Continuum closure update*

Eric Held, J. Andy Spencer, Trevor Taylor, Brett Adair, Jeong-Young Ji, and Hankyu Lee

Department of Physics
UtahStateUniversity

NIMROD Summer Meeting, Logan, UT

*Work supported by DOE under grant nos. DE-FC02-08ER54973, DE-FG02-04ER54746, and DE-SC0018146.

Various forms for drift kinetic equation

Drift-kinetic equation (DKE) using different variables:

Hazeltine: $\bar{f}\,(U,\mu,\mathbf{x},t) \rightarrow \frac{\partial \bar{f}}{\partial t} + \mathbf{v}_{\mathrm{gc}} \cdot \nabla \bar{f} + \frac{dU}{dt}\frac{\partial \bar{f}}{\partial U} + \frac{d\mu}{dt}\frac{\partial \bar{f}}{\partial \mu} = C$

Ramos: $f\left(v_{\parallel},v_{\perp},\mathbf{x},t\right) \rightarrow \frac{\partial \bar{f}}{\partial t} + \mathbf{v}_{\mathrm{gc}} \cdot \nabla \bar{f} + \frac{dv_{\parallel}}{dt}\frac{\partial \bar{f}}{\partial v_{\parallel}} + \frac{dv_{\perp}}{dt}\frac{\partial \bar{f}}{\partial v_{\perp}} = C$

NIMROD: $\bar{f}\,(s,\xi,\mathbf{x},t) \rightarrow \frac{\partial \bar{f}}{\partial t} + \mathbf{v}_{\mathrm{gc}} \cdot \nabla \bar{f} + \frac{ds}{dt}\frac{\partial \bar{f}}{\partial s} + \frac{d\xi}{dt}\frac{\partial \bar{f}}{\partial \xi} = C$

Guiding center velocity and acceleration terms

δf DKE in NIMROD uses the coordinates $(\mathbf{x}, t, s, \xi)$, where $s = v/v_0$ and $\xi = v_\parallel/v$

$$\begin{aligned}
\mathbf{v}_{\rm gc} &= v_0 s\xi\mathbf{b} + \frac{\mathbf{E}\times\mathbf{B}}{B^2} + \frac{T_0 s^2}{qB}\left(1+\xi^2\right)\mathbf{b}\times\nabla\ln B \\
&\quad + \frac{2T_0 s^2}{qB^2}\left[\xi^2\left(\mathbf{I}-\mathbf{bb}\right) + \frac{1}{2}\left(1-\xi^2\right)\mathbf{bb}\right]\cdot\nabla\times\mathbf{B} + \frac{mv_0 s\xi}{qB^2}\mathbf{b}\times\frac{\partial\mathbf{B}}{\partial t} \\
\dot{s} &= -s\frac{d\ln v_0}{dt} + \frac{s\left(1-\xi^2\right)}{2}\frac{\partial\ln B}{\partial t} + \frac{q}{2T_0 s}\left(\mathbf{v}_\parallel + \mathbf{v}_{\rm c}\right)\cdot\mathbf{E} + \frac{s}{2}\left(1+\xi^2\right)\frac{\mathbf{E}\times\mathbf{B}}{B^2}\cdot\nabla\ln B \\
\dot{\xi} &= \frac{1-\xi^2}{2\xi}\left\{-\xi^2\frac{\partial\ln B}{\partial t} + \left(\mathbf{v}_\parallel + \mathbf{v}_{\rm c}^*\right)\cdot\left(\frac{q\mathbf{E}}{T_0 s^2} - \nabla\ln B\right) + \xi^2\frac{\mathbf{E}\times\mathbf{B}}{B^2}\cdot\nabla\ln B\right\} \\
&\quad - \xi(1-\xi^2)\left\{\frac{\mu_0}{2B^2}\mathbf{J}_\parallel\cdot\mathbf{E} + \frac{T_0 s^2}{q}\mathbf{b}\cdot\nabla\left(\frac{\mu_0 J_\parallel}{B^2}\right)\right\} + (1-\xi^2)\frac{T_0 s}{v_0 qB}\nabla\cdot\left(\mathbf{b}\times\frac{\partial\mathbf{b}}{\partial t}\right)
\end{aligned}$$

This agrees exactly with Hazeltine's DKE and includes "twist" term which stems from $\dot{\mu}$.

