Grungy details for continuum kinetics in NIMROD
NIMROD Team Meeting, Madison, WI

E. Held[1] S. Kruger[2] J. King[2] NIMROD Team

[1]Department of Physics
Utah State University

[2]Tech-X Corp.

April 2, 2016

Electron, ion and hot particle DKEs in NIMROD

- Hazeltine's form for the first-order drift kinetic equation in energy, ε, and magnetic moment, μ, variables:

$$\frac{\partial f}{\partial t} + (\mathbf{v}_{||} + \mathbf{v}_D) \cdot \nabla f + \left(\mu \frac{\partial B}{\partial t} + e(\mathbf{v}_{||} + \mathbf{v}_D) \cdot \mathbf{E} \right) \frac{\partial f}{\partial \varepsilon} = C(f)$$

- Transforming to $\xi = v_{||}/v$ and $s = v/v_0$ yields

$$\frac{\partial f}{\partial t} + (\mathbf{v}_{||} + \mathbf{v}_D) \cdot \left[\nabla f - \frac{1-\xi^2}{2\xi} \nabla \ln B \frac{\partial f}{\partial \xi} - \frac{s}{2} \nabla \ln T_0 \frac{\partial f}{\partial s} \right] - C(f) +$$
$$\frac{1-\xi^2}{2\xi} \left[-\xi^2 \frac{\mathbf{b}}{B} \cdot \frac{\partial \mathbf{B}}{\partial t} + \frac{q}{s^2 T_0} (\mathbf{v}_{||} + \mathbf{v}_D) \cdot \mathbf{E} + \xi^2 \frac{\mathbf{E} \times \mathbf{B}}{B^2} \cdot \nabla \ln B \right] \frac{\partial f}{\partial \xi} +$$
$$\frac{s}{2} \left[-(1-\xi^2) \frac{\mathbf{b}}{B} \cdot \frac{\partial \mathbf{B}}{\partial t} + \frac{q}{s^2 T_0} (\mathbf{v}_{||} + \mathbf{v}_D) \cdot \mathbf{E} + (1+\xi^2) \frac{\mathbf{E} \times \mathbf{B}}{B^2} \cdot \nabla \ln B \right] \frac{\partial f}{\partial s} = 0$$

where

$$\mathbf{v}_D = \frac{\mathbf{E} \times \mathbf{B}}{B^2} + \frac{T_0 s^2}{q B^2} \left[(1+\xi^2) \mathbf{b} \times \nabla B + 2\xi^2 \mu_0 \mathbf{J}_\perp + (1-\xi^2) \mu_0 \mathbf{J}_{||} \right] + \frac{m v_0 s \xi}{q B^2} \mathbf{b} \times \frac{\partial \mathbf{B}}{\partial t}.$$

Use NIMROD's spatial representation and efficient 2D velocity space representation.

Distribution functions expanded as

$$f(R,Z,\phi,\xi,s,t) = \sum_i f_{i,n=0}(\xi,s,t)\alpha_{i,n=0} +$$

$$\sum_{i,n>0} f_{i,n}(\xi,s,t)\alpha_{i,n} + f^*_{i,n}(\xi,s,t)\alpha^*_{i,n},$$

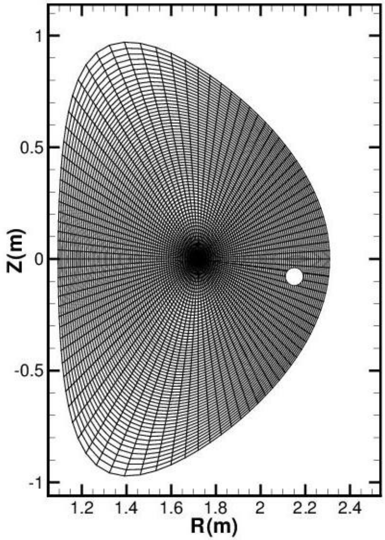
where $\alpha_{i,n} \equiv \psi_i(x,y)\exp(in\phi)$ and for 2D velocity space

