Numerical implementation of the fully non-linear Fokker-Planck-Landau operator

R. Hager[1)], E. S. Yoon[2)], S. Ku[1)], E. F. D'Azevedo[3)], P. H. Worley[3)], C.S. Chang[1)]

[1)]Princeton Plasma Physics Laboratory
[2)]Rensselaer Polytechnic Institute
[3)]Oak Ridge National Laboratory

EPSi Edge Physics Simulation

PPPL PRINCETON PLASMA PHYSICS LABORATORY

Outline

- **Introduction: Need for non-linear collision operator**
- Implementation of the non-linear, multi-species Fokker-Planck-Landau operator in XGC1 and XGCa
  - Spatial and time discretization
  - Conservation properties
  - Verification and scalability
- Alternative approaches
  - RMJ collision operator
  - Explicit time stepping
- Summary

Introduction

- Collisions are essential for the simulation of fusion plasma
- In most kinetic codes, a linearized form of the Fokker-Planck collision operator is used

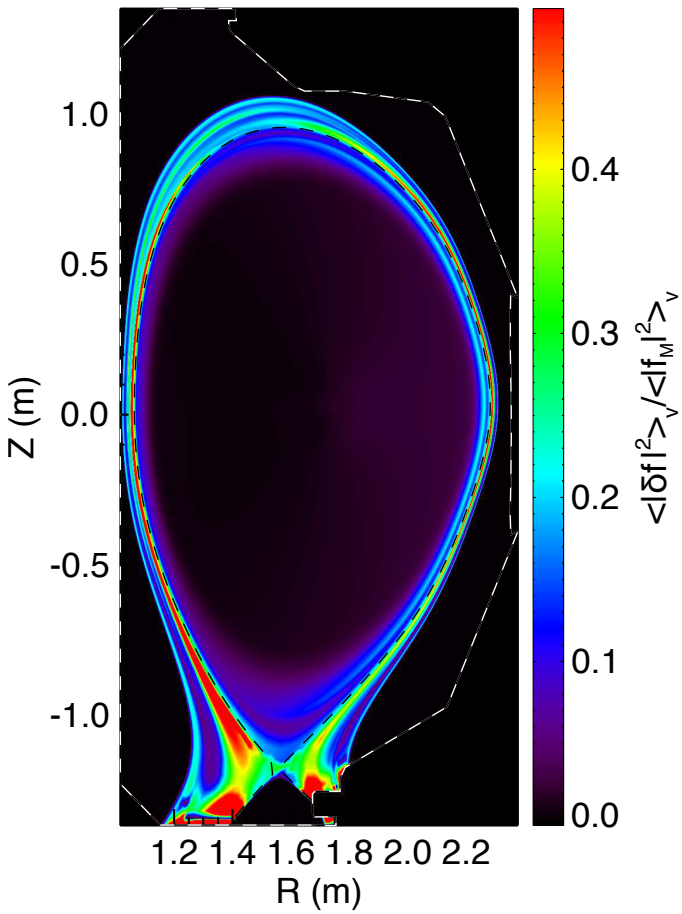
$$f = f_M + \delta f, \ \frac{\delta f}{f_M} \sim \epsilon \ll 1$$

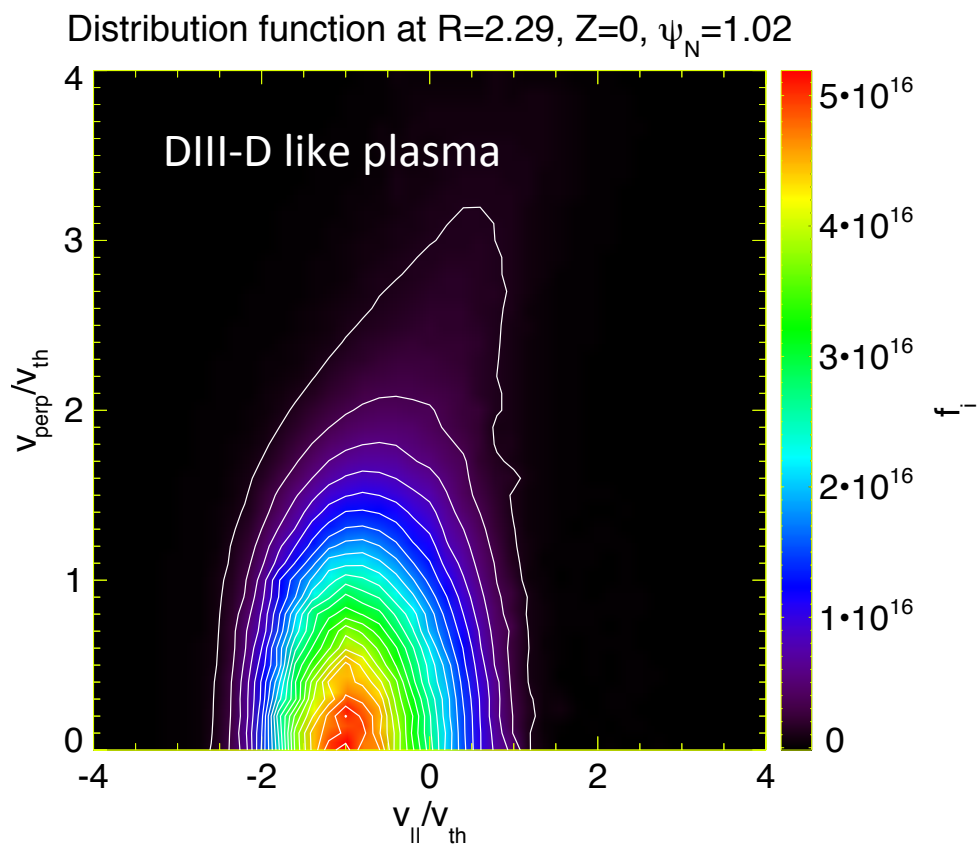
- But in the plasma edge pedestal and scrape-off layer
  - $L_{\nabla P}$ ~ orbit width
  - $\delta f/f$ ~ 1
  - Non-thermal plasma

  → Non-linear collision operator is needed for accurate results
- A non-linear Fokker-Planck-Landau (FPL) operator for single-species particle-in-cell simulations was developed by Yoon and Chang (PoP, 2014)
- This work: generalization to multi-species and HPC-efficient implementation in XGC1 and XGCa

Non-maxwellian ions in the tokamak edge require a non-linear collision operator

- Deviations of the ion distribution function from Maxwellian become substantial in the edge pedestal and scrape-off layer





