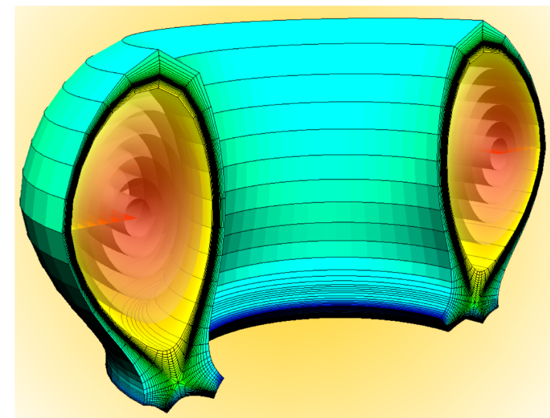
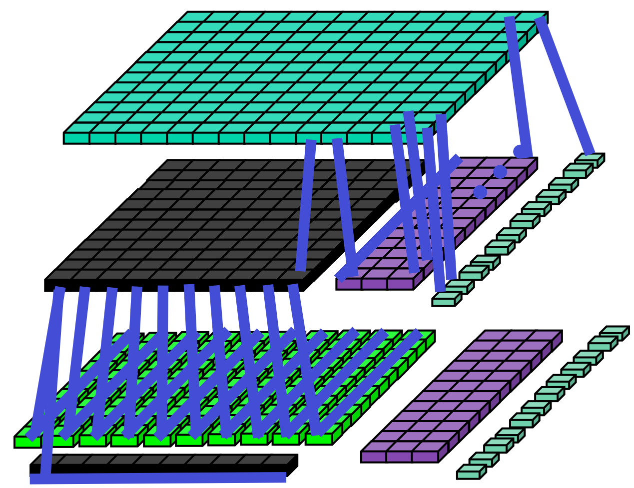
Dynamic Equilibrium Discussions

S. Kruger, L. LoDestro, D. McCune,
A. Pletzer, L. LoDestro



- Overview of fluxmappers
- Brief discussion of refactoring and babelization
- How it fits into dynamic equilibria development
- How do we fix EqMagGeom in FMCFM?



Core Transport Equations

Ion species density

$$\left[\frac{1}{V'}\frac{\partial}{\partial t}V' + \dot{\rho}\frac{\partial}{\partial \rho}\right] n \quad + \frac{1}{V'}\frac{\partial}{\partial \rho}\left[V' \ \Gamma_n\right] = \left\langle S_n \right\rangle$$

N_i equations
(N_i=# of ion species)

Species energy

$$\left[\frac{1}{V'^{5/3}}\frac{\partial}{\partial t}V'^{5/3} + \dot{\rho}\frac{\partial}{\partial \rho}\right] \frac{nT}{\gamma - 1} + \frac{1}{V'}\frac{\partial}{\partial \rho}\left[V' \ \Gamma_E\right] = \left\langle Q_{net} \right\rangle$$

N_i +1 equations

Total angular momentum

$$\left[\frac{1}{V'}\frac{\partial}{\partial t}V' + \dot{\rho}\frac{\partial}{\partial \rho}\right] L_T \quad + \frac{1}{V'}\frac{\partial}{\partial \rho}\left[V' \ \Gamma_\Omega\right] = \left\langle S_\Omega \right\rangle$$

1 equation

Poloidal flux

$$\frac{\partial}{\partial t}\psi_p \qquad - \frac{\eta_\parallel^{nc}}{\mu_0}\Delta^+\psi_p = \left\langle S_\psi \right\rangle$$

1 equation

3 + N_i equations

$$\Delta^*\psi = -R^2\mu_0\frac{\partial p}{\partial \psi} - F\frac{\partial F}{\partial \psi}$$

2D Equilibria

And neutrals equation which is not shown (potentially 2D)

