Magnetic modeling and control of tokamaks, Part IV: Grad-Shafranov equations, Free boundary inverse solvers, MEQ suite of codes, FBT & LIUQE

Antoine Merle, Federico Felici

Ecole Polytechnique Fédérale de Lausanne (EPFL),
Swiss Plasma Center (SPC), CH-1015 Lausanne, Switzerland

EPFL Doctoral School Course PHYS-748, February 2025

EPFL

Outline I

Section 1

Tokamak magnetic equilibrium

Subsection 1

Magnetic field and current description

Magnetic field in a tokamak plasma

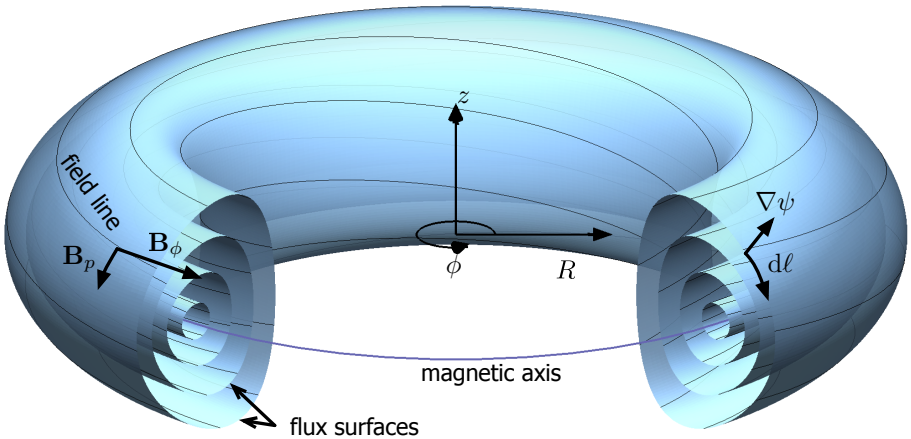
- The magnetic field is assumed to be axisymmetric $\partial/\partial\phi = 0$
- We also assume the existence of nested flux-surfaces
- We describe the *total* **B** field as toroidal + poloidal field

$$\mathbf{B} = T\nabla\phi + \mathbf{B_p}, \qquad (T = rB_\phi) \tag{1}$$

$$\mathbf{B_p} = \frac{-1}{2\pi r}\frac{\partial\psi}{\partial z}\mathbf{e_r} + \frac{1}{2\pi r}\frac{\partial\psi}{\partial r}\mathbf{e_z} = \frac{1}{2\pi}\nabla\psi\times\nabla\phi \tag{2}$$

with ψ the upward magnetic flux through a loop at constant (r, z)

- note that $\nabla\psi\cdot\mathbf{B} = 0$: *Field lines lie on flux surfaces*



Currents in the plasma

- Apply Ampère's law and some vector calculus to (1)

$$\mu_0 \mathbf{j} = \nabla \times \mathbf{B} = \nabla \times \left(T \nabla \phi + \mathbf{B_p} \right) \tag{3}$$

$$\mu_0 \mathbf{j} = \mu_0 j_\phi \mathbf{e}_\phi + \nabla \times (T \nabla \phi) = \mu_0 j_\phi \mathbf{e}_\phi + \nabla T \times \nabla \phi \tag{4}$$

with

$$\mu_0 j_\phi \mathbf{e}_\phi = \nabla \times \left(\frac{\nabla \psi}{2\pi} \times \nabla \phi \right) = -\frac{1}{2\pi r} \underbrace{r^2 \nabla \cdot \left(\frac{\nabla \psi}{r^2} \right)}_{\Delta^* \psi} \mathbf{e}_\phi \tag{5}$$

$$\Delta^* \psi = r \frac{\partial}{\partial r} \left(\frac{1}{r} \frac{\partial \psi}{\partial r} \right) + \frac{\partial^2 \psi}{\partial z^2} = -2\pi r \mu_0 j_\phi \tag{6}$$

- This relation is valid for any axisymmetric magnetic configuration. It is valid in the coils, vacuum vessel, vacuum region and plasma with no other assumption.
- It is often referred to as the *Poisson equation*.

Another interpretation for T

$$\mathbf{B} = T\nabla\phi + \frac{1}{2\pi}\nabla\psi \times \nabla\phi,$$
$$\mu_0\mathbf{j} = \mu_0 j_\phi \mathbf{e}_\phi + \nabla T \times \nabla\phi$$

- $2\pi/\mu_0 T(r,z)$ is the total upward current flowing through the loop at constant (r,z).
- This can also be seen from Ampere's law

Subsection 2

MHD model for a tokamak

MHD equations

- The Magneto HydroDynamical (or MHD) model is a magnetised fluid model for the plasma
- It describes the evolution of the velocity **v**, magnetic field **B**, current density **j**, plasma pressure tensor p (assumed to be isotropic), mass density ρ, and the electric resistivity η.

$$\frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \mathbf{v}) = 0 \quad \text{continuity} \tag{7}$$

$$\rho \left(\frac{\partial}{\partial t} + \mathbf{v} \cdot \nabla \right) \mathbf{v} = \mathbf{j} \times \mathbf{B} - \nabla p \quad \text{momentum} \tag{8}$$

$$\frac{\mathrm{d}}{\mathrm{d}t} \left(\frac{p}{\rho^{\gamma}} \right) = 0 \quad \text{Equation of state} \tag{9}$$

$$\mathbf{E} + \mathbf{v} \times \mathbf{B} = \eta(\mathbf{j} - \mathbf{j}_{ni}) \quad \text{Ohm's law} \tag{10}$$

$$\frac{\partial \mathbf{B}}{\partial t} = -\nabla \times \mathbf{E},\ \mu_0 \mathbf{j} = \nabla \times \mathbf{B},\ \nabla \cdot \mathbf{B} = 0 \quad \text{Maxwell} \tag{11}$$

MHD equations for a static equilibrium

- Let's derive the equations for a plasma at equilibrium.
- Assuming that the equilibrium is static $\mathbf{v} = 0$

$$\mathbf{j} \times \mathbf{B} = \nabla p \quad \text{momentum} \tag{12}$$

$$\mathbf{E} = \eta(\mathbf{j} - \mathbf{j}_{ni}) \quad \text{Ohm's law} \tag{13}$$

$$\nabla \times \mathbf{E} = 0, \quad \mu_0 \mathbf{j} = \nabla \times \mathbf{B}, \quad \nabla \cdot \mathbf{B} = 0 \quad \text{Maxwell} \tag{14}$$

- Note that the mass density ρ is not constrained at equilibrium.

What about transient phases in a tokamak?

- As mentioned in part 3, inertial effects act at a much shorter timescale than the timescales of interest for plasma control.
- We invoke again the idea of "instantaneous force balance", the force balance $\mathbf{j} \times \mathbf{B} = \nabla p$ is always verified.
- This is empirically confirmed during controlled operation of tokamaks.

