Magnetic modeling and control of tokamaks, Part V: Free boundary equilibrium evolution and control

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

1. Free boundary evolution modeling

2. Plasma discharge evolution, from breakdown to plasma termination
   - Plasma breakdown
   - Ramp-up phase
   - Flat-top
   - Ramp-down

3. Equilibrium calculation workflows for tokamak operation

Section 1

Free boundary evolution modeling

'Forward' Grad-Shafranov equilibrium problem

- In previous lectures we saw how to solve the Grad-Shafranov equation for equilibrium (re)construction problems. We distinguished two versions of an 'optimization-based' solver:
  - the *inverse problem* (FBT) where we seek an equilibrium that minimizes a cost function based on the 'desired equilibrium properties' in terms of LCFS location, strike points, etc
  - The *reconstruction problem* (LIUQE) where we seek an equilibrium that minimizes a cost function based on measurements.
- Now consider the *forward problem* of finding an equilibrium given:
  - External currents $I_e (= [I_a; I_u])$
  - The total plasma current I_p
  - Other constraints equations on moments of the internal plasma profiles (e.g. β_p, q_A, ℓ_i)

Forward Grad-Shafranov equilibrium problem

- Given I_e, I_p, c_o, we seek a plasma current distribution vector I_y and basis function coefficients a_g such that:

$$F_y(I_y, I_e, I_p, a_g, c_o) = 0 \quad \text{residual related to the GS equation} \qquad (1)$$

$$F_g(I_y, I_e, I_p, a_g, c_o) = 0 \quad \text{residual of the } I_p \text{ and constraint equations} \qquad (2)$$

- One specific choice of F_y (other choices exist):
  1. Given $I_y^{[n-1]} = j_\phi^{[n-1]}/\Delta S$ from a previous iteration, compute boundary condition $\psi_b = M_{by} I_y^{[n-1]} + M_{be} I_e$
  2. Compute new flux by inverting Laplace operator: $\Delta^* \psi^{[n]} = -2\pi R \mu_0 I_y^{[n-1]} \Delta S$ with boundary condition ψ_b
  3. Find plasma boundary and domain where $I_y^{[n]} \neq 0$
  4. Compute mapping between plasma current and basis function coefficients T_{yg}, by evaluating basis function expressions on $\psi^{[n]}$: $p' = \sum_i b_i(\psi^{[n]}) a_g^i$, $TT' = \sum_j b_j(\psi^{[n]}) a_g^j$
  5. Compute new plasma current distribution $I_y^{[n]} = T_{yg}^{[n]} a_g$
  6. Return *plasma current distribution residual* $F_y = I_y^{[n]} - I_y^{[n-1]}$

Forward Grad-Shafranov equilibrium problem

- Residual equations are computed directly from the equilibrium at the present iteration, for example if imposing I_p, β_p, ℓ_i:

$$F_g = \begin{bmatrix} I_{p,ref} - \sum_y I_y \\ \beta_{p,ref} - \beta_{p,eq}(I_y, I_e, a_g) \\ \ell_{i,ref} - \ell_{i,eq}(I_y, I_e) \end{bmatrix} \qquad (3)$$

Forward Grad-Shafranov equilibrium problem

- We have a problem of the form

$$F(x) = 0 \tag{4}$$

with unknowns $x = \begin{bmatrix} I_y \\ a_g \end{bmatrix}$.

- Solve using Newton method, iterating

$$x^{[n]} = x^{[n-1]} - \left(\frac{\partial F}{\partial x}\right)^{-1} F(x^{[n-1]}) \tag{5}$$

  - Construct full Jacobian by Finite Differences or analytical expressions
  - Jacobian-Free Newton-Krylov method [1]: Find Newton step direction by approximating the column space of the Jacobian.
- This is implemented in the `FGS` code in the `MEQ` suite

The plasma equilibrium response matrix

- Once the solution is found, we can construct the *plasma equilibrium response matrices*

$$\frac{\partial I_y}{\partial I_e} \quad \text{response to variation in external currents} \tag{6}$$

$$\frac{\partial I_y}{\partial I_p} \quad \text{response to variation in plasma current} \tag{7}$$

$$\frac{\partial I_y}{\partial c_o} \quad \text{response to variation in internal constraints} \tag{8}$$

