M3D: Stellarator Simulations, Parallelization, and Resistive Wall

H.R. Strauss
New York University, New York, New York

W. Park, X. Tang, G. Y. Fu,
A. Pletzer, I. Szczesniak
Princeton University Plasma Physics Laboratory, Princeton, New Jersey
L.E. Sugiyama
Massachusetts Institute of Technology, Cambridge, Massachusetts

- M3D Stellarator Simulations
- Integration of M3D / Parm3d
- GRIN

1. Stellarator Simulations

M3D is initialized with VMEC, an ideal MHD equilibrium code. To compare with PIES, we want resistive equilibria with islands. Solving with full pressure equation, including parallel smoothing, gives resistive ballooning.

Source terms $J_{\phi 0}, p_0$ are included to maintain the toroidal current and pressure.

$$E_\phi + (\mathbf{v} \times \mathbf{B})_\phi = \eta(J_\phi - J_{\phi 0})$$

$$\frac{\partial p}{\partial t} = -\mathbf{v} \cdot \nabla p - (\gamma - 1) p \nabla \cdot \mathbf{v} - \nabla_\parallel v_{a\parallel} + \kappa_\perp \nabla_\perp^2 (p - p_0)$$

$$\frac{\partial v_{a\parallel}}{\partial t} = -c_a^2 \nabla_\parallel p + D_a \nabla_\perp^2 v_{a\parallel}$$

Pressure is advanced by both advection and "artificial sound" to relax towards a state with

$$\nabla_\parallel p = 0.$$

(HINT) omit pressure advection. Apply artificial sound every n_a timesteps ($100 \geq n_a \geq 1$). Convergence criterion

$$|\nabla_\parallel p| / |\nabla_\perp p| < \delta.$$

Examples:

(a) resistive ballooning in li383

(b) (5,3) island with "artificial sound" only.

2. Integration of M3D / Parm3d

Purpose: to reuse physics code in M3D not implemented in ParM3D

M3D

- Physics Driver
  - advances MHD equations
  - 2-fluid, neoclassical, particles
- Mesh Modules
  - 1D finite - difference, 2D spectral
  - 2D poloidal (RZ) unstructured mesh (linear triangles & bilinear rectangles), spectral toroidal
  - 2D poloidal unstructured mesh, toroidal finite difference
    - OMP (shared memory) toroidal domains
    - MPI (distributed memory) interface to ParM3D

- Physics Driver - M3D
- MPI Mesh Module - ParM3D
  - Petsci library of parallel structures (Vec, Mat) and solvers
  - initialization
    - initialize communications
    - read VMEC equilibrium data
    - set up grid
      - 2D (R,Z) unstructured & toroidal structured mesh
      - poloidal & toroidal domain decomposition
    - calculate finite element matrices
  - routines called from physics driver
    - differential and integral operators (grad, div, curl),
    - elliptic eqs. ($\nabla^2_\perp, \Delta^*$, ...)
  - parallel I/O
- validation

3. GRIN

- plasma surrounded by a toroidally symmetric vessel
- vacuum beyond vessel
- vessel's conductivity σ $\times$ thickness $\delta \sim$ finite

Compute the vacuum 'impedance' $\mathcal{Z} \equiv$ linear matrix relating the normal magnetic perturbation $\mathbf{B}_n$ to the tangential component $\mathbf{B}_t$.

- $\mathbf{B}_n$ continuous across shell (thin wall approximation)
- jump $[[\mathbf{B}_t]] \approx \sigma\delta \;\; \mathbf{E} \times \mathbf{n}$

Natural boundary conditions involve $\mathbf{E}\times\mathbf{n}$ at the plasma edge.

From $\mathbf{B}_n \rightarrow \mathbf{B}_t^{outside} \rightarrow [[\mathbf{B}_t]] \leftarrow \mathbf{B}_t^{inside}$.

- M3D provides $\mathbf{B}_n$ and $\mathbf{B}_t^{inside}$
- GRIN computes $\mathcal{Z}$
- $\mathbf{B}_t^{outside} = \mathcal{Z}\mathbf{B}_n$

The $\mathcal{Z}$ matrix depends on the shell geometry only. It is computed before launching a nonlinear M3D calculation.

B in vacuum

- poloidal and toroidal slit $\Rightarrow m = 0$ and $n = 0$ $\mathbf{B}$ components are continuous
- no external $n \neq 0$ currents
- $\boldsymbol{\nabla}\cdot\mathbf{B} = 0 \Rightarrow \mathbf{B} = \nabla\lambda$
- $\nabla^2\lambda = 0$
- $\lambda = \Sigma_{n=1}^{N} \lambda_n \exp in\phi$

Green's functions

From Green's identity get integral equation relating λ_n to given $\partial\lambda_n/\partial n = \mathbf{B}_n$ on boundary contour, following Chance [*Phys. Plasmas* **4**, 2161 (1997)]. GRIN calculates 'impedance' $\mathcal{Z}$ matrix elements

- $\int d\ell' K(\ell, \ell')\alpha'(\ell')$
- $\int_s d\ell\alpha(\ell) \int_{s'} d\ell' K(\ell, \ell')\alpha'(\ell')$

for arbitrary open or closed segments s and s', and K a user-supplied kernel with a singularity as $\ell \rightarrow \ell'$.