Subsection 3

The Grad-Shafranov equation

Flux functions

- Recall the plasma force balance

$$\nabla p = \mathbf{j} \times \mathbf{B} \tag{15}$$

This implies that $\mathbf{B} \cdot \nabla p = \mathbf{j} \cdot \nabla p = 0$: the pressure gradient is perpendicular to the local current and magnetic field.

- Using the expressions for $\mathbf{j}$ and $\mathbf{B}$:

$$\mathbf{B} \cdot \nabla p = (\frac{\nabla \psi}{2\pi} \times \nabla \phi) \cdot \nabla p = \nabla \phi \cdot (\nabla p \times \frac{\nabla \psi}{2\pi}) = 0 \tag{16}$$

$$\mathbf{j} \cdot \nabla p = (\nabla T \times \nabla \phi) \cdot \nabla p = \nabla \phi \cdot (\nabla p \times \nabla T) = 0 \tag{17}$$

- We see that $\nabla p \parallel \nabla \psi$, $\nabla T \parallel \nabla p$ so $p(\psi)$, $T(\psi)$ are flux functions, i.e. constant on a flux surface.

Updated formulas for magnetic field and current

- The updated expressions for **B** and **j** are

$$\mathbf{B} = \mathbf{e}_\phi \frac{T(\psi)}{r} + \frac{1}{2\pi r}(\nabla\psi \times \mathbf{e}_\phi) \tag{18}$$

$$\mu_0 \mathbf{j} = -\mathbf{e}_\phi \frac{1}{2\pi r}\Delta^*\psi + \frac{1}{r}\frac{dT}{d\psi}(\nabla\psi \times \mathbf{e}_\phi) \tag{19}$$

The Grad-Shafranov equation

- Now use decomposition of **j** and **B** into poloidal and toroidal components:

$$\nabla p = (j_\phi \mathbf{e}_\phi + \mathbf{j}_p) \times (B_\phi \mathbf{e}_\phi + \mathbf{B}_p) \tag{20}$$

$$= -\frac{T}{\mu_0 r^2} \nabla T + j_\phi \frac{\nabla \psi}{2\pi r} \tag{21}$$

using $\nabla T = T'(\psi)\nabla\psi$ and $\nabla p = p'(\psi)\nabla\psi$:

$$p'\nabla\psi = -\frac{T}{\mu_0 r^2} T'\nabla\psi + \frac{j_\phi}{2\pi r}\nabla\psi \tag{22}$$

$$j_\phi = 2\pi \left(r p'(\psi) + \frac{1}{\mu_0 r} T T'(\psi) \right) \tag{23}$$

Combining with (5) we get the **Grad-Shafranov equation** [1], [2]:

$$\boxed{\Delta^* \psi = -4\pi^2 \left(\mu_0 r^2 p'(\psi) + T T'(\psi) \right).} \tag{24}$$

The Grad-Shafranov equation

$$\boxed{\Delta^* \psi = -4\pi^2 \left(\mu_0 r^2 p'(\psi) + TT'(\psi) \right).} \tag{25}$$

- It is a nonlinear, 2D elliptic PDE. ψ appears nonlinearly in the right hand side via the dependence of the free p and T functions.
- **Free functions?**
  - The profiles of p and T are set by the transport dynamics in the plasma. Hence when solving only the Grad-Shafranov equation, we consider them as given (or p' and TT', or some other combination).
  - Usually p and T are not a function of the local ψ value alone, but also of the value of ψ on axis ψ_A and at the boundary ψ_B. It is also necessary to discriminate locations in and out of the plasma with some mask function $\Omega_p(r, z)$.

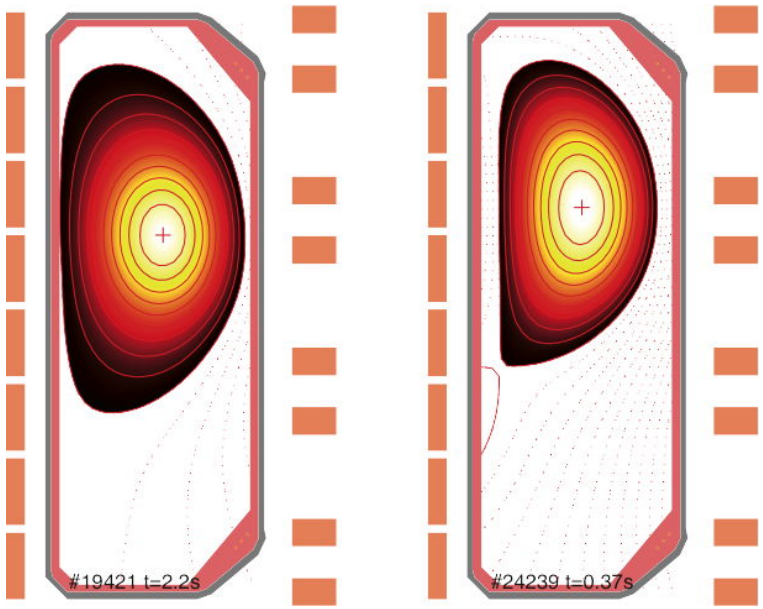
$$p(r, z) = p(\psi(r, z), \psi_A, \psi_B, \Omega_p(r, z))$$
$$T(r, z) = T(\psi(r, z), \psi_A, \psi_B, \Omega_p(r, z))$$

Subsection 4

Characteristics of tokamak equilibria

Limited vs diverted plasmas

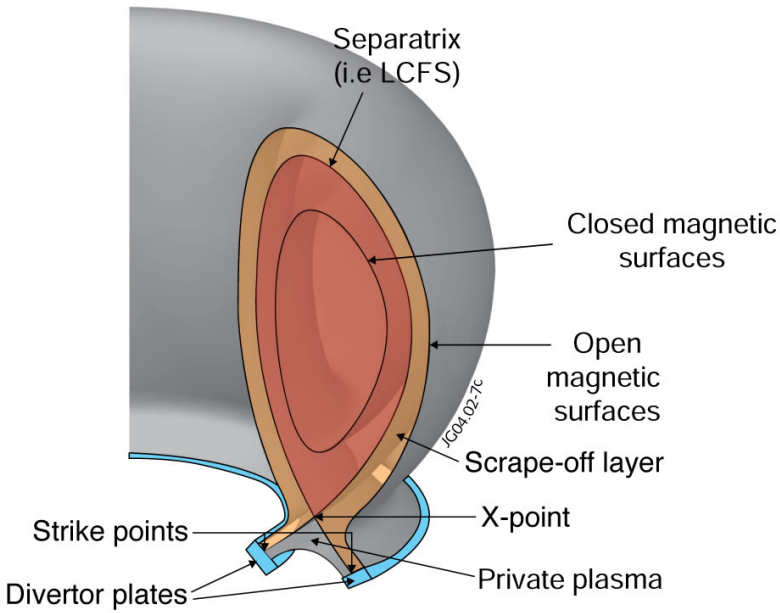
- The last closed flux surface (LCFS) is last flux surface that has closed magnetic field lines, i.e. that do not intersect the wall.