- These can be obtained by finite differences, or if Jacobians are known, by rewriting

$$0 = \begin{bmatrix} \frac{\partial F}{\partial I_y} & \frac{\partial F}{\partial a_g} \end{bmatrix} \begin{bmatrix} \delta I_y \\ \delta a_g \end{bmatrix} + \begin{bmatrix} \frac{\partial F}{\partial I_p} & \frac{\partial F}{\partial c_o} & \frac{\partial F}{\partial I_e} \end{bmatrix} \begin{bmatrix} \delta I_p \\ \delta c_o \\ \delta I_e \end{bmatrix} \tag{9}$$

in the form $\delta I_y = \frac{\partial I_y}{\partial I_e}\delta I_e + \frac{\partial I_y}{\partial c_o}\delta c_o + \frac{\partial I_y}{\partial I_p}\delta I_p$

Free-boundary Grad-Shafranov evolution

- So far we assumed I_e, I_p are given: this makes this a *static* problem.
- In reality I_e, I_p will evolve in response to voltages, following Farday/Ohm's law
- Add a circuit equation, and discretize

$$M_{ee}\dot{I}_e + R_{ee}I_e + M_{ey}\dot{I}_y = V_e \tag{10}$$

- Add a plasma current evolution equation

$$\frac{I_y^T}{I_p}M_{yy}\dot{I}_y + \frac{I_y^T M_{ye}}{I_p}\dot{I}_e + R_p I_p = 0 \tag{11}$$

- Discretize:

$$M_{ee}(I_e^k - I_e^{k-1}) + \Delta t R_{ee} I_e^k + M_{ey}(I_y^k - I_y^{k-1}) = \Delta t V_e \tag{12}$$

$$\frac{I_y^T}{I_p}M_{yy}(I_y^k - I_y^{k-1}) + \frac{I_y^T M_{ye}}{I_p}(I_e^k - I_e^{k-1}) + \Delta t R_p I_p^k = 0 \tag{13}$$

Free-boundary Grad-Shafranov evolution

- Extended system
$$F(x^k) = 0 \tag{14}$$
with unknowns $x^k = \begin{bmatrix} I_y^k \\ a_g^k \\ I_p^k \\ I_e^k \end{bmatrix}.$
- Solve using similar JFNK or other techniques (Stabilized Picard, etc)
- This is done using the `FGE` code in the `MEQ` suite

Linearized deformable plasma evolution model

- Recall the circuit equation with a generic induction term due to the plasma:

$$M_{ee}\dot{I}_e + R_{ee}I_e + \dot{\psi}_{ep} = V_e \tag{15}$$

with $\psi_{ep} = \frac{d}{dt}(M_{ey}I_y) = M_{ey}\frac{d}{dt}(I_y)$. This expression works for any time-varying change of plasma current, not only rigid ones.
- In part III, we parametrized the plasma current distribution using the rigid body assumption as $I_y = I_y(R_p, Z_p, I_p)$.
- Instead, we now keep the general form $I_y = I_y(I_e, I_p, c_o)$ where I_e are the external currents, I_p the plasma current, and c_o any externally imposed profile constraints (e.g. β_p,q_A,ℓ_i).
- We can again linearize using the plasma response matrices (6)-(8)

$$\dot{I}_y = \underbrace{\dot{I}_{y0}}_{=0} + \frac{\partial I_y}{\partial I_e}\delta\dot{I}_e + \frac{\partial I_y}{\partial I_p}\delta\dot{I}_p + \frac{\partial I_y}{\partial c_o}\delta\dot{c}_o \tag{16}$$

Linearized deformable plasma evolution model

- Similarly to the rigid model, we realize that because we assume $\dot{I}_{y0} = 0$, this implies $\frac{\partial I_y}{\partial I_e}\dot{I}_{e0} = 0$. Hence $\frac{\partial I_y}{\partial I_e}\delta\dot{I}_e = \frac{\partial I_y}{\partial I_e}(\dot{I}_{e0}(t) + \delta\dot{I}_e) = \frac{\partial I_y}{\partial I_e}\dot{I}_e$
- Collecting terms yields:

$$(M_{ee} + X_{ee})\dot{I}_e + (M_{ep} + X_{ep})\dot{I}_p + X_{eo}\delta\dot{c}_o + R_{ee}I_e = V_e \tag{17}$$

where:

- $X_{ee} = M_{ey}\frac{\partial I_y}{\partial I_e}$
- $M_{ep} = M_{ey}\frac{I_{y0}}{I_{p0}}$
- $X_{ep} = M_{ey}\frac{\partial I_y}{\partial I_p} - M_{ep}$
- $X_{eo} = M_{ey}\frac{\partial I_y}{\partial c_o}$

Generalized rigid plasma evolution model

- Now consider the circuit equation for the plasma current:

$$\dot{\psi}_y + R_{yy} I_y = 0 \tag{18}$$

with

$$\dot{\psi}_y = M_{yy} \dot{I}_y + M_{ye} \dot{I}_e \tag{19}$$

- Again parametrizing $I_y = I_y(I_e, I_p, c_o)$, linearizing as in (16), multiplying from the left by I_{y0}^T / I_{p0}, and assuming plasma resistance does not change with the plasma shape, yields:

$$\underbrace{\frac{I_{y0}^T}{I_{p0}} M_{yy} \frac{\partial I_y}{\partial I_p}}_{L_{pp}+X_{pp}} \dot{I}_p + \left(\underbrace{\frac{I_{y0}^T}{I_{p0}} M_{ye}}_{M_{pe}} + \underbrace{\frac{I_{y0}^T}{I_{p0}} M_{yy} \frac{\partial I_y}{\partial I_e}}_{X_{pe}} \right) \dot{I}_e + \underbrace{\frac{I_{y0}^T}{I_{p0}} M_{yy} \frac{\partial I_y}{\partial c_o}}_{X_{po}} \dot{c}_o + \underbrace{\frac{I_{y0}^T}{I_{p0}} R_{yy} \frac{I_{y0}}{I_{p0}}}_{R_{pp}} I_p = 0 \tag{20}$$

Generalized rigid plasma evolution model

- Hence:

$$(L_{pp} + X_{pp})\dot{I}_p + (M_{pe} + X_{pe})\dot{I}_e + X_{po}\delta\dot{c}_o + R_{pp}I_p = 0 \qquad (21)$$

with

- $L_{pp} = I_{y0}^T M_{yy} I_{y0} / I_{p0}^2$
- $X_{pp} = \frac{I_{y0}^T}{I_{p0}} M_{yy} \frac{\partial I_y}{\partial I_p} - L_{pp}$
- $M_{pe} = \frac{I_{y0}^T}{I_{p0}} M_{ye}$
- $X_{pe} = \frac{I_{y0}^T}{I_{p0}} M_{yy} \frac{\partial I_y}{\partial I_e}$
- $X_{po} = \frac{I_{y0}^T}{I_{p0}} M_{yy} \frac{\partial I_y}{\partial c_o}$
- $R_{pp} = \frac{I_{y0}^T}{I_{p0}} R_{yy} \frac{I_{y0}}{I_{p0}}$

Generalized rigid plasma evolution model

- We obtain the complete dynamic model or a rigid plasma

$$\begin{pmatrix} (\mathbf{M_{ee}} + \mathbf{X_{ee}}) & (\mathbf{M_{ep}} + \mathbf{X_{ep}}) \\ (\mathbf{M_{pe}} + \mathbf{X_{pe}}) & (L_{pp} + X_{pp}) \end{pmatrix} \begin{pmatrix} \dot{\mathbf{I}}_{\mathbf{e}} \\ \dot{I}_p \end{pmatrix} \tag{22}$$

$$+ \begin{pmatrix} \mathbf{R_{ee}} & 0 \\ 0 & R_{pp} \end{pmatrix} \begin{pmatrix} \mathbf{I_e} \\ I_p \end{pmatrix} = \begin{pmatrix} \mathbf{V_a} \\ 0 \end{pmatrix} \tag{23}$$

- This model has exactly the same structure as the RZIP model, just with more general expressions for X_{**} terms owing to the deformable plasma response matrix.
- Removing the X_{**} terms yields the model excluding the effects due to the plasma motion and deformation.
- We can combine this with a measurement equation as shown in part II.