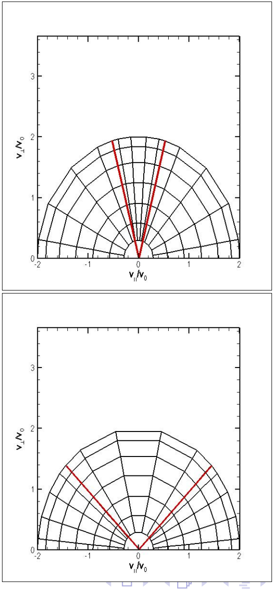
$$f_{i,n}(\xi,s,t) = \sum_l \sum_{k=0}^{N_s-1} f_{i,n,l,k}(t)P_l(\xi)\delta(s-s_k).$$

$P_l(\xi)$ are 1D FE in a pitch-angle type variable, ξ, and DKEs are solved at N_s collocation points in normalized speed, s.

Vertex nodes in ξ grid at $\pm\xi_t(\psi)$ may help convergence.

t/p bnd varies in poloidal plane





Correct streaming term with trapped/passing grids.

- Started with $(\mathbf{v}_{||} + \mathbf{v}_D) \cdot \nabla|_{\mu,w,t} f$.
- Rewrote as $(\mathbf{v}_{||} + \mathbf{v}_D) \cdot \left(\nabla|_{\xi,s,t} + \nabla\xi\partial/\partial\xi + \nabla s\partial/\partial s\right) f$.
- Finally transform from ξ to logical pitch-angle variable η: $(\mathbf{v}_{||} + \mathbf{v}_D) \cdot \left(\nabla|_{\eta,s,t} + \nabla|_\xi \eta\partial/\partial\eta\right) f$
- Second term needed when distribution functions vary in pitch-angle and $\nabla|_\xi \eta = \nabla R\partial\eta/\partial R + \nabla Z\partial\eta/\partial Z$.
- In negative passing space mapping is $\xi = -\cos\left(\theta_{tp}\eta/m\right)$ where $\theta_{tp} = \theta_{tp}(R,Z)$

Can also interpret as 3D FE method.

- Basis functions: $\alpha = \alpha(R(x,y), Z(x,y), \xi(x,y,\eta))$

- 3D Jacobian: $J_{3D} = \left(\frac{\partial R}{\partial x}\frac{\partial Z}{\partial y} - \frac{\partial R}{\partial y}\frac{\partial Z}{\partial x}\right)\frac{\partial \xi}{\partial \eta}$

$$\frac{\partial}{\partial R} = J_{2D}^{-1}\left(Z_y\frac{\partial}{\partial x} - Z_x\frac{\partial}{\partial y}\right) + J_{3D}^{-1}\left(Z_x\xi_y - Z_y\xi_x\right)\frac{\partial}{\partial \eta}$$

$$\frac{\partial}{\partial Z} = J_{2D}^{-1}\left(R_x\frac{\partial}{\partial y} - R_y\frac{\partial}{\partial x}\right) + J_{3D}^{-1}\left(R_y\xi_x - R_x\xi_y\right)\frac{\partial}{\partial \eta}$$

$$\frac{\partial}{\partial \xi} = \frac{\partial \eta}{\partial \xi}\frac{\partial}{\partial \eta}$$

- Both approaches were implemented as a check. They yield the same results.

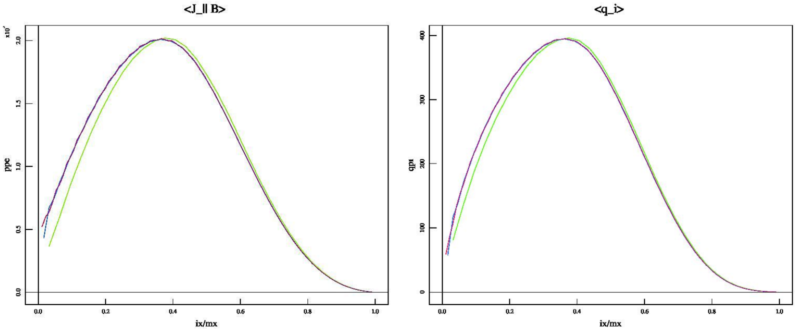
Velocity space coefficients coupled explicitly in integrands.f90.

- Field comps type routines construct coupling array, vpn.