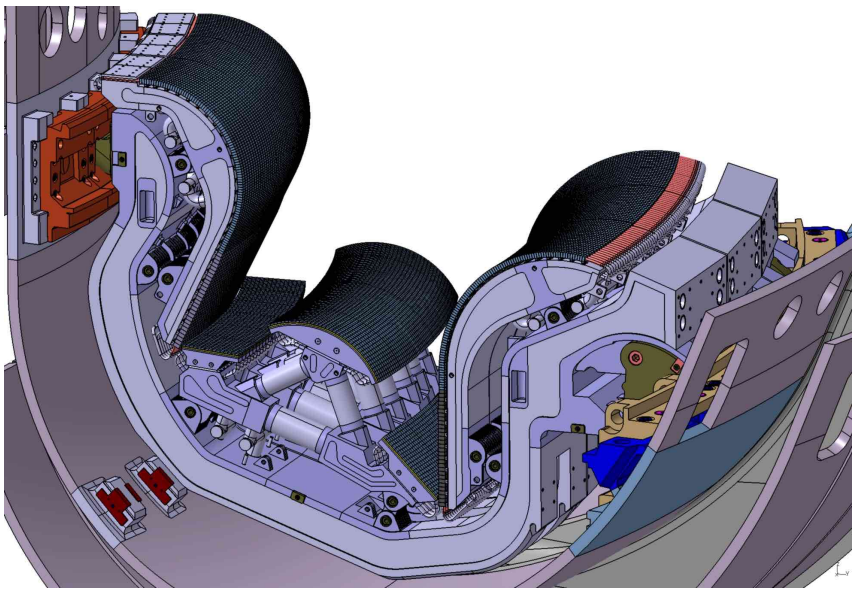


- **Limited plasma:** LCFS is determined by a direct contact between the flux surface of the plasma and the wall.
- **Diverted plasma:** LCFS is determined by the presence of an X-point (saddle point of $\psi(r, z)$) which separates closed from open field lines.

Diverted plasmas

- Diverted plasmas are good for many reasons:
  - The confined region is not in direct contact with the wall: plasma is hotter and 'cleaner'. The plasma-wall interaction is limited to a well-engineered region called the *divertor*.
  - The locally low poloidal field near the X-point ($\nabla\psi = 0$) creates toroidally elongated trajectories for particles so they can cool down before reaching the divertor.





Figures: ITER.org and EUROfusion.org

Plasma beta

- To achieve fusion we need high pressure
- A dimensionless figure of merit for high pressure in magnetic confinement devices is the plasma 'beta'

$$\beta = \frac{\text{volume-averaged kinetic pressure}}{\text{volume-averaged magnetic pressure}}$$

- Volume-averaged kinetic pressure: $\langle p \rangle_V = \frac{1}{V} \int_V p dV$.
- Volume-averaged magnetic pressure: $\langle B^2/2\mu_0 \rangle$
- Depending on a particular choice for the magnetic field to consider, various βs can be defined
  - Toroidal beta[1]: $\beta_t = \frac{2\mu_0 \langle p \rangle}{B_0^2}$
  - Poloidal beta: $\beta_p = \frac{2\mu_0 \langle p \rangle}{\mu_0^2 I_p^2 / L_p^2}$
  - Normalized beta: $\beta_N = \frac{\beta_t[\%]}{I/aB_0} \leq 2.8$ (Troyon β limit)

[1] B_0 is the vacuum toroidal field at some reference radius r_0, $r_0 = 0.88\,\mathrm{m}$ for TCV.

Internal inductance

The distribution of current in the plasma can vary depending on the equilibrium shape and the profiles of the p and T functions.

- The plasma internal inductance is defined as $L_i = 2W_i/I_p^2$ where W_i is the magnetic energy stored in the poloidal field inside the plasma volume.

$$W_i = \int_V \frac{B_p^2}{2\mu_0} dV = \frac{1}{8\pi^2\mu_0} \int_V \frac{|\nabla\psi|^2}{r^2} dV$$

- The normalised internal inductance per unit length is defined as

$$\ell_i = \frac{4\pi}{\mu_0} \frac{L_i}{2\pi r_0}$$

- Different current profiles lead to different ℓ_i values
  - Peaked current profile leads to high ℓ_i and is bad for vertical stability
  - Wide current profile leads to low ℓ_i and is bad for MHD stability

The q profile

- The magnetic field lines helically warp around the torus
- Their pitch angle will vary along the flux surface, however we can define a "mean pitch angle".

$$q = \frac{\text{number of turns in toroidal direction}}{\text{number of turns in poloidal direction}} \tag{26}$$

- More formally [3]:

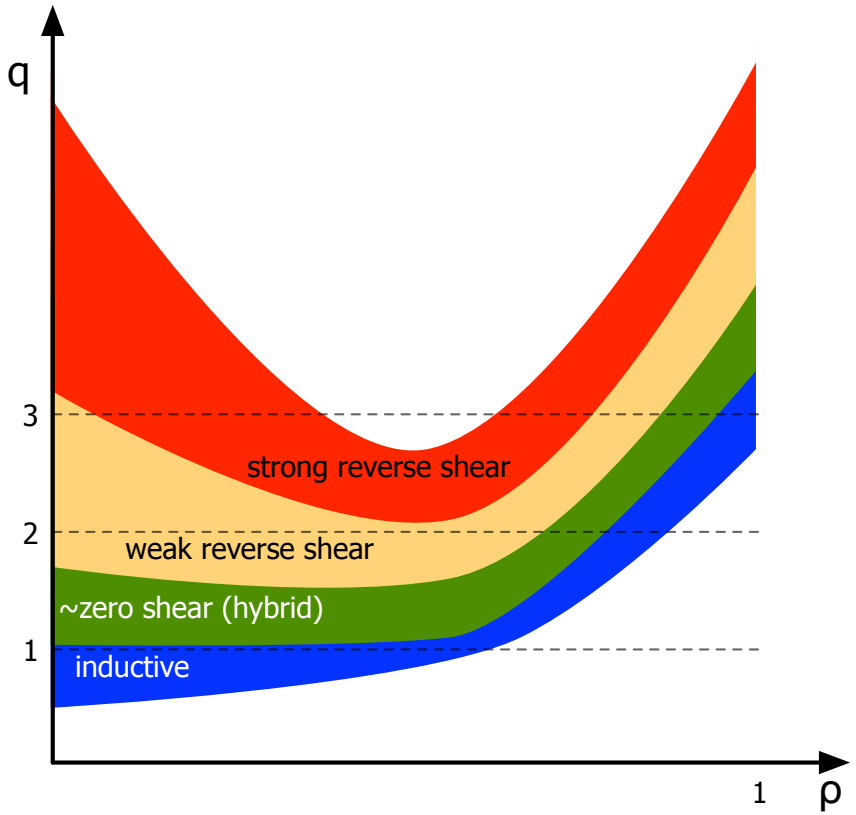
$$q(\psi) = \frac{1}{2\pi} \oint \frac{1}{r} \frac{B_\phi}{B_p} d\ell = \frac{T(\psi)}{2\pi} \oint \frac{d\ell}{r^2 B_p} \tag{27}$$