Summary of plasma equilibrium evolution models

We have seen:

- Conductor-only models. No plasma
- Fixed-plasma models:

$$I_y = I_y(I_p) \tag{24}$$

- Rigid-plasma linearized model:

$$I_y = I_y(R_p, Z_p, I_p) \tag{25}$$

- Deformable-plasma linearized model:

$$I_y = I_{yo} + \frac{\partial I_y}{\partial I_e} I_e + \frac{\partial I_y}{\partial I_p} \delta I_p + \frac{\partial I_y}{\partial c_o} \delta c_o \tag{26}$$

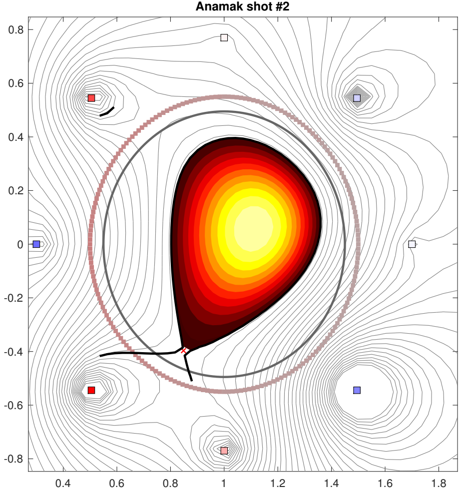
- Full evolution model-plasma model:

$$I_y = I_y(I_e, I_p, c_o) \tag{27}$$

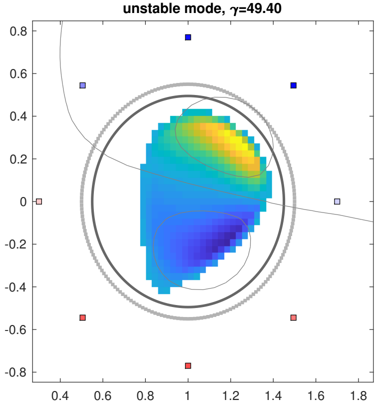
Examples of equilibria using MEQ

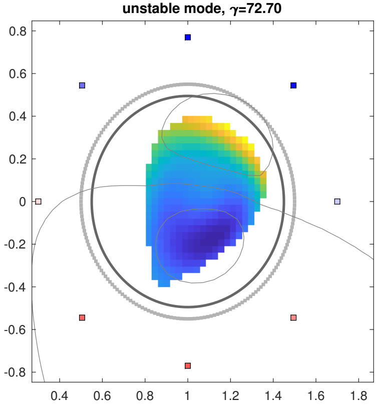
```
addpath ~/matlab/meq/ % adjust to suit your needs
addpath ~/matlab/meq/genlib % adjust to suit your needs

%% RZP model
[L,LX,LY] = rzp('ana',shot,time,'izgrid',true,'cde','OhmTor_rigid');
meas = {'zIp','rIp','Ip'}; % measurements from model
Ts = 0; % sample time: 0=continuous
sys = fgess(L,0,meas); % linearized model for rzp
fprintf('RZP unstable pole growth rate: %2.2f [1/s]\n',max(real(esort(pole(sys)))));

%% FGE: Free boundary Grad-Shafranov Evolution
[L2,LX2,LY2] = fge('ana',shot,time,'izgrid',true,'cde','OhmTor_rigid');
meas = {'zIp','rIp','Ip'}; % measurements from model
Ts = 0; % sample time: 0=continuous
sys = fgess(L,0,meas); % linearized model for fge
fprintf('FGE unstable pole growth rate: %2.2f [1/s]\n',max(real(esort(pole(sys)))));

%% Plot equilibrium
figure(1); set(gcf,'position',[0 0 600 500]); clf;
meqplotfancy(L,LY);
title(sprintf('Anamak shot #%d',shot))
set(gca,'box','on');
set(gcf,'paperpositionmode','auto');
print('-depsc','anamak_eq_2');

%% Plot eigenmode structures
figure(2); set(gcf,'position',[0 0 800 400]); clf;
subplot(121), fgeploteig(L)
subplot(122), fgeploteig(L2)
set(gcf,'paperpositionmode','auto'); print('-depsc','anamak_growth_rates');
```