1D Core Transport Equations

$$\left[\frac{1}{V'}\frac{\partial}{\partial t}V' + \dot{\rho}\frac{\partial}{\partial \rho}\right] n + \frac{1}{V'}\frac{\partial}{\partial \rho}\left[V' \Gamma_n\right] = \langle S_n \rangle$$

$$\left[\frac{1}{V'^{5/3}}\frac{\partial}{\partial t}V'^{5/3} + \dot{\rho}\frac{\partial}{\partial \rho}\right] \frac{nT}{\gamma - 1} + \frac{1}{V'}\frac{\partial}{\partial \rho}\left[V' \Gamma_E\right] = \langle Q_{net} \rangle$$

$$\left[\frac{1}{V'}\frac{\partial}{\partial t}V' + \dot{\rho}\frac{\partial}{\partial \rho}\right] L_T + \frac{1}{V'}\frac{\partial}{\partial \rho}\left[V' \Gamma_\Omega\right] = \langle S_\Omega \rangle$$

$$\frac{\partial}{\partial t}\psi_p - \frac{\eta_{\parallel}^{nc}}{\mu_0}\Delta^{*}\psi_p = \langle S_\psi \rangle$$

Sources

Equilibrium

Core Profiles
(dependent variables)

Transport

1D Core Transport Equations

$$\left[\frac{1}{V'}\frac{\partial}{\partial t}V' + \dot{\rho}\frac{\partial}{\partial \rho}\right] n + \frac{1}{V'}\frac{\partial}{\partial \rho}\left[V' \Gamma_n\right] = \langle S_n \rangle$$

$$\left[\frac{1}{V'^{5/3}}\frac{\partial}{\partial t}V'^{5/3} + \dot{\rho}\frac{\partial}{\partial \rho}\right] \frac{nT}{\gamma - 1} + \frac{1}{V'}\frac{\partial}{\partial \rho}\left[V' \Gamma_E\right] = \langle Q_{net} \rangle$$

$$\left[\frac{1}{V'}\frac{\partial}{\partial t}V' + \dot{\rho}\frac{\partial}{\partial \rho}\right] L_T + \frac{1}{V'}\frac{\partial}{\partial \rho}\left[V' \Gamma_\Omega\right] = \langle S_\Omega \rangle$$

$$\frac{\partial}{\partial t}\psi_p - \frac{\eta_{\parallel}^{nc}}{\mu_0}\Delta^{*}\psi_p = \langle S_\psi \rangle$$

Sources

Dynamic Equilibria

Equilibrium

Core Profiles (dependent variables)

Transport

TECH-X CORPORATION

1D Core Transport Equations

$$\left[\frac{1}{V'}\frac{\partial}{\partial t}V' + \dot{\rho}\frac{\partial}{\partial \rho}\right] n + \frac{1}{V'}\frac{\partial}{\partial \rho}\left[V' \Gamma_n\right] = \langle S_n \rangle$$

$$\left[\frac{1}{V'^{5/3}}\frac{\partial}{\partial t}V'^{5/3} + \dot{\rho}\frac{\partial}{\partial \rho}\right] \frac{nT}{\gamma - 1} + \frac{1}{V'}\frac{\partial}{\partial \rho}\left[V' \Gamma_E\right] = \langle Q_{net} \rangle$$

$$\left[\frac{1}{V'}\frac{\partial}{\partial t}V' + \dot{\rho}\frac{\partial}{\partial \rho}\right] L_T + \frac{1}{V'}\frac{\partial}{\partial \rho}\left[V' \Gamma_\Omega\right] = \langle S_\Omega \rangle$$

$$\frac{\partial}{\partial t}\psi_p - \frac{\eta_\parallel^{nc}}{\mu_0}\Delta^*\psi_p = \langle S_\psi \rangle$$

Sources

Dynamic Equilibrium

Hyperviscosity should be added to flux eq.

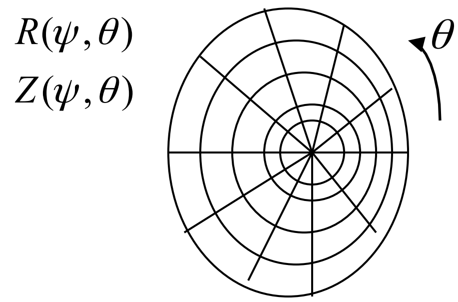
Equilibrium

Core Profiles (dependent variables)