Chapman-Enskog-like DKEs in NIMROD

$$\frac{\partial \bar{f}_{\mathrm{NM}}}{\partial t} + v'_{\|}\mathbf{b}\cdot\nabla \bar{f}_{\mathrm{NM}} - \frac{1-\xi^2}{2\xi} v'_{\|}\mathbf{b}\cdot\nabla \ln B \frac{\partial \bar{f}_{\mathrm{NM}}}{\partial \xi}$$

$$+\frac{v_0}{2}(\mathbf{b}\cdot\nabla \ln n)[\xi \frac{\partial \bar{f}_{\mathrm{NM}}}{\partial s} + \frac{1-\xi^2}{s}\frac{\partial \bar{f}_{\mathrm{NM}}}{\partial \xi}] - s[\xi \mathbf{b}\cdot\nabla + \frac{\partial}{\partial t}]\ln v_0 \frac{\partial \bar{f}_{\mathrm{NM}}}{\partial s} = \langle C(f)\rangle$$

$$+\Bigg[(\frac{5}{2}-s^2)v'_{\|}\mathbf{b}\cdot\nabla \ln T + \frac{v'_{\|}}{nT}\mathbf{b}\cdot[\frac{2}{3}\nabla\pi_{\|} - \pi_{\|}\nabla \ln B - \mathbf{F}^{\mathrm{coll}}]$$

$$+2s^2(\frac{3}{2}\xi^2 - \frac{1}{2})[\frac{1}{3}\nabla\cdot\mathbf{u} - \mathbf{bb}\cdot\nabla\mathbf{u}] + \frac{2}{3nT}(s^2 - \frac{5}{2})[\mathbf{b}\cdot\nabla q_{\|} - q_{\|}\mathbf{b}\cdot\nabla \ln B - G^{\mathrm{coll}}]$$

$$+\frac{2}{3eB}s^2(\frac{3}{2}\xi^2 - \frac{1}{2})[(\frac{5}{2} - s^2)(\nabla \ln B - 2\kappa) + \nabla \ln n]\cdot\nabla T\times\mathbf{b}$$

$$+\frac{4}{3eB}(\frac{s^4}{2} - \frac{5}{2}s^2 + \frac{15}{8})(\nabla \ln B + \kappa)\cdot\nabla T\times\mathbf{b}\Bigg] f_M$$

Continuum representation for f

For toroidal computations, distribution functions are expanded in NIMROD-esque fashion:

$$f(R, Z, \phi, \xi, s, t) =$$

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

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

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

Code development lines

Until two days ago, kinetic capability existed in several branches.

- **electron_cel**: implementation of CEL-DKEs started in summer of 2016.
  Required testing of fluid/coupling so made sense to start a branch.
- **newpart**: development of δf-PIC algorithm with map_mod capability.
  Processors have global field information to push particles throughout domain.
- **nimdevel**: where everything should end up.
  Successfully merged of all capability back into nimdevel.

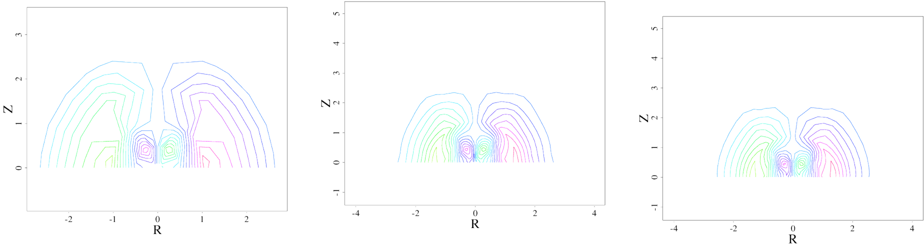
Application of δf-DKE

- Preparing paper on continuum energetic particle algorithm.
- Preparing companion paper comparing continuum and δf-PIC (newpart branch) approaches.
- Assisting in poloidal flow damping studies (verification against CEL-DKE version to come).
- Beginning global equilibrium/neoclassical transport calculations.

Implemented remapping capability for f

Bootstrap up to higher velocity-space resolution in linear scans and neoclassical transport calculations.

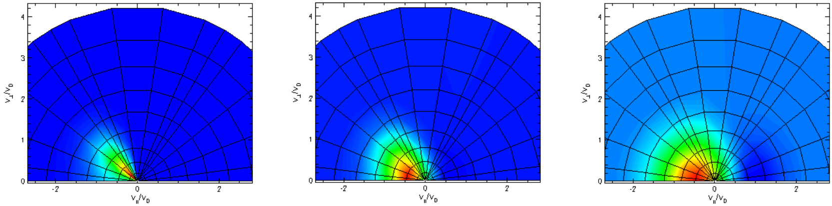
Lower resolution (ns, pdxi=5, left) mapped to higher resolution (ns, pdxi=9, center). Actual ns, pdxi=9 result at right.