Differences between the Landau and the RMJ form of the FP operator

- Landau form:
  - Integral definition of drag and diffusion coefficients E and D → impacts performance [O(N²)]
  - Symmetry of tensor U beneficial for numerical conservation laws
- RMJ form:
  - Poisson-type equations for drag and diffusion coefficients → can be solved efficiently [O(N)]
  - Achieving good conservation requires more complicated algorithms

$$\left.\frac{\mathrm{d}f_a}{\mathrm{d}t}\right|_{col} = \sum_b C_{ab}(f_a, f_b')$$
$$= -\sum_b \frac{e_a^2 e_b^2 \ln\Lambda_{ab}}{8\pi\epsilon_0^2 m_a} \nabla\cdot\left[\int \mathrm{d}^3v'\, \underline{\boldsymbol{U}}\cdot\left(\frac{f_a}{m_b}\nabla' f_b' - \frac{f_b'}{m_a}\nabla f_a\right)\right]$$
$$= -\sum_b \nabla\cdot(\boldsymbol{E}_{ab} f_a + \underline{\boldsymbol{D}}_{ab}\cdot\nabla f_a) = -\sum_b \nabla\cdot\boldsymbol{J}_{ab}$$

$$C_{ab}(f_a, f_b') = \frac{\gamma_{ab}}{2}\nabla\cdot\left[(\nabla\nabla G_b)\cdot\nabla f_a - 2\frac{m_a}{m_b} f_a \nabla H_b\right],$$
$$\nabla^2 G_b = 2H_b,$$
$$\nabla^2 H_b = -4\pi f_b,$$
$$\gamma_{ab} = \frac{4\pi e_a^2 e_b^2 \ln\Lambda}{m_a^2}$$

Requires f_a and f_b to be known in the same velocity boundaries!

Common simplifications of the Fokker-Planck operator

- Linearization:

$$f_{a/b} \approx f_{a/b,M} + \delta f_{a/b} \Rightarrow C_{ab}\left(f_a, f_b\right) \approx C_{ab}\left(\delta f_{a/b}, f_{b,M}\right) + C_{ab}\left(f_{a,M}, \delta f_b\right)$$

  - The full linearized FP operator is still complicated because drag and diffusion from δf_b need to be evaluated in the second term (Belli, Candy PPCF 2012)
  - The second term can be approximated so that mass, momentum and energy are conserved, e.g. Hirshman-Sigmar (Phys. Fluids 1976) or W. X. Wang (PPCF 1999)
- Lorentz pitch angle collision operator
  - Justifiable for electron-ion collisions: $m_e/m_i \ll 1$

Collision operators in fusion codes

- **GYRO:** pitch angle scattering or Krook operator (Gyro technical manual)
- **CGYRO:** Hirshman-Sigmar (Phys. Fluids 1976), Lorentz model, and others
- **NEO:** linearized FP in RMJ form with accurate field-particle term (Belli, Candy, PPCF 2012) but with $\ln \Lambda_{ii} = \ln \Lambda_{ee} = \ln \Lambda_{ie}$
- **XGC0:** linear collision operator, Monte-Carlo (MC) (Boozer, Phys. Fluids 1981) collisions for test-particle term + Hirshman-Sigmar for field-particle term
- **GTS and GTC-NEO:** MC test-particle collisions, Wang's method (PPCF 1999) for field particle collisions
- **GENE:** linearized RMJ FP operator with simplified field-particle term (F. Merz, dissertation 2008)
- **COGENT:** non-linear FP in RMJ form ($v_{\parallel}$-μ space) (Dorf et al., Contrib. Plasma Physics 2014)
- **XGC1/XGCa:** non-linear FP in Landau form ($v_{\perp}$-$v_{\parallel}$ space) (Yoon, Chang, Phys. Plasmas 2014; Hager, Chang JCP 2016)

Outline

- Introduction: Need for non-linear collision operator
- **Implementation of the non-linear, multi-species Fokker-Planck-Landau operator in XGC1 and XGCa**
  - **Spatial and time discretization**
  - Conservation properties
  - Verification and scalability
- Alternative approaches
  - RMJ collision operator
  - Explicit time stepping
- Summary

Generalization of the Yoon-Chang operator to multiple-species is straightforward

- Yoon and Chang started from FPL collision operator for better conservation properties
- Advection-diffusion equation → use finite volume discretization
- Assume that f and f' are independent of the gyro-phase → simplification of **E** and **D**
- Simply sum over species:

$$2\pi \frac{\partial}{\partial t} \int \phi_a f_a \mathrm{d}^2 v = \sum_b \Gamma_{ab} \int \mathrm{d}^2 v \, \{ \nabla \phi_a \cdot \underbrace{\left[\left(\int \mathrm{d}^2 v' \underbrace{\begin{pmatrix} U_{\perp\perp'} & U_{\perp\parallel} \\ U_{\parallel\perp'} & U_{\parallel\parallel} \end{pmatrix}}_{U_E} \cdot \nabla' f_b' \right) f_a - \left(\int \mathrm{d}^2 v' f_b' \underbrace{\begin{pmatrix} U_{\perp\perp} & U_{\perp\parallel} \\ U_{\parallel\perp} & U_{\parallel\parallel} \end{pmatrix}}_{U_D} \right) \cdot \nabla f_a \right]}_{J_{ab}} \}$$

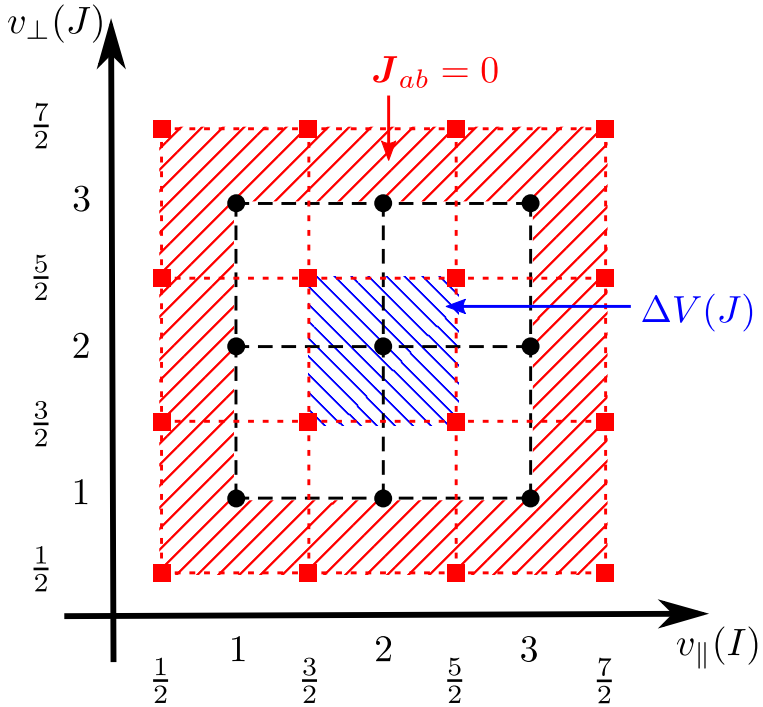
E_{ab}

D_{ab}

- Time evolution of the distribution functions of all species has to be evaluated together