- Implicit streaming terms for trapped/passing pitch-angle grids is

```
IF (gridshape_v(5:7)=='tpb') THEN
 int(jq,iq,:,jv,iv)=int(jq,iq,:,jv,iv)+SUM(
    dt*fcel*alpha(:,:,iv)*alpha(:,:,jv)*
   (vpn(1,il+n,il+m,:,:)*tpr(1,:,:)
   +vpn(2,il+n,il+m,:,:)*tpz(1,:,:))  ,1)
ENDIF
```

Revisit high-beta NEO/NIMROD benchmark case.



- Local trapped/passing boundary used througout poloidal plane, implementation no longer assumes grid alignment with flux surfaces.
- Can now test convergence properties of trapped/passing grids.
- Improved equilibria with resolve of GS equation help.
- Alternate solves for F_i and F_e speeds up computation.

Full linearized Coulomb collision operator important.

- Implemented full, linearized collision operator using 1D finite-elements in pitch angle and non-classical, orthogonal speed polynomials.
- Expand $F = \sum_l F_l(\mathbf{x}, s, t) P_l(\xi) = \sum_{lk} F_{lk}(\mathbf{x}, t) L_k(s) P_l(\xi)$ and use in

$$
\begin{aligned}
C_{ab}^{test} &= \Gamma_{ab}\{\frac{1}{2v^3} g_v^0 \mathscr{L}(f) + \frac{1}{v^2} g_v^0 f_v + \frac{1}{2} g_{vv}^0 f_{vv} + (1 - \frac{m_a}{m_b}) h_v^0 f_v + \frac{m_a}{m_b} 4\pi f_b^0 f_a\} \\
C_{ab}^{field} &= \Gamma_{ab}\{\frac{1}{2v^3} f_v^0 \mathscr{L}(g) + \frac{1}{v^2} f_v^0 g_v + \frac{1}{2} f_{vv}^0 g_{vv} + (1 - \frac{m_a}{m_b}) f_v^0 h_v + \frac{m_a}{m_b} 4\pi f_a^0 f_b\}.
\end{aligned}
$$

where $\mathscr{L} = \partial_\xi (1 - \xi^2) \partial_\xi$ and g and h are Rosenbluth potentials.

Issues with Lorentz operator?

- Weak form of Lorentz operator with “surface" term is

$$-\int_{-1}^{1} d\xi \, \frac{\partial \alpha'}{\partial \xi}(1-\xi^2)\frac{\partial F}{\partial \xi} + \int_{-1}^{1} d\xi \, \frac{\partial}{\partial \xi}\left[\alpha'(1-\xi^2)\frac{\partial F}{\partial \xi}\right]$$

- Considering that pitch-angle basis functions are in C^0, should jump conditions be kept at t/p boundaries?

$$\Delta\left[\alpha'(1-\xi^2)\frac{\partial F}{\partial \xi}\right]\Big|_{-\xi_t} + \Delta\left[\alpha'(1-\xi^2)\frac{\partial F}{\partial \xi}\right]\Big|_{+\xi_t}$$

Could static condensation in pitch-angle speed up nonlinear solves? Timing results from NEO/NIMROD benchmark.

```
Timing statistics:
                               CPU secs      % of loop
   Iteration time     =   4.15608E+02 3.42237E+01
   Factoring time     =   2.94936E+02 2.42868E+01
   SuperLU time       =   4.15697E+02 3.42310E+01
   FFT    time        =   6.08556E-02 5.01122E-03
   FE_matrix time     =   1.31587E+00 1.08357E-01
   FE_vector time     =   2.71233E+02 2.23350E+01
   Static_con time    =   3.35868E+00 2.76574E-01
   CEL_comp time      =   1.21408E+03 9.99746E+01
                      CPU secs    % of total
   Loop   time        =   1.21439E+03 9.48839E+01
   Setup time         =   6.54791E+01 5.11609E+00
   Total  time        =   1.27987E+03 1.00000E+02
```

Is static condensation in pitch-angle worth the effort?

- Preconditioning for nonlinear solves uses matrices that are diagonal in speed variable, s.
- Full coupling in pitch angle, however, seems wasteful with 3 FE cells and very high-order polynomials, pdxi = 5,...,15.
- Majority of nodes are interior and could be eliminated prior to preconditioning.
- Reduce matrix memory by improving storage scheme?
- All of this would likely require developing routines for matrix manipulation.