Examples of equilibria using MEQ



Examples of equilibria using MEQ





Section 2

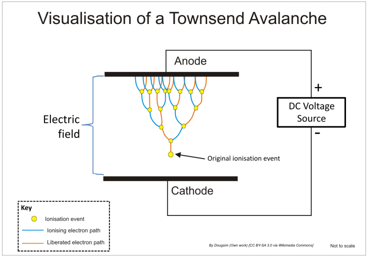
Plasma discharge evolution, from breakdown to plasma termination

Subsection 1

Plasma breakdown

Plasma breakdown conditions

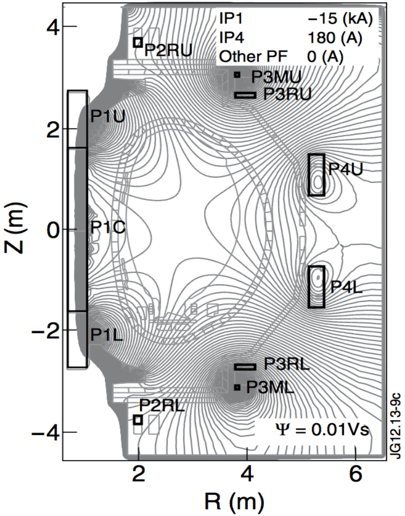
- Plasma breakdown occurs when the gas in the torus chamber ionizes.
  - A single electron is accelerated and collides with a neutral atom, ionizing it.
  - This liberates more electrons
  - These accelerate and collide with other atoms
  - This results in an ionization “avalanche” that quickly ionizes a large part of the gas.



Setting up the field at breakdown

- Plasma breakdown requires:
  - An electric field to accelerate the electrons.
  - A large *connection length*: distance for electrons to travel along the magnetic field to allow ionization of other atoms.
  - An appropriate pressure (not too many, not too few particles)
- The first two conditions are created by a combination of coils.
  - TF coils generate a toroidal magnetic field.
  - The OH (or CS) coils are ramped to induce a loop voltage (electric field).
  - PF coils are used to create a point with 0 poloidal field at the desired breakdown location and time. The resulting **B** field is locally almost exclusively toroidal - large connection length.

Example: JET breakdown field



JET poloidal flux at breakdown
from: Albanese et al, 2012 *Nucl. Fusion* **52** 123010

Breakdown design and optimal control problem

Desiderata for (Ohmic) breakdown:

- Field evolution:
  - Before breakdown: vertical field to avoid breakdown.
  - At breakdown: null field maximizing connection length.
  - After breakdown: ramping vertical field to maintain radial force balance + positive curvature for vertical stability.

$$\mathbf{B_p} = B_z(t)\mathbf{e_z} + \mathbf{nullfield}(t) \tag{28}$$

- Loop voltage evolution:
  - Sufficient loop voltage at $t = 0$ and later to breakdown, burn-through, and ramp I_p.
  - Low loop voltage otherwise to avoid consuming Ohmic coil flux.
- Coil evolution:
  - Pre-charge OH coils to have maximum flux swing.

Breakdown design and optimal control problem

- We have linear relations between circuit/passive current evolution and vacuum fields/loop voltage (excluding plasma):

$$B_{r,x}(t) = B_{r,xe} I_e(t) \tag{29}$$

$$B_{z,x}(t) = B_{z,xe} I_e(t) \tag{30}$$

$$V_x(t) = M_{xe} \dot{I}_e(t) \tag{31}$$

- Also a linear model for the conductor (passive + active) current evolution:

$$M_{ee} \dot{I}_e(t) + R_{ee} I_e(t) = V_e \quad (36) \tag{32}$$

Breakdown design and optimal control problem

- Starting from the circuit equation:

$$M_{ee}\,\dot{I}_e(t) + R_{ee}\,I_e(t) = \mathbb{I}_{ea}V_a(t). \tag{33}$$

- Define discrete time points $t_k = k\,\Delta t$ and approximate $I_{e,k} \approx I_e(t_k), \quad V_{a,k} \approx V_a(t_k)$.
- Using the backward Euler approximation, $\dot{I}_e(t_k) \approx \frac{I_{e,k}-I_{e,k-1}}{\Delta t}$. Substituting into the circuit equation and evaluating I_e at time t_k:

$$M_{ee}\frac{I_{e,k} - I_{e,k-1}}{\Delta t} + R_{ee}I_{e,k} = \mathbb{I}_{ea}V_{a,k}. \tag{34}$$