Transport

In core: two approaches exist for Grad-Shafranov Equation

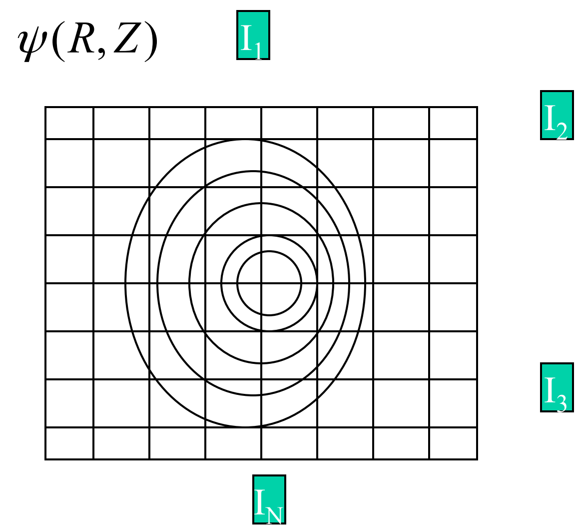
- **Inverse solver:**
  - **Define R(ψ,θ), Z(ψ,θ) (only in core obviously)**
  - **Use mapping relations to re-express Grad-Shafranov operator in terms of these variables**



Do this first

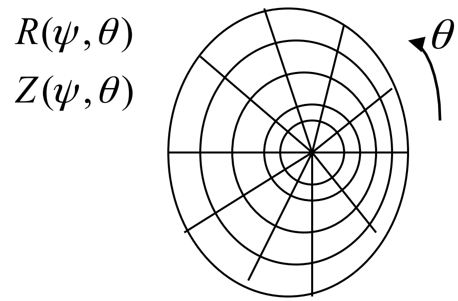
- **Direct solver**
  - **Traditional GS equation**
  - **Coils and being able to specify shape are key to use though**

Do next



Inverse solver more straightforward

- **Inverse solver inputs:**
  - **$R_{ob}(\theta)$, $Z_{ob}(\theta)$ (often parameterized)**
  - **Pressure profile: $p(\psi)$**
  - **Another method to specify the current profile:**
    - **$F(\psi)$ (Rarely used)**
    - **$q(\psi)$ ($q=d\psi_t/d\psi_p$) (Most common)**
    - **<J.B>**
    - **…**



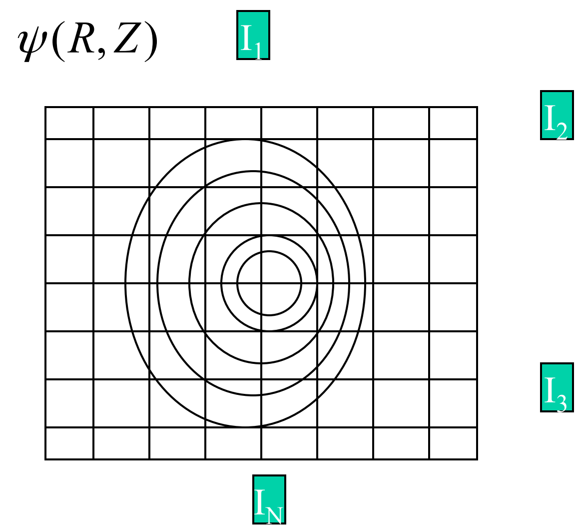
- **Common codes**
  - **TEQ inverse solver (TRANSP workhorse now)**
  - **TOQ (from GA)**
  - **ESC, JSOLVER, …**

Direct solver complicated because of coil currents

- **Direct solver inputs:**
  - $R_{ob}(\theta)$, $Z_{ob}(\theta)$ (often parameterized)
  - Pressure profile: $p(\psi)$
  - Another method to specify the current profile:
    - $F(\psi)$ (Rarely used)
    - $q(\psi)$ ($q=d\psi_t/d\psi_p$) (Most common)
    - <J.B>
    - …
  - Coil currents (solved with equations)
    - Introduces implicitness issues

- **Available codes: isolver (TRANSP), TEQ**

Machine data:
Coil current locations
(need to distinguish
between internal and
external coil currents)



Coupling to the circuits equation:

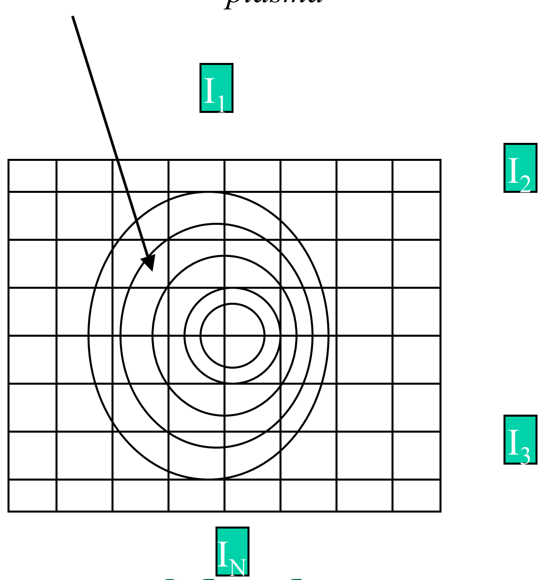
Poloidal field coils and metallic structures obey circuits equations coupling them to the plasma:

$$\frac{d}{dt}\left[L_i I_i(t) + \sum_{i\neq j} M_{ij} I_j + \int_{plasma} J(\vec{R}')G(\vec{R}_i,\vec{R}')d\vec{R}'\right] + R_i I_i = V_i, \quad i = 1, N$$

$$\psi_b(\vec{R}) = \int_{plasma} J(\vec{R}')G(\vec{R},\vec{R}')d\vec{R}' + \sum_{i=1}^{N} I_i G(\vec{R},\vec{R}_i)$$

$$G(\mathbf{R};\mathbf{R}') = \frac{\sqrt{RR'}}{2\pi k}\left[(2-k^2)K(k^2) - E(k^2)\right]$$

$$k^2 = \frac{4RR'}{\left[(R+R')^2 + (Z-Z')^2\right]}, \qquad K, E \text{ elliptic integrals}$$



Virtual casing theorem (Green's second identity) is used to turn 2D integral into 1D line integral:

$$\int_{plasma} J(\vec{R}')G(\vec{R},\vec{R}')d\vec{R}' = \oint_{boundary} \frac{d\mathit{l}}{R} G(\vec{R},\vec{R}')\frac{\partial U}{\partial n}$$

$$\Delta^* U = \Delta^* \psi, \qquad U = 0 \text{ on boundary}$$

From Jardin

Robustness issues

- There are the formal issues, and then there are the usability issues.
- In practice, "cold starts" are difficult
- Need to get a good initial guess
- TEQ comes with library of initialization files. The files contain everything (currently a ball of stuff)
  - Proposal to separate things out, but has not been done yet.
  - Lots of good data exists in TEQ but difficult to get out

Table 1: Device configuration input files

File-Name	*Page*	*File contents*
`coils.in`	14	Poloidal field coils and other toroidal current sources
`dgaps.in`	15	Diagnostic gap specifications
`limits.in`	16	Coil current, field and force limits
`params.in`	18	Nominal plasma and device parameters
`passive.in`	20	Passive structure specifications
`shape.in`	21	Target plasma boundary coordinates
`tfcoil.in`	22	Toroidal field coil specifications
`wall.in`	23	Plasma-facing first-wall, divertor and limiter geometry

Mapping issues

- "Mapping code" is used to calculate metric elements given an equilibrium
- Fluxgrid is used for core-edge simulations via file-based coupling
- New "babelized" version is under development to enable in-memory coupling
  - Currently in debugging phase

- Want to enable a common class which can use either fluxgrid or plasma_state mapper
  - Discussion needs to occur here

Status

- Eqcodes has TEQ with new facets_driver.
- Still some minor work to understand things on driver side
  - (~100 variables in interface).
- Need to figure things out on the FACETS side
  - What does our input file look like?
  - See draft

Results based on discussions

- TRANSP has some code that tries to make TEQ more robust
  - Initial investigation looks like it could be extracted without too much difficulty
  - Similar to iSolver interface so we would have both codes available through similar interfaces

- Action items:
- Componentization
  - Work with PPPL to extract TEQ interfaces and iSolver from TRANSP
  - This will go into eqcodes.
  - Facets_driver will be extended to use new interfaces
  - Eqtests set up
- Framework side:
  - Have meeting to draft input file
  - Ask Rob Andre to review TRANSP experience

Staged testing

- Get sequence of equilibria
  - Use trxpl2ps to get sequence of equilibria for shots of relevance,
  - Get them from ONETWO runs
- Use equilibria for simulations (tests the rho-dot terms)
  - Need to make sure equations conserve quantities

- Use similar case for V&V of flux evolution
  - First do fixed boundary functionality
  - Second do free boundary