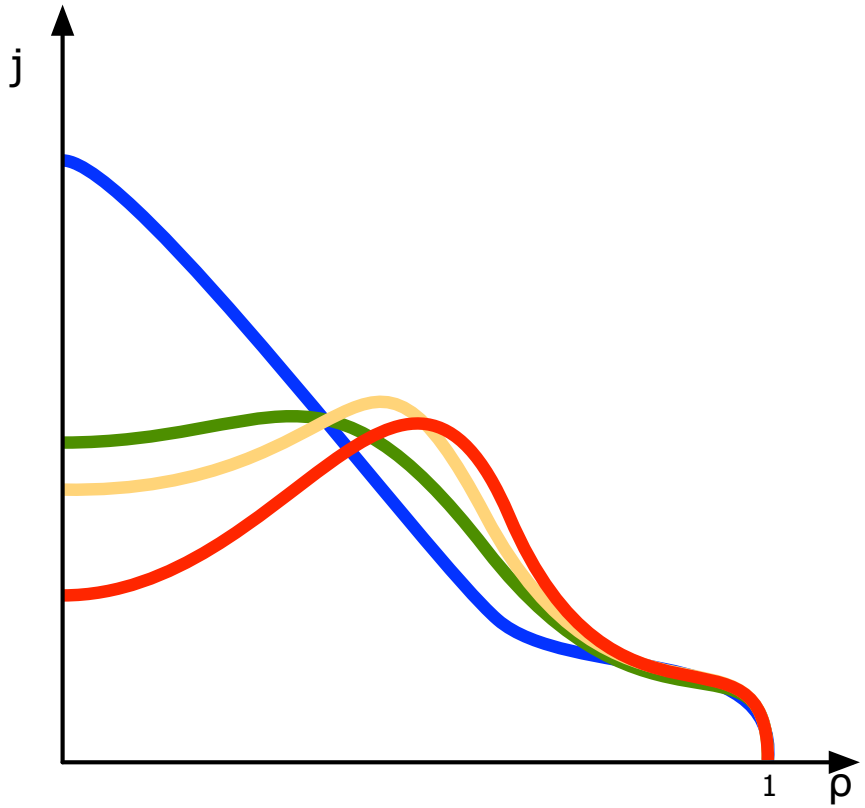
  where the integral is done over one poloidal circuit on a flux surface.
- q is a measure for the total current contained for a given toroidal field. *Engineering* q: $q_* = \frac{2\pi a^2 B_0}{\mu_0 r_0 I_p} \left(\frac{1+\kappa^2}{2} \right)$

The importance of the q profile

- The q profile plays an essential role in tokamak plasma stability. It is referred to as the 'safety factor'.
- Plasmas with $q_{edge} < 2$ are MHD unstable. It is a 'Hard' operating limit for tokamaks.
- If the plasma has regions where (locally) $q < 1$, this region will be unstable, leading to periodic internal crashes called *sawteeth*.
- At surfaces where q is rational $q = (\frac{4}{3}, \frac{3}{2}, \frac{2}{1}, \frac{3}{1}, \ldots)$, the field lines close on themselves periodically and are prone to develop magnetic instabilities known as *tearing modes*
- Plasmas with non-monotonic or 'flat' q profiles tend to have good confinement properties (more in lecture on kinetic control)

Examples of q profiles





- Current and safety factor profiles are strongly linked.

Shafranov shift

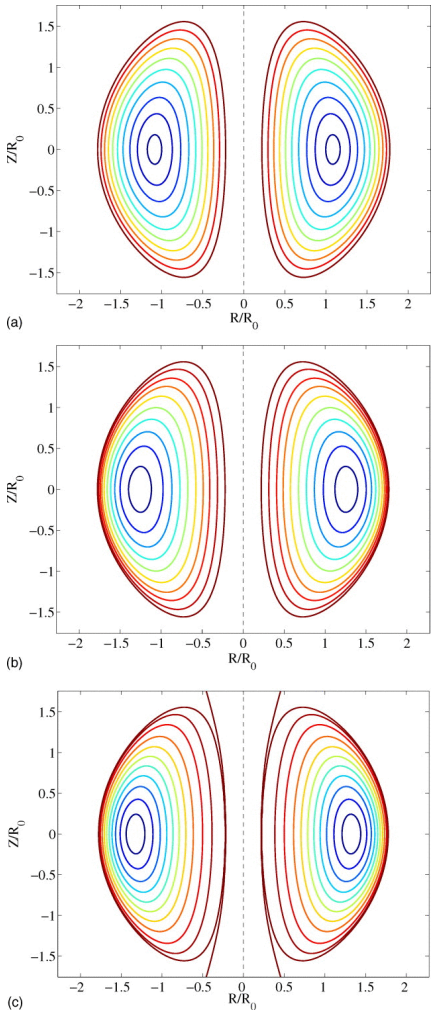
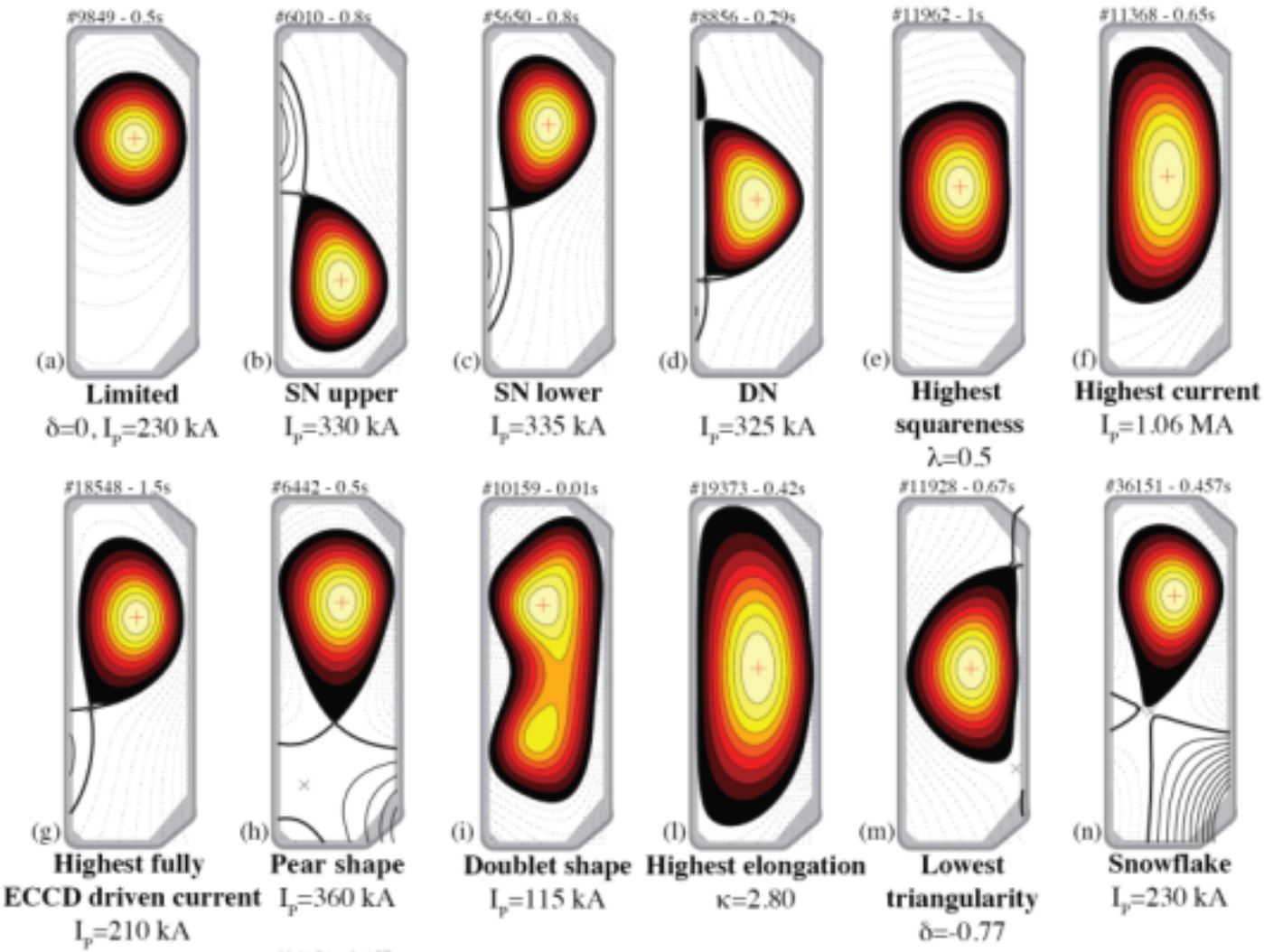