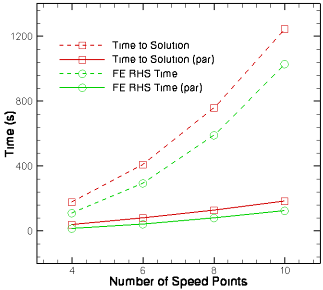
Implemented parallelization over speed coordinate

- For linear energetic particle calculations, parallelization over speed is trivial.
- For nonlinear calculations, matrix-vector products and right-hand-side computations done by processors working at single s.
- s loop moved outside others in integrand routines.
-
```
!---------------- ! sum terms in DKE. !-------------
DO is=1,ns(2)
   IF (ss_rhs.AND.is/=nss) CYCLE
   s = xg_s(2,is)
   DO imode=1,nmodes
      DO iv=1,nv
         ..........
```
- Additional optimization possible using BLAS routines for matrix-vector operations.

Parallelization over s used in flow damping studies.



- Plot shows the decrease in time to solution and matrix/vector product time (FE RHS) between un-parallelized (dashed lines) and parallelized (solid lines, par) kinetic algorithms.



Continuum CEL-DKE/T coupling implemented

Temperature evolution with kinetic closure for parallel heat flow:

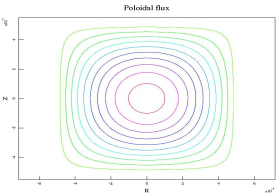
$$\frac{3}{2}n\frac{\partial T}{\partial t} = \kappa_{\perp}\nabla\cdot[(\mathbf{I}-\mathbf{bb})\cdot\nabla T] - \nabla\cdot\mathbf{q}_{\|} + Q$$

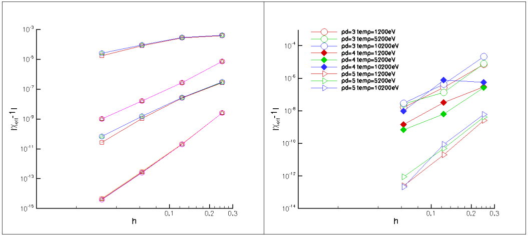
$$\mathbf{q}_{\|} = \frac{m}{2}\int d\mathbf{v}v^2 v_{\|}F = \pi m v_T^6\int_{-1}^{1}d\xi\int_0^{\infty}ds\left(s^5\xi F\right)$$

$$\frac{\partial F}{\partial t} + \mathbf{v}_{\|}\cdot\nabla F - \frac{1-\xi^2}{2\xi}\mathbf{v}_{\|}\cdot\nabla\ln B\frac{\partial F}{\partial\xi} - \frac{s}{2}\left(\mathbf{v}_{\|}\cdot\nabla + \frac{\partial}{\partial t}\right)\ln T\frac{\partial F}{\partial s}$$
$$= C + \left(\frac{5}{2}-s^2\right)\mathbf{v}_{\|}\cdot\nabla\ln T f_M + \frac{2}{3nT}\left(s^2-\frac{3}{2}\right)\left(\nabla\cdot\mathbf{q}_{\|} - Q\right)f_M$$

Application to JCP anisotropic conduction case

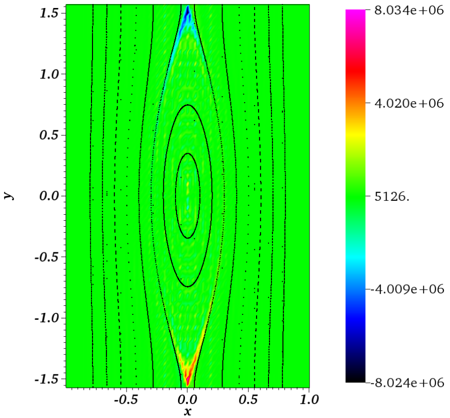
Test accuracy of $\mathbf{b} \cdot \nabla$in anisotropic conduction and CEL-DKE.





Kinetic heat flux from simultaneous advance of T and F

- Objective: take as large time steps as possible to get to steady state with kinetic parallel heat flux
- Using Chapman-Enskog like (CEL) method applied to drift-kinetic equation
- Advance T and F simultaneously
- Using Newton or Picard iterations to converge on nonlinear solution at each time step
- Test problem: magnetic island in slab geometry



Ongoing work on continuum kinetics

Since Sherwood meeting

- Implemented s-parallelism for simultaneous advance
  - Preconditioner for simultaneous F and T advance diagonal in speed points with additional submatrix preconditioning T equation
  - Groups of processors split work to compute and invert diagonal submatrices
  - Also split work to compute right hand side and matrix-vector dot product
- Merged Newton iterations and s-parallelism into trunk
- Found instability: ran slab island case with Newton iterations successfully for 6.73 ms with 1 μs time steps before becoming unstable

Future work

- Implement full speed preconditioning in simultaneous advance
- Possibly speed-up Newton iterations
- Adaptive time step