Multiply both sides by Δt and rearranging grouping terms in $I_{e,k}$:

$$(M_{ee} + R_{ee}\,\Delta t)\,I_{e,k} = M_{ee}I_{e,k-1} + \Delta t\mathbb{I}_{ea}V_{a,k}. \tag{35}$$

Finally, solving for $I_{e,k}$:

$$I_{e,k} = (M_{ee} + R_{ee}\,\Delta t)^{-1}\left(M_{ee}\,I_{e,k-1} + \Delta t\mathbb{I}_{ea}V_{a,k}\right). \tag{36}$$

Breakdown design and optimal control problem

- Define

$$A = (M_{ee} + R_{ee}\,\Delta t)^{-1}\,M_{ee}, \quad B = (M_{ee} + R_{ee}\,\Delta t)^{-1}\,\Delta t\,S_a. \tag{37}$$

Hence we can write

$$I_{e,k} = A\,I_{e,k-1} + B\,V_{a,k}. \tag{38}$$

By iterating from $k = 1$ to N and stacking all $\{I_{e,1}, \ldots, I_{e,N}\}$ and $\{V_{a,1}, \ldots, V_{a,N}\}$, we obtain the lifted representation:

$$\begin{pmatrix} I_{e,1} \\ I_{e,2} \\ \vdots \\ I_{e,N} \end{pmatrix} = \underbrace{\begin{pmatrix} A \\ A^2 \\ \vdots \\ A^N \end{pmatrix}}_{S} I_{e,0} + \underbrace{\begin{pmatrix} B & 0 & \cdots & 0 \\ A\,B & B & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ A^{N-1}B & A^{N-2}B & \cdots & B \end{pmatrix}}_{T} \begin{pmatrix} V_{a,1} \\ V_{a,2} \\ \vdots \\ V_{a,N} \end{pmatrix} \tag{39}$$

Breakdown design and optimal control problem

- Using this representation we time-history of currents, but also of fields and fluxes (at spatial points of interest) to the time-history of circuit voltages:

$$\begin{pmatrix} B_{r,k=1} \\ B_{r,k=2} \\ \vdots \\ B_{r,k=N} \end{pmatrix} = T_{r,a} \begin{pmatrix} V_{a,k=1} \\ V_{a,k=2} \\ \vdots \\ V_{a,k=N} \end{pmatrix}, \quad \begin{pmatrix} B_{z,k=1} \\ B_{z,k=2} \\ \vdots \\ B_{z,k=N} \end{pmatrix} = T_{z,a} \begin{pmatrix} V_{a,k=1} \\ V_{a,k=2} \\ \vdots \\ V_{a,k=N} \end{pmatrix} \tag{40}$$

- Similarly for the circuit currents:

$$\begin{pmatrix} I_{a,k=1} \\ I_{a,k=2} \\ \vdots \\ I_{a,k=N} \end{pmatrix} = T_{Ia} \begin{pmatrix} V_{a,k=1} \\ V_{a,k=2} \\ \vdots \\ V_{a,k=N} \end{pmatrix} \tag{41}$$

Breakdown design and optimal control problem

- This ultimately allows us to define a constrained least-squares problem for the (time-history of) the fields and voltages (see also [2, 3]):

$$\min_x J(x) \quad \text{subject to} \quad Cx \leq d \tag{42}$$

where

$$J(x) = \nu_r \left\| B_{r,\text{target}} - T_{r,a} x \right\|^2 + \nu_z \left\| B_{z,\text{target}} - T_{z,a} x \right\|^2 + \nu_V \left\| V_{\text{target}} - T_{V,a} x \right\|^2 + \nu_x \left\| x \right\|^2, \tag{43}$$