Figure: NSTX equilibria, from: Phys. Plasmas 17, 032502 (2010)

- The **magnetic axis** is defined as the point where $\psi(r, z)$ has a maximum or minimum ψ_a.
- This magnetic axis is displaced with respect to the LCFS geometric centroid. The displacement is called the **Shafranov Shift** Δ.
- Increases with plasma pressure and with internal inductance

Plasma shaping



Subsection 5

Equilibrium calculations

Solver types: fixed or free boundary

- The Grad-Shafranov equation is a non-linear 2D PDE:

$$\Delta^* \psi = -2\pi r \mu_0 j_\phi = -4\pi^2 \left(\mu_0 r^2 p'(\psi) + TT'(\psi) \right) \tag{28}$$

- Given $p(\psi)$ and $T(\psi)$ we can solve for $\psi(r, z)$.
- Several options are available to solve it:
  - **Fixed-boundary**: we prescribe the outline of the plasma boundary and we solve the GS equation strictly inside this domain using a Dirichlet condition $\psi = \psi_b$ at the boundary.
  - **Free-boundary**: the computational domain includes the whole limiter domain, we look for the solution including all conductors such that $\psi \to 0$ as $(r, z) \to \infty$.

$$\Delta^* \psi = -2\pi r \mu_0 j_\phi(p', TT', \psi) \qquad \text{inside the plasma} \tag{29}$$

$$\Delta^* \psi = 0 \qquad \text{in vacuum} \tag{30}$$

$$\Delta^* \psi = -2\pi r \mu_0 j_\phi \qquad \text{in other conductors} \tag{31}$$

Determining the plasma boundary is part of the solving process and reinforces the nonlinear character of the problem.

Solver types: forward and inverse solvers

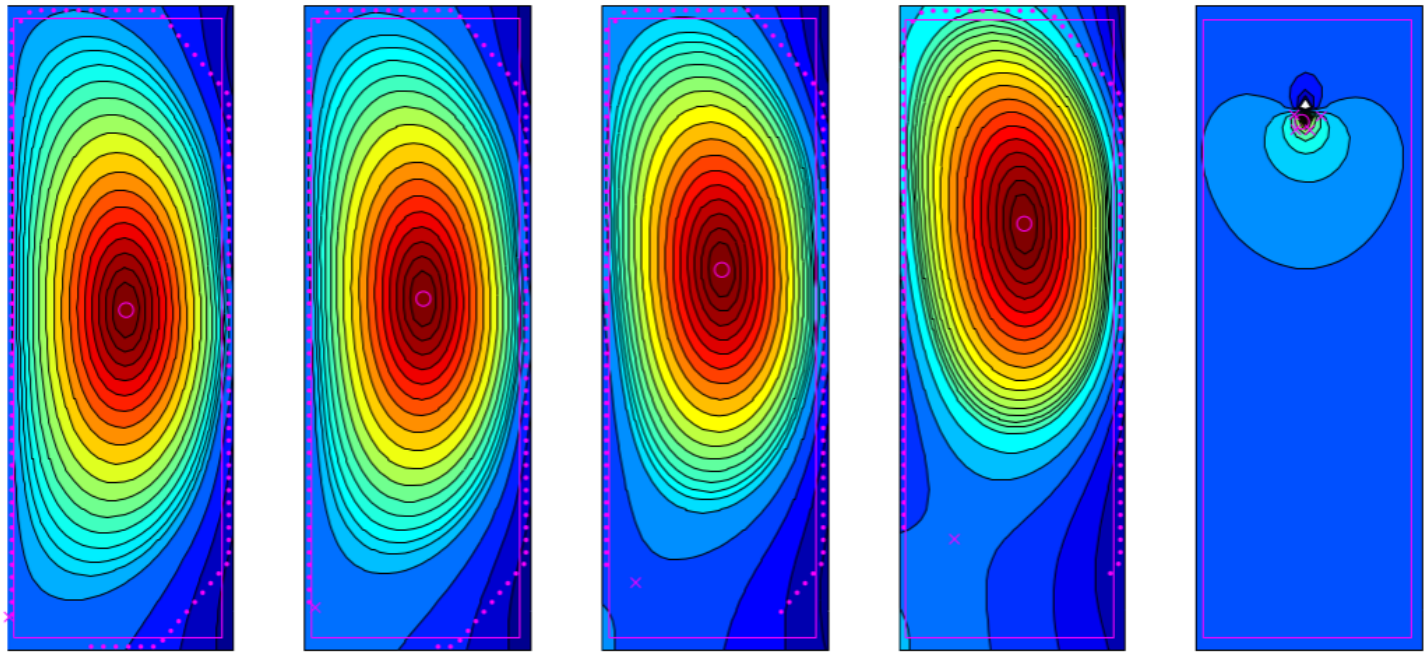
We also distinguish solvers based on the formulation of the problem