Finite volume discretization

- Discretize the FPL equation using a uniform 2D ($v_{||}$-$v_{\perp}$) velocity grid spanning
  - $-N\, v_{th} \le v_{||} \le N\, v_{th}$
  - $0 \le v_{\perp} \le N\, v_{th}$
  - Usually: N=3-4 with a 40x40 grid
  - Midpoint quadrature for integrals
- Distribution functions $f_{a/b}$ are evaluated on the full-integer grid, fluxes $\mathbf{J}_{ab}$ are evaluated on the half-integer grid



- Important question here: Do we have to use identical velocity grids for each species? → enormous grids for $m_a \gg m_b$!!!
- Fortunately, we find excellent conservation properties when cutting off the velocity grids of individual species at N $v_{th,s}$ with N being a small integer ($f \sim \exp(-(v/v_{th,s})^2)$

Implicit time stepping

- Explicit time integration, e.g. $\delta f_{col} = \delta t\, C(f,f')$, requires very small time steps between subsequent collision operations ($\tau_e \ll \tau_i$)
- Therefore, we use implicit time marching (backward Euler)

$$\frac{f(t+\delta t) - f(t)}{\delta t} = C\left(f(t+\delta t), f(t+\delta t)\right)$$

- This equation is solved using a Picard iteration scheme together with direct solvers (LAPACK, SuperLU) or iterative solvers (PETSc)

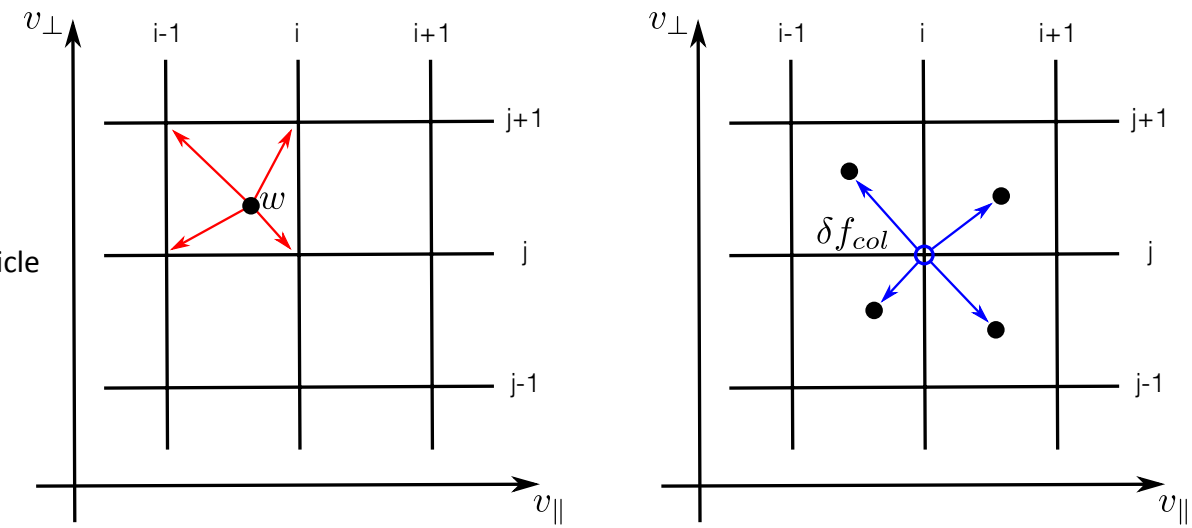
$$\frac{f_{a,i+1}^{(k)} - f_{a,i}}{\Delta t} = \sum_b C(f_{a,i+1}^{(k)}, f_{b,i+1}^{(k-1)})$$

  – **E** and **D** are evaluated with f of the previous iteration step
- The relative errors of total mass, momentum and energy are used as convergence criterion

Mesh-particle interpolation needed to combine PIC simulation with grid based collision operator

- For the collision operation, the plasma distribution functions of each species need to be compiled on the v-space grid
- Linear interpolation between particles and velocity grid

Particles → $f_{particle}$



δf_{col} → δw_i

- Mesh-particle interpolation is another source of error → can be controlled by the number of particles per collision cell and Δv

Outline

- Introduction: Need for non-linear collision operator
- **Implementation of the non-linear, multi-species Fokker-Planck-Landau operator in XGC1 and XGCa**
  - Spatial and time discretization
  - **Conservation properties**
  - Verification and scalability
- Alternative approaches
  - RMJ collision operator
  - Explicit time stepping
- Summary

Conservation laws

- Start with weak form

$$\frac{\partial}{\partial t}\sum_a \int \mathrm{d}^2 v \phi_a f_a = \sum_a \int \mathrm{d}^2 v \left[\phi_a \sum_b C_{ab}(f_a, f_b')\right]$$
$$= -\sum_{a,b} \int \mathrm{d}^2 v \phi_a \nabla \cdot \boldsymbol{J}_{ab} = \sum_{a,b} \int \mathrm{d}^2 v \nabla \phi_a \cdot \boldsymbol{J}_{ab}$$

- For each species pair

$$\int \mathrm{d}^2 v \int \mathrm{d}^2 v' \left[\phi_a C(f_a, f_b') + \phi_b' C(f_b', f_a)\right] =$$
$$= \int \mathrm{d}^2 v' \int \mathrm{d}^2 v' \left[m_b f_a \boxed{\left(\nabla \phi_a \cdot \underline{\boldsymbol{U}}_E - \frac{m_a}{m_b}\nabla' \phi_b' \cdot \underline{\boldsymbol{U}}_D'\right)} \cdot \nabla' f_b' \right.$$
$$\left. - m_a f_b' \boxed{\left(\nabla \phi_a \cdot \underline{\boldsymbol{U}}_D - \frac{m_a}{m_b}\nabla' \phi_b' \cdot \underline{\boldsymbol{U}}_E'\right)} \cdot \nabla f_a \right] = 0$$

