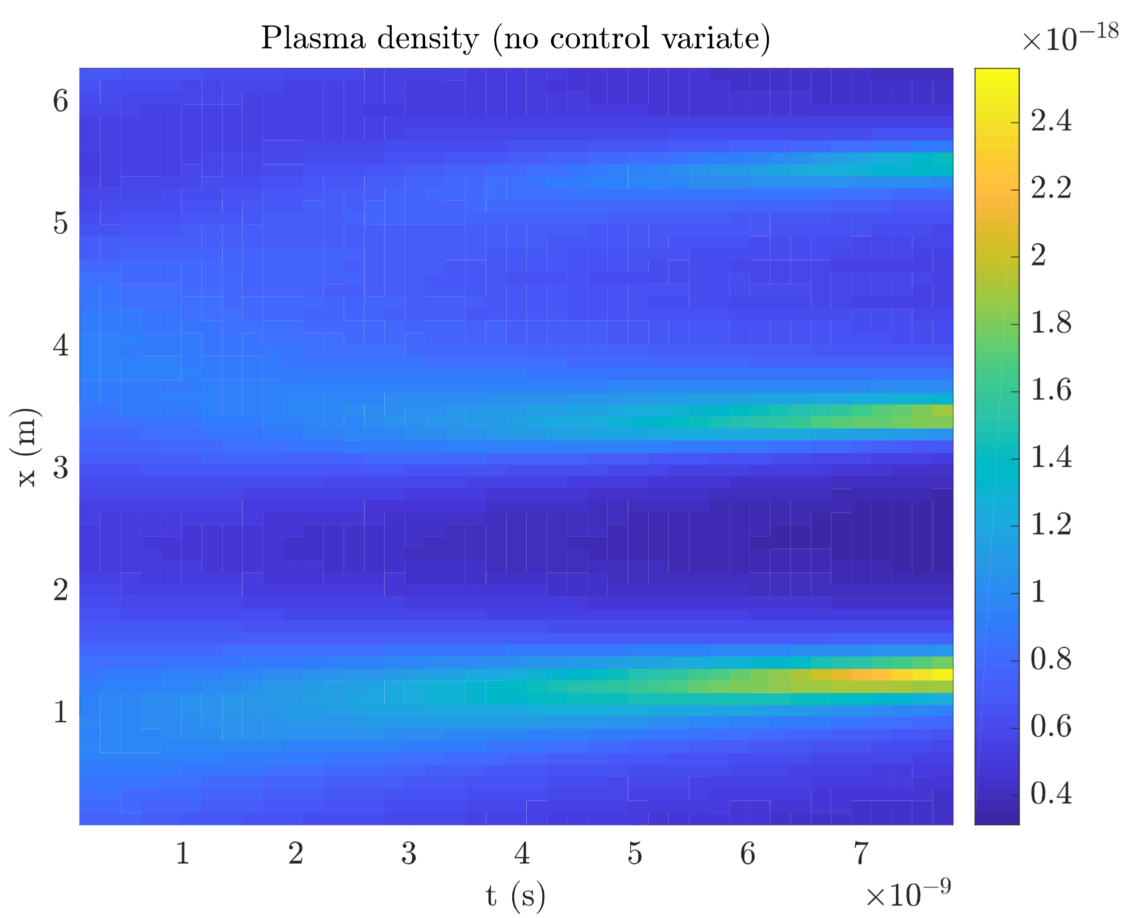
$$\frac{\partial[3nkT + nv_0^2]}{\partial t} + \nabla.\left[\mathbf{v_0}\{3nkT + nv_0^2\} + \int dV(\mathbf{v}-\mathbf{v_0})(v-v_0)^2 g\right] - \mathbf{F}.\mathbf{v_0} = \int d\mathbf{v}v^2 S.$$

$$\frac{dg}{dt} = -\frac{df_0}{dt} + S(g + f_0).$$

Very similar to proposed Eulerian GK scheme (Neptune/Gyrokinetics): rescaling velocities not useful for particle case but can share fluid scheme.

Fluid/PIC (M5/M6)

- Simple 1D force-free testcase.
- Control variates simulation; Reduction in variance in Monte-Carlo integral by factor of ~1000. (i.e. 1000 times fewer markers for same noise level).



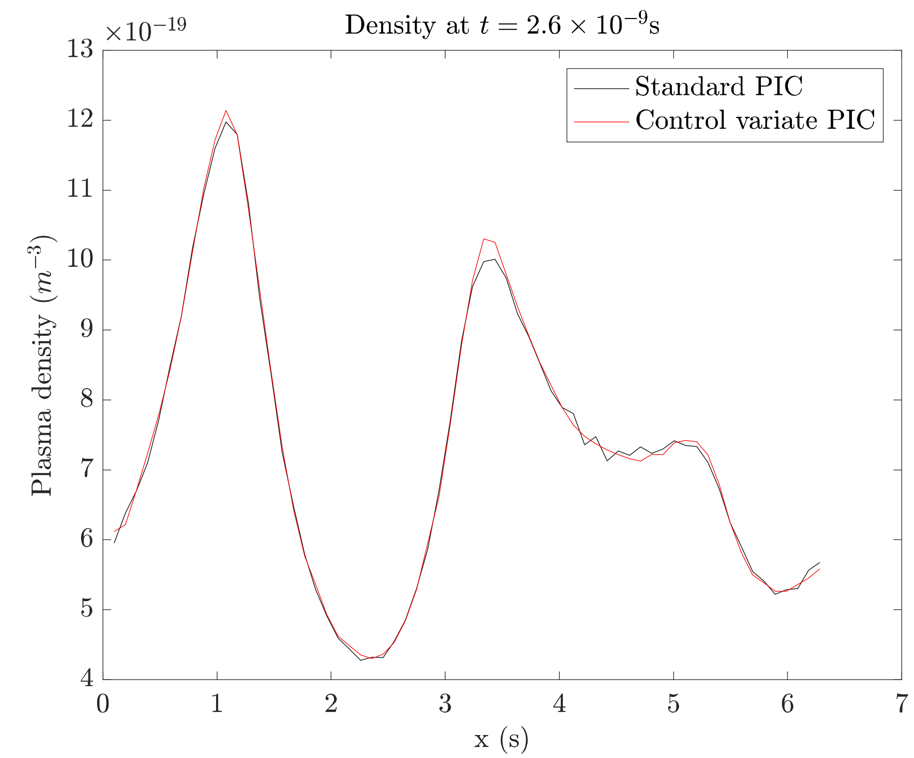


Figure 1: Density versus position at an intermediate simulation time.

View to future: codesign (M6)

- Existing PIC with structured FEM grids (ORB5/XGC). Unstructured meshes?
- Nektar++ has particle tracing, and plasma equations (Hasegawa-Wakatani)
- These are extensible to general particle equations.
- Proof of principle: seed particles in Nektar-Driftwave.
- Potential accuracy issues for piecewise curvilinear grid? EM fields?