- **Forward solvers** (or predictive solvers): solve the GS problem for given external currents I_e and given functions $p(\psi)$, $T(\psi)$
  - One can also parametrise the free functions p and T and imposing some local or integral constraints for the solution (e.g. total current I_p, poloidal beta value β_p, value of q at the axis, etc.)
  - One can also extend the GS equation with the external circuit equations to model the evolution of the equilibrium (see Part 5)
  - Examples of FB forward solvers: DINA [4], CEDRES++ [5], NICE [6], CORSICA [7], MEQ-FGE [8], ...
- **Inverse solvers** (or optimisation solvers): find the GS solution that minimises a certain cost function (some examples later)
  - Again free function parametrisation can be employed as well as local or integral constraints
  - The values of the external currents are usually also part of the solution (i.e. not a given input)
  - Examples of FB inverse solvers: EFIT [9], MEQ-LIUQE [10], EQUINOX [11], NICE [6], MEQ-FBT [12], ...

Iterative Picard method

- A popular method to treat this nonlinear problem is to successively iterate between the current and the flux distributions
- At step $[n-1]$ the current distribution is $j_\phi^{[n-1]}$ in the plasma, then step $[n]$ is reached using the following steps:
  1. Solve the Poisson equation $\Delta^* \psi^{[n]} = -2\pi\mu_0 r j_\phi^{[n-1]}$ in Ω to get $\psi^{[n]}$
  2. Find the plasma domain (LCFS) $\Omega_p^{[n]}$
  3. Update the plasma current density $j_\phi^{[n]} = 2\pi \left(r p'(\psi^{[n]}) + \frac{1}{\mu_0 r} TT'(\psi^{[n]}) \right)$ in $\Omega_p^{[n]}$
  4. Test the convergence criterion based on change of j_ϕ or ψ
- A suitable initial guess for j_ϕ or ψ has to be provided.
- This can be refined by allowing an update of the external conductor currents I_e and of the plasma profiles p', TT' at each iteration.

The free-boundary Picard method is unstable for $\kappa > 1$



- Why? Shortly, the same arguments for the derivation of plasma vertical stability can be made for the stability of the Picard method.
- One common feature is the relative rigidity of the distribution of currents in the external conductors.
- Fixed-boundary solvers are stable since the “external field” reacts instantly to maintain the iso-flux surface at the domain boundary.

Common stabilisation schemes

- Allow one or more coil current to vary to maintain the plasma vertical position (add a feedback loop in your iterations)
- Add an artificial radial field to maintain the plasma vertical position
- Allow a shift between the current and flux distributions

$$j_\phi^{[n]}(R,Z) = j_\phi(p', TT', \psi(R, Z + \delta Z^{[n]}))$$

δZ is an additional free parameter. In equilibrium reconstruction algorithms it is determined from fitting the measurements data.

- All schemes result in either a modification of the original problem or a violation of the Grad-Shafranov equation.
  - Add an outer loop to minimise the artificial field

Newton method I

- A better option is to formulate the GS problem as a root finding problem $F(x) = 0$ (where x can be ψ or j_ϕ) and use a Newton method to solve it.
- Given a current estimate $x^{[n]}$, $x^{[n+1]}$ is determined by:

$$J(x^{[n]}) \cdot (x^{[n+1]} - x^{[n]}) = -F(x^{[n]})$$

  where $J = \frac{\partial F}{\partial x}$ is the Jacobian of the function F.
- This method is stable and yields excellent convergence properties compared to the Picard method (quadratic vs. linear). Many algorithms to achieve global convergence exist in the literature.
- One difficulty is the evaluation of the jacobian J which can be alleviated using Jacobian-free Newton-Krylov techniques, although analytical expressions can also be derived.
- The Picard method is a Newton method with an approximate jacobian where we have assumed $\frac{\partial j_\phi}{\partial \psi} = 0$.

Definition of F for the free-boundary GS problem

- One can simply define F to be the residual of the GS equation, including the boundary condition. The Δ^* operator is discretised using either finite differences or FEM methods (e.g. NICE, CREATE-NL).
- Another option is to find the fixed-point of a single Picard iteration. I.e. given j_ϕ
  1. Solve the Poisson equation $\Delta^*\psi = -2\pi\mu_0 r j_\phi$ to get ψ
  2. Find the plasma domain (LCFS) Ω_p
  3. Update the plasma current density

$$j_\phi^1 = 2\pi \left(r p'(\psi) + \frac{1}{\mu_0 r} T T'(\psi) \right) \text{ in } \Omega_p$$

and define $F(j_\phi) = j_\phi^1 - j_\phi$

Solving the Poisson equation

Boundary condition at the edge of the computational domain

- $\psi_b = \psi_{b,p}(j_\phi) + M_{b,e} I_e$ on $\partial\Omega$
- The contribution from the plasma current can be computed by replacing it by a sheet current j_ϕ^* at the edge of the domain that produces the same flux at this boundary.
- j_ϕ^* can actually be computed from a solution of the Poisson equation with $\psi = 0$ on $\partial\Omega$ (see [10], original idea by K. Lackner).
- We solve 2 Poisson equations at each iteration but this is faster than computing $\psi_{b,p}$ using Green functions.

Poisson solver

- The Δ^* operator is not too different from the Laplace operator so the usual methods for fast inversion work well (e.g. DFT, cyclic reduction or tri-diagonal matrix inversion).

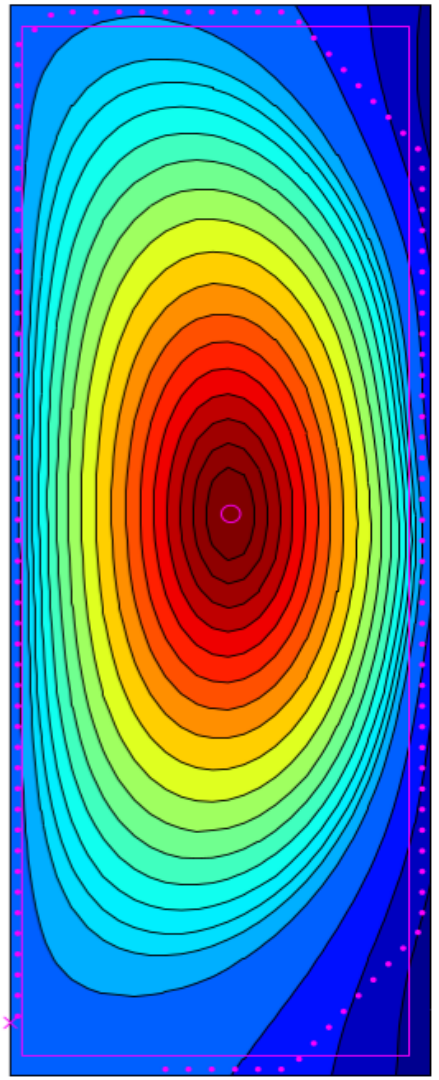
Identifying the plasma domain I

Magnetic axis

- The poloidal field is zero: $\nabla\psi = 0$
- ψ has an extremum: $\det H(\psi) > 0$
- Only points within the limiter contour are considered
- There may be several magnetic axes (doublet)
- For $I_p > 0$, the flux decreases away from the magnetic axis. The LCFS is the closed flux surface with the smallest flux.