  - For mass ($\phi_a = m_a$),
  - Momentum ($\phi_a = m_a v_{||}$),
  - Energy ($\phi_a = m_a v^2/2$)
  - Due to symmetries of $\mathbf{U}_E$ and $\mathbf{U}_D$

=0 independently → detailed conservation laws

Discrete conservation laws are exact

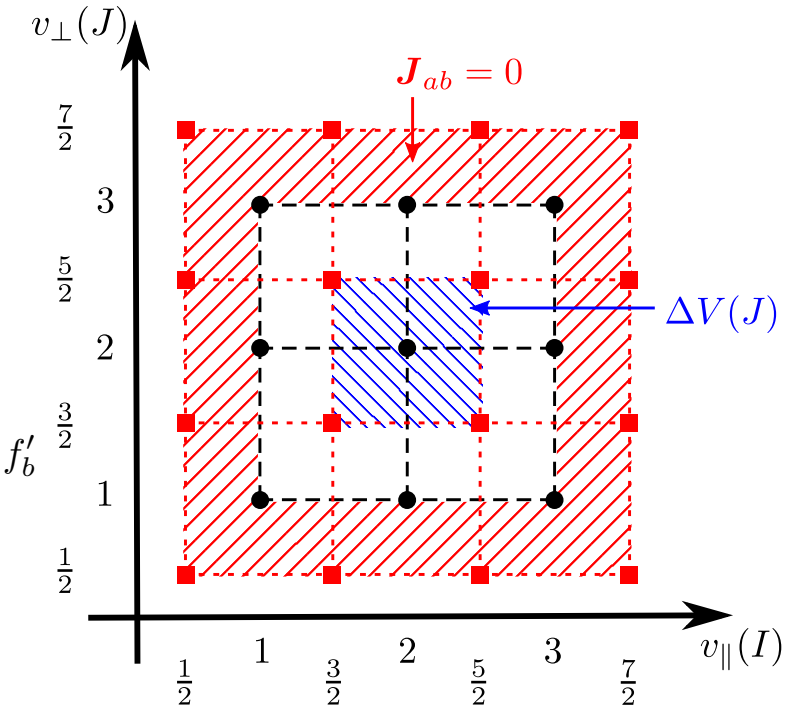
$$\frac{\partial}{\partial t}\sum_a\sum_{I=1}^{N}\sum_{J=1}^{M}\Delta V_a(J+\frac{1}{2})\phi_a f_a$$

$$=\sum_{a,b}\sum_{I=1}^{N-1}\sum_{J=1}^{M-1}\Delta V_a(J+\frac{1}{2})(\nabla\phi_a)(I+\frac{1}{2},J+\frac{1}{2})$$

$$\cdot\,\boldsymbol{J}_{ab}(I+\frac{1}{2},J+\frac{1}{2})$$

$$=\sum_{a,b}\sum_{I,J}\sum_{i,j}{}'\Big[m_b f_a\boxed{\left(\nabla\phi_a\cdot\underline{\boldsymbol{U}}_E-\frac{m_a}{m_b}\nabla'\phi_b'\cdot\underline{\boldsymbol{U}}_D'\right)}\cdot\nabla' f_b'$$

$$-m_a f_b'\boxed{\left(\nabla\phi_a\cdot\underline{\boldsymbol{U}}_D-\frac{m_a}{m_b}\nabla'\phi_b'\cdot\underline{\boldsymbol{U}}_E'\right)}\cdot\nabla f_a\Big]=0$$



- Distribution functions f are evaluated on the (I,J) grid
- Fluxes **J** are evaluated on staggered grid (I±1/2,J±1/2)
- $\nabla\cdot$ **J** is evaluated on the faces of the finite volume elements
- J is zero outside of the (I,J) grid (f=∇f=0)

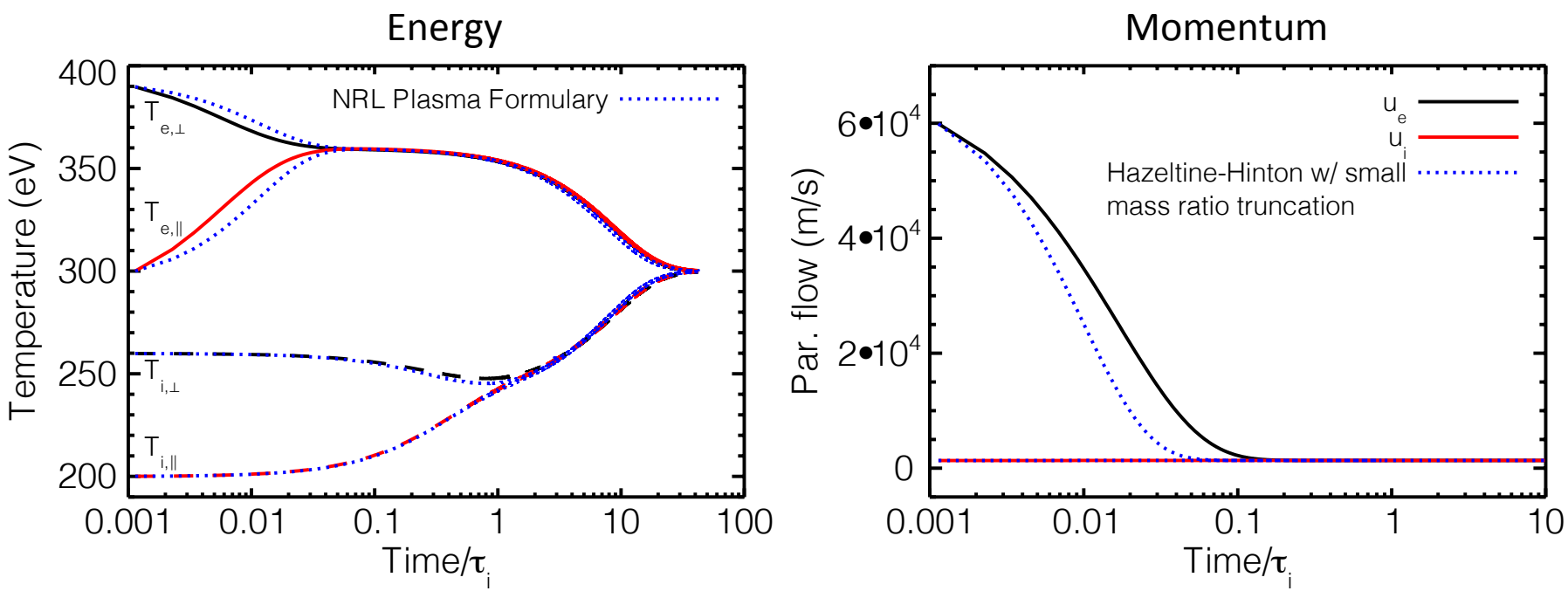
→ No flux of phase space density into or out of the velocity grid (I,J)