Limited plasma

- For a limited plasma the boundary point is where the flux on the limiter is maximum (assuming $I_p > 0$ and in the absence of X-points).



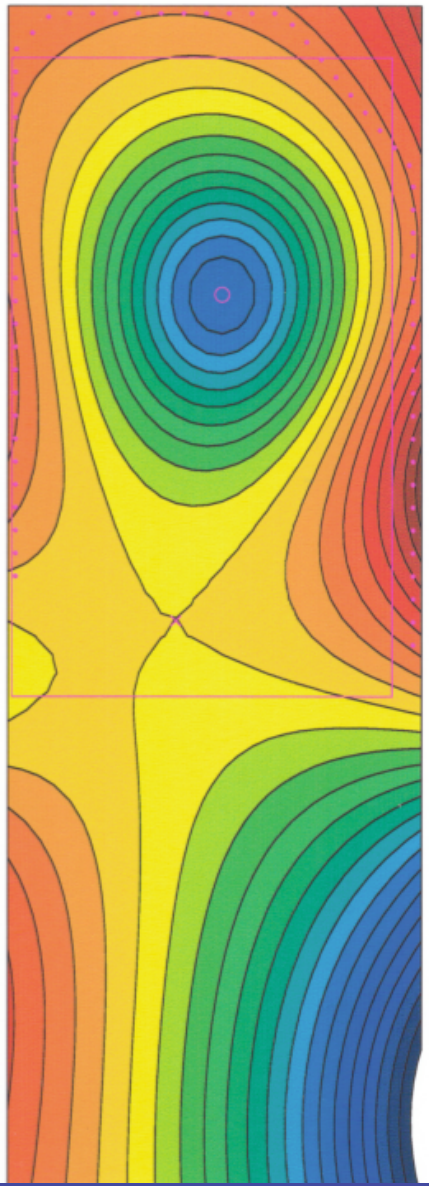
Identifying the plasma domain II

X-point(s)

- The poloidal field is zero: $\nabla\psi = 0$
- ψ has a saddle point: $\det H(\psi) < 0$
- There may be several X-points
- The active X-point defines the plasma boundary
- Following the straight line going from the magnetic axis to the X-point, ψ increases again after the X-point. This region is called the private flux region.

Diverted plasma

- The active X-point has the maximum flux compared to all limiter points and to all X-points once those in private flux regions of all X-points have been discarded.



Updating the plasma current density

- We need to compute

$$j_\phi^1 = 2\pi \left(r p'(\psi) + \frac{1}{\mu_0 r} T T'(\psi) \right) \text{ in } \Omega_p$$

but since we have imposed the functions p and T such that

$$p(r,z) = p(\psi(r,z), \psi_A, \psi_B, \Omega_p(r,z))$$
$$T(r,z) = T(\psi(r,z), \psi_A, \psi_B, \Omega_p(r,z))$$

and we have already computed all necessary quantities, this task presents no other difficulties.

Subsection 6

Inverse Equilibrium solvers

Inverse solvers: a quick peek at the numerical methods I

- Inverse solvers try to find the Grad-Shafranov solution that minimises some cost function W
- An optimality condition is added to our systems of equations. If u represents the optimisation parameters (typically the external currents I_e) then the solution also has to verify:

$$\frac{\partial W}{\partial u} + \frac{\partial W}{\partial x}\frac{\mathrm{d}x}{\mathrm{d}u} = 0 \tag{32}$$

with

$$\frac{\mathrm{d}x}{\mathrm{d}u} = -\left(\frac{\partial F}{\partial x}\right)^{-1}\frac{\partial F}{\partial u} \tag{33}$$

Inverse solvers: a quick peek at the numerical methods II

- Picard iterations can be employed but vertical stabilisation can be necessary and are slow to converge.
- Newton or Quasi-Newton methods are more efficient and stable (see e.g. [6]).
- One can also show in this case that Picard iterations are Newton iterations with an approximate Jacobian. This means that the optimality condition is only approximately verified, but in most cases the Picard and Newton solutions remain very close.

Inverse mode: PF coil current calculations I

- **Inputs:** Desired plasma boundary shape.
  Outputs: Required PF coil currents.
- This problem typically has to be solved during the shot preparation to determine the sequence of I_a to get the desired plasma shape at a sequence of pre-defined times.
- Typically we assume a nearly steady-state equilibrium: vessel currents and the transformer coil currents required for maintaining the plasma current are neglected.
- Note that this calculation automatically yields the required vertical field: GS equation *is* the force balance.
- It can also be used in the design phase of a tokamak to determine the position and size of the coils etc.

Inverse mode: PF coil current calculations II

- Cost function structure:
  - Given control points (R_c, Z_c) on our desired LCFS
  - Penalise the departure of the flux at these control points from an unknown reference value ψ_0

$$W = \sum_c \|\psi(R_c, Z_c) - \psi_0\|^2 \tag{34}$$

  - The plasma profiles p and T are chosen to satisfy some constraints based on some scalar equilibrium characteristics e.g. $I_p, q_0, W_k, \beta_p, \ell_i, \ldots$

Inverse mode: PF coil current calculations III

- Additional constraints can be added to the optimisation problem:
  - Absolute value of ψ to use shaping coils to induce V_{loop}
  - Value for B_R, B_z at (R_c, Z_c), e.g. X-points have $B_R = B_Z = 0$
  - Second derivative of ψ can be used to produce hexapoles (Snowflakes) or different levels of field-line flaring
  - Regularisation terms
    - to minimise the current in each coil (and power requirements)
    - to minimise current dipoles (minimise regions with large poloidal field and strong LCFS curvature).
  - All constraints can be made exact or have a relative weight.
  - Inequality constraints can also be used to make sure the solution stays within a predefined operational domain.
- With enough control points, the Picard iteration method can converge without vertical stabilisation.

Equilibrium reconstruction I

- **Inputs:** measurements of coil currents, flux loops, fields, ...
  Outputs: Profiles $p'(\psi)$, $TT'(\psi)$ and plasma equilibrium $\psi(R, Z)$
- This is usually a (non-linear) least squares problem using the measurements

$$W = \left\| \frac{\mathbf{y} - \mathbf{h}}{\mathbf{e}} \right\|_2^2 \tag{35}$$

  where the **y** vector contains the actual measurements, **h** the synthetic measurements and **e** the expected errors
- This is solved by so-called **equilibrium reconstruction codes**. Examples: EFIT [9], LIUQE [10], EQUINOX [11], ...
- In many tokamaks, this is done in **real-time** to determine the plasma equilibrium.

Equilibrium reconstruction II

- The initial guess for the current distribution can be obtained using a Finite Element description similar to what was seen in Part II.
- The internal profiles p' and TT' are supposed to be a linear combination of a set of basis functions

$$p'(\psi) = \sum_k \alpha_k f_k(\psi), \quad TT'(\psi) = \sum_k \beta_k g_k(\psi) \tag{36}$$

- The optimisation parameters are $(I_a, I_v, \alpha, \beta[, \delta z])$
- Magnetic measurements: magnetic probes, flux loops, I_p measurements/estimators, diamagnetic flux loop, etc.
- Others: q markers (from MHD activity), MSE (Motional Stark Effect) measurements, Faraday rotation from polarimetry, local kinetic measurements (or estimates) of p ...

Limitations of equilibrium reconstruction

- Using only external magnetic measurements, the details of the internal plasma profiles p' and TT' cannot be fully resolved
- Although the LCFS shape can be reconstructed with good accuracy, it is hard to separate the contributions from p' and TT' to the current profile, especially at low elongation.
- TCV uses a diamagnetic loop (DML) to measure the plasma contribution to T (i.e. the modification of the toroidal field due to the plasma) and thus helps disentangle the two contributions.
  - The measurement is quite challenging as $(T/T_{vacuum} - 1) \sim \beta \sim 1\%$
- On TCV standard settings for LIUQE with the magnetics and DML only include only one basis function for p' and two for TT'
- Including internal constraints helps overcome these limitations
  - local measurements of the plasma pressure or poloidal field
  - estimates of the plasma profiles from transport simulations

Real-time equilibrium reconstruction

- Typically RT constraints impose a cycle time below 1 ms
- Over this timescale the equilibrium cannot change too much. We then use this fact to speed up our computation by
  - Using the previous time step as the initial guess
  - Skipping the convergence test and doing a fixed number of iterations per time step (typically 1)
- With standard settings on TCV a single RT-LIUQE iteration takes about 200us
- This is fast enough for shape control but not vertical stabilisation.
- Coupling with RAPTOR to provide p' and TT' profiles allowed RT kinetic reconstructions [13]

Section 2

Introduction to the MEQ suite of codes

MEQ - Matlab EQuilibrium suite

- A suite of codes for solving tokamak equilibrium problems
- Main features:
  - Written in C and Matlab (Octave-compatible)
  - A “toolbox” with “onion-skin” API
    - Basic users need to know only high-level calls
    - Advanced users can use lower-level functions for their purposes
  - Unified equilibrium representation for all different equilibrium problems
  - Maximum re-use of common equilibrium processing routines across code
  - Suite of plotting, visualisation tools

MEQ - Codes in the suite

	Inverse	Reconstruction	Forward
Static	**FBT** Given (constraints on) $\psi(R,Z)$, $c(\psi,p',TT',I_a)=0$ Find I_a	**LIUQE** Given measurements, Find $\psi(R,Z)$, $p'(\psi)$, $TT'(\psi)$, I_a, I_u	**FGS** Given I_a, $c(\psi,p',TT')=0$ Find $\psi(R,Z)$
Evolutive	**N/A** Given $\psi(R,Z,t)$, $c(\psi,p',TT',t)=0$, Find $I_a(t)$, $V_a(t)$	**LIUQE** **w/ Kalman Filter** Given measurements (t), Find $\psi(R,Z,t)$, $p'(\psi,t)$, $TT'(\psi,t)$, $I_a(t)$, $I_u(t)$	**FGE** Given $V_a(t)$, $c(\psi,p',TT',t)=0$ Find $I_a(t)$,$I_u(t)$ $\psi(R,Z,t)$

$I_a(t)$, $I_u(t)$: coil, eddy currents
$V_a(t)$: applied coil voltages

Update: FBT-GSPULSE is now solving the inverse evolutive problem

MEQ - History

- In-house SPC FORTRAN codes developed by F. Hofmann to compute equilibria for highly shaped TCV (incl. doublets)
  - FBT - [14] - Inverse problem
  - LIUQE - [12] - Equilibrium reconstruction
- Rewritten in C & MATLAB (& Simulink) including real-time compatibility [10]
  - Real-time code equal to off-line, just do N iterations per time step
  - TCV grid 28x65 <200us per time step, runs routinely in TCV PCS
- LIUQE extensively tested for ITER as part of F4E-OPE883 contract (with CREATE)
  - ITER equilibrium solved on 65x128 grid <1ms
- Use experience with fast equilibrium codes to write ‘forward’ solvers using the same fast low-level routines [8]
  - FGS (Forward GS Static)
  - FGE (Forward GS Evolutive)

MEQ - Code

- Two main developers (A. Merle, F. Felici) plus important contributions from 4-6 developers in the last 5 years
- Code is hosted on `https://gitlab.epfl.ch`,
  - Systematic code review / merge request practice
  - Regular use of issue tracking
  - High test coverage, test suite run on every push
  - Regular tagged releases to TCV clusters
- Growing user base: CFS, UKAEA, EAST, IPP-Prague, ...

Licensing

- The code is provided to you for the exercise session.
- Now using an Apache 2.0 license, if you are interested in it for your work, please contact us.

Bibliography I

- Grad, H. *et al.* 1958 in *Proceedings of the 2nd UN Conf. on the Peaceful Uses of Atomic Energy* volume 31 IAEA, Geneva 190
- Shafranov, V. 1958 *Soviet Journal of Experimental and Theoretical Physics* **6** 545
- Freidberg, J.P. 2014 *Ideal MHD* Cambridge University Press
- Khayrutdinov, R.R. *et al.* 1993 *Journal of Computational Physics* **109** 193
- Hertout, P. *et al.* 2011 *Fusion Engineering and Design* **86** 1045
- Faugeras, B. 2020 *Fusion Engineering and Design* **160** 112020
- Crotinger, J.A. *et al.* 1997 *UCRL-ID-126284* Technical report LLNL
- Carpanese, F. 2021 *Development of free-boundary equilibrium and transport solvers for simulation and real-time interpretation of tokamak experiments* Ph.D. thesis Ecole Polytechnique Fédérale de Lausanne EPFL Lausanne
- Ferron, J.R. *et al.* 1998 *Nuclear Fusion* **38** 1055
- Moret, J.M. *et al.* 2015 *Fusion Engineering and Design* **91** 1
- Blum, J. *et al.* 2008 *Journal of Physics: Conference Series* **135** 12019

Bibliography II


Hofmann, F. 1988 *Computer Physics Communications* **48** 207


Carpanese, F. *et al.* 2020 *Nuclear Fusion* **60** 066020


Hofmann, F. *et al.* 1988 *Nuclear Fusion* **28** 1871