Outline

- Introduction: Need for non-linear collision operator
- **Implementation of the non-linear, multi-species Fokker-Planck-Landau operator in XGC1 and XGCa**
  - Spatial and time discretization
  - Conservation properties
  - **Verification and scalability**
- Alternative approaches
  - RMJ collision operator
  - Explicit time stepping
- Summary

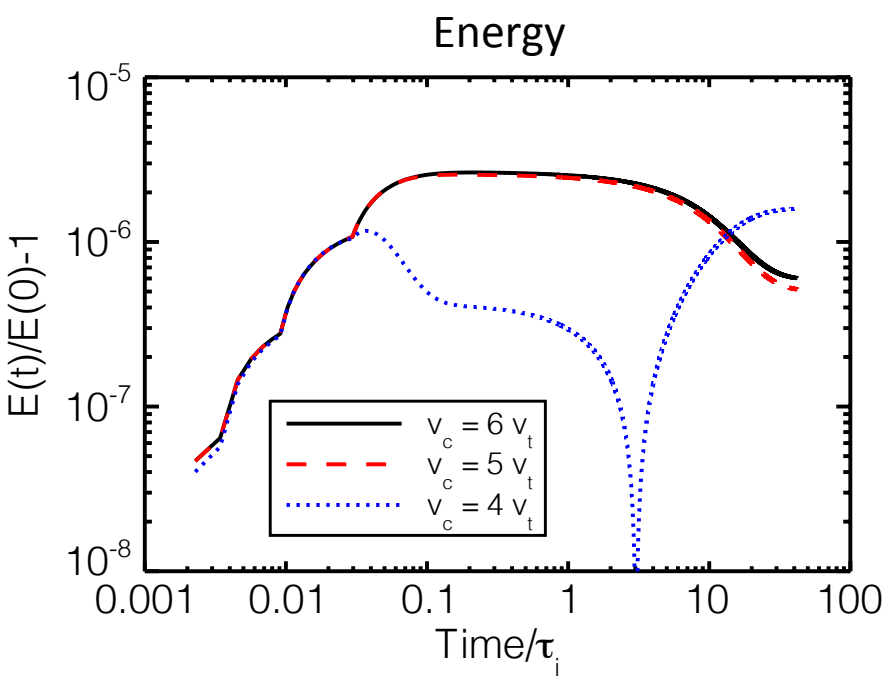
XGCa collision operator shows excellent conservation

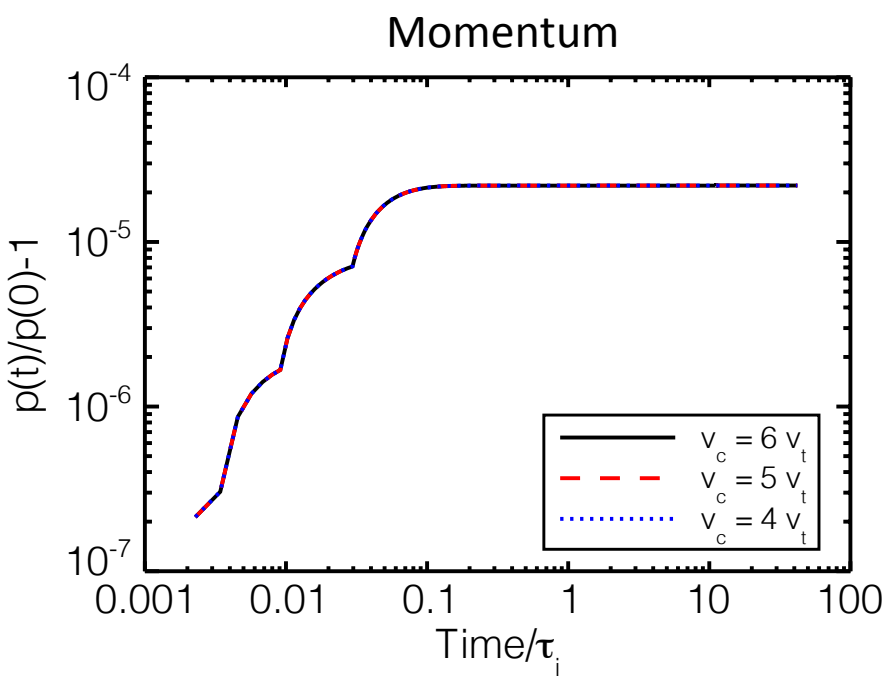
- Simultaneous relaxation of (realistic) parallel flows, temperature anisotropy, and T_i-T_e difference, frozen particles, **~30,000 collisions**
  - $T_{||e}$ (t=0)=300 eV, $T_{\perp e}$(t=0)=390 eV, $u_{||e}$(t=0)= 59.9 km/s
  - $T_{||i}$ (t=0)=200 eV, $T_{\perp i}$(t=0)=260 eV, $u_{||i}$(t=0)= 1.3 km/s



XGCa collision operator shows excellent conservation

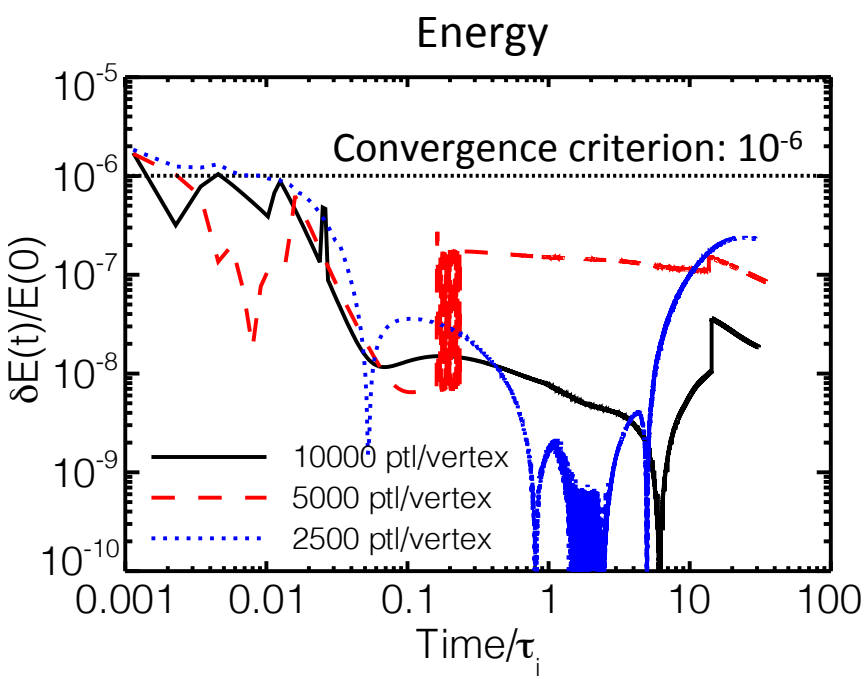
- Conservation **without** mesh-particle interpolation → error due to collision solver only
- Accuracy is adjustable by the convergence criterion (# of iterations) of the implicit time integrator

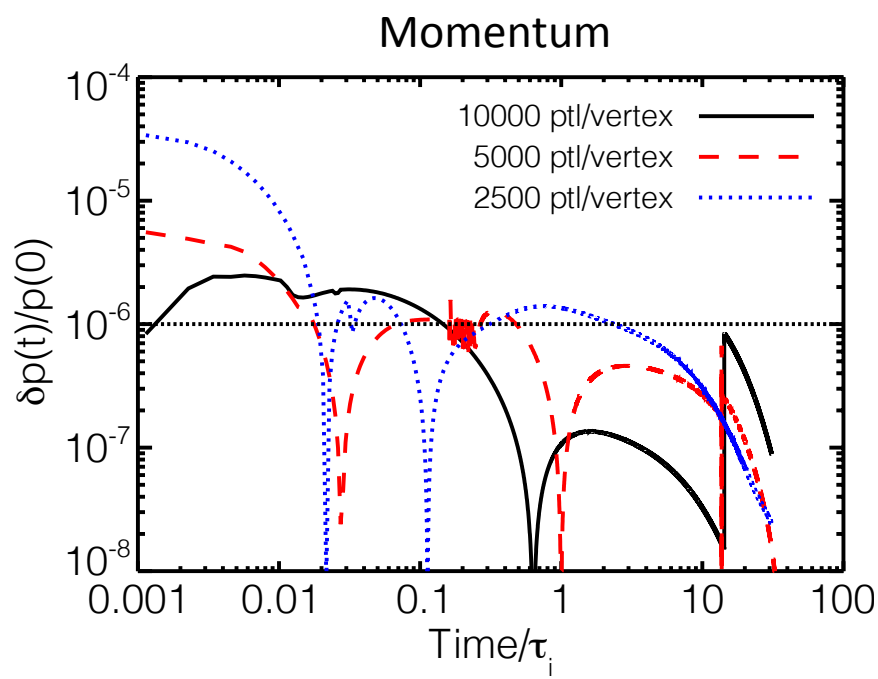




XGCa collision operator shows excellent conservation

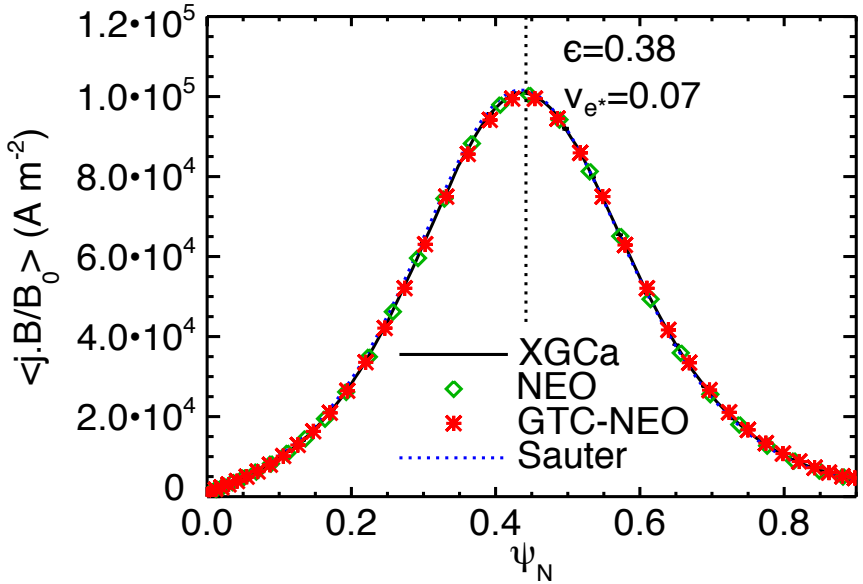
- Conservation **with** mesh-particle interpolation → additional interpolation error (Monte-Carlo type error if particles were not frozen)
- Accuracy needed for long time simulations is adjustable by the number of particles and convergence criterion



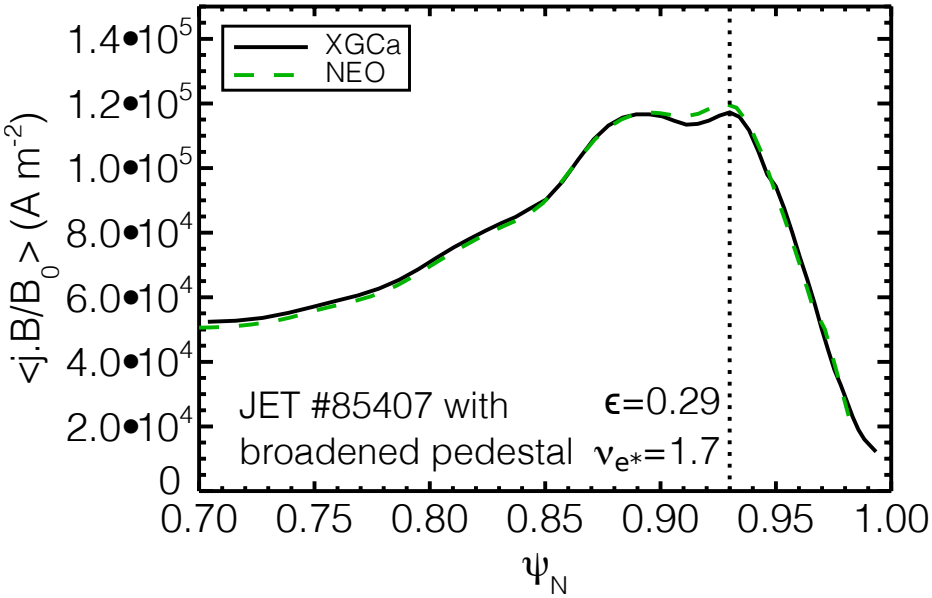


Good cross-verification with conserving linear collision operators where linearization is justified

- In the **local regime** (orbit width $\ll L_{\nabla P}$), excellent agreement between XGCa, XGC0/NEO (with conserving linear collision operator) is observed
- Good agreement in circular and realistic tokamak geometry

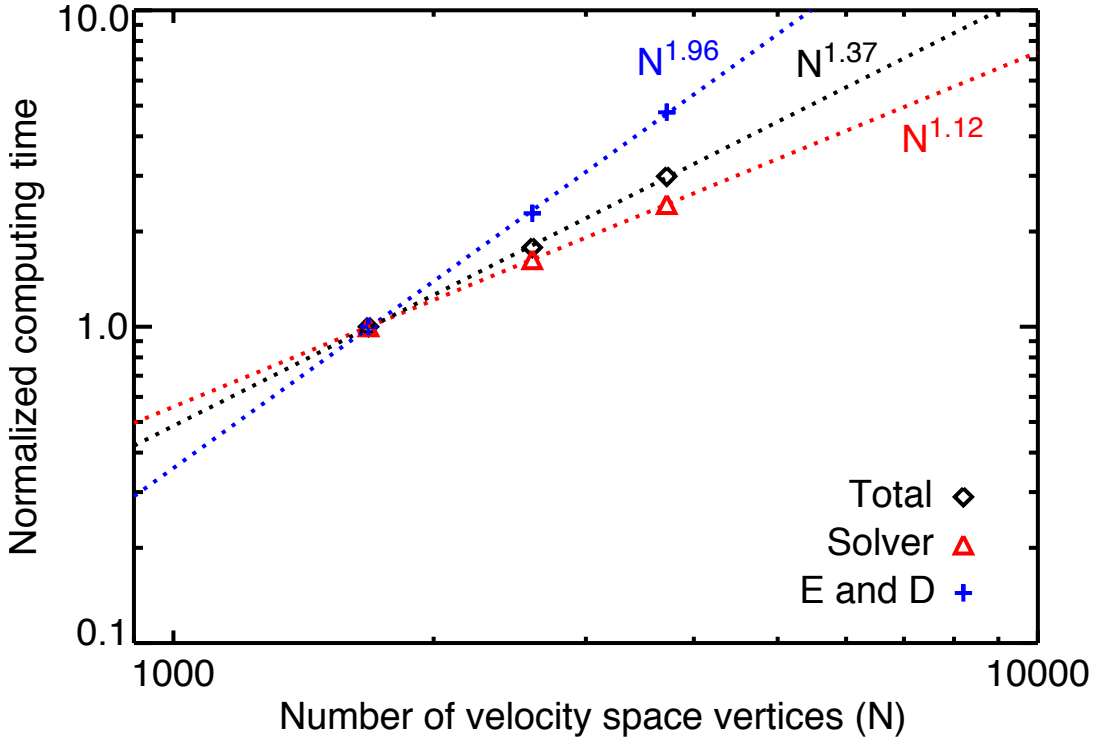


GTC-NEO result provided by
S. Ethier and W. Wang



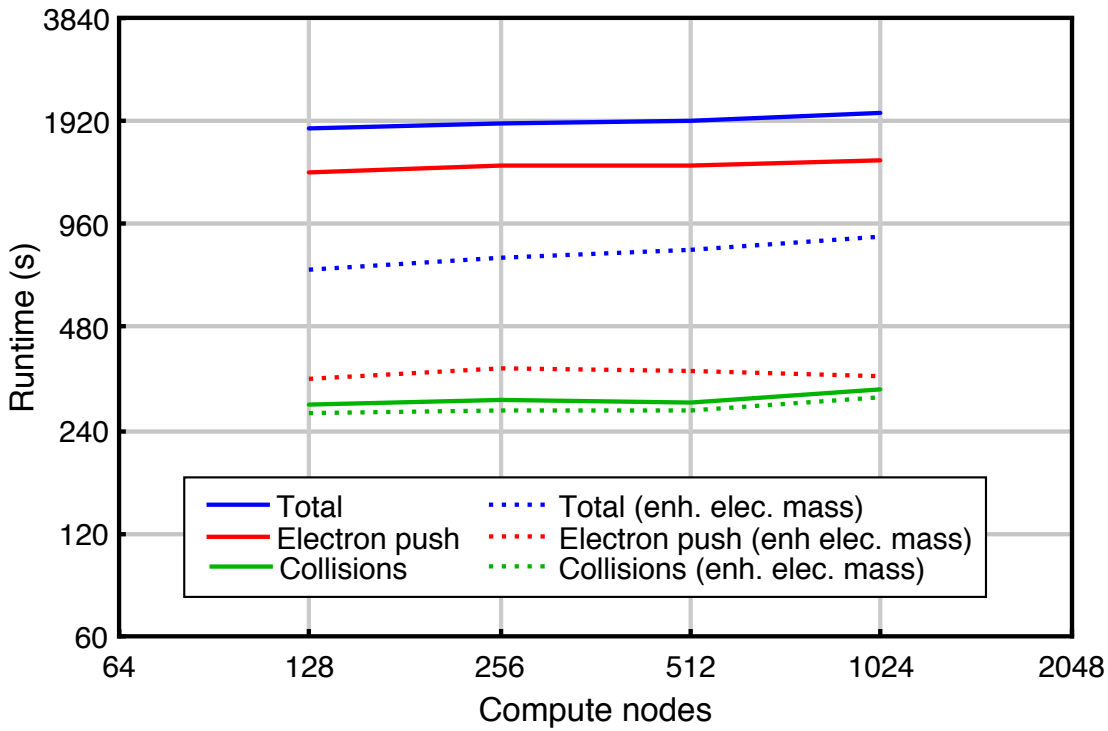
Velocity-space scaling

- Overall scaling dominated by solver [O(N)] for the practical mesh sizes used in XGC1 and XGCa
- Calculation of drag and diffusion coefficients scales like $O(N^2)$



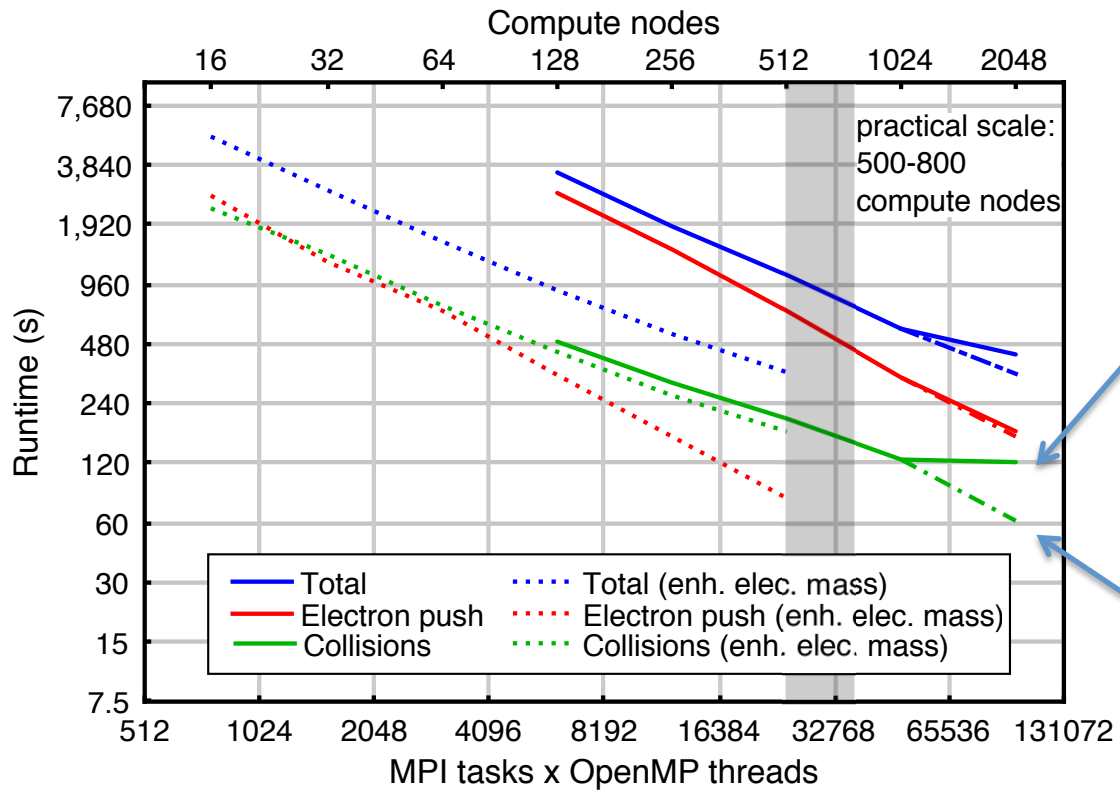
Weak scaling

- 10,188, 22,109, 40,197, 81,936 configuration space grid points, 41x41 velocity space grid points, 10,000 particles per configuration space grid point, 100 time steps
- Edison: 128, 256, 512, 1,024 nodes, 2 MPI/node, 24 OMP threads/MPI, all OpenMP threads used in outer level OpenMP parallelization
- Realistic and artificially enhanced electron mass



Strong scaling

- 20694 configuration space grid points, 41x41 velocity space grid points, 100 time steps
- Edison: 256 nodes, 2 MPI/node, 24 OMP threads/MPI, all OpenMP threads used in outer level OpenMP parallelization
- $2 \cdot 10^8$ particles with realistic, 10^8 particles with artificially enhanced electron mass



Saturation when all threads are used in the outer OpenMP level and # of computational threads ~ # of configuration space grid points

Shifting some threads to inner OpenMP level restores scalability

Outline

- Introduction: Need for non-linear collision operator
- Implementation of the non-linear, multi-species Fokker-Planck-Landau operator in XGC1 and XGCa
  - Spatial and time discretization
  - Conservation properties
  - Verification and scalability
- **Alternative approaches**
  - RMJ collision operator
  - Explicit time stepping
- Summary

Alternative approaches using the non-linear RMJ operator

- **Pataki, Greengard (JCP 2011):**
  - 3D velocity space, cylindrical coordinates,
  - Use Fourier space for $v_{||}$ and the gyro-phase
  - needs high resolution in and high cut-off velocity for good conservation properties
- **Dorf et al. (Contrib. Plasm Physics 2014):**
  - 2D velocity space ($v_{||}$-μ)
  - finite difference solver
- **Taitano, Chacón et al. (JCP 2015):**
  - 2D velocity space ($v_{||}$-$v_{\perp}$)
  - Finite volume approach with special attention on accurate conservation laws and preservation of positivity
  - 2nd order implicit time stepping with Jacobian-free Newton-Krylov solver

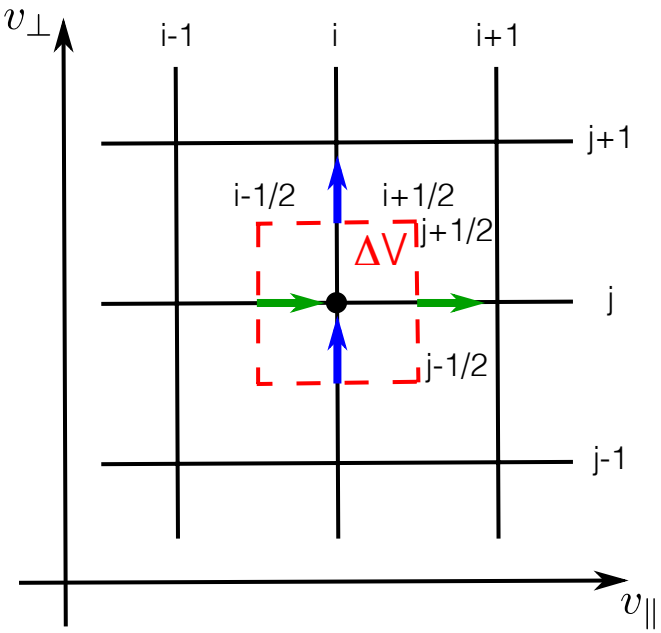
Alternative time stepping schemes

- Simple explicit time stepping, instead of our implicit time stepping, is too expensive for many applications

  → Larroche (JCP 2007) uses an explicit sub-cycling technique with individual time steps for each velocity space cell

1) Determine the update frequency of the fluxes on each cell face from stability condition.
2) Whenever a flux is updated, the distribution functions on which it depends must be updated, too.
3) The update frequency of any cell must be equal to or higher than the update frequencies of the fluxes through its boundaries

→ Saves computing time



Summary

- The non-linear Fokker-Planck-Landau collision operator by Yoon and Chang was generalized to multiple species and efficiently implemented in total-δf codes XGC1 and XGCa: enabling non-Maxwellian edge simulation
- Multi-species collision operator verified in
  - simple relaxation tests and
  - full simulations in the local regime (where linearized collision operators can be used)
  - Good agreement with theory and NEO code
- Performance engineering is essential to make non-linear collisions affordable
- Alternative approaches using the non-linear RMJ collision operator or optimized explicit time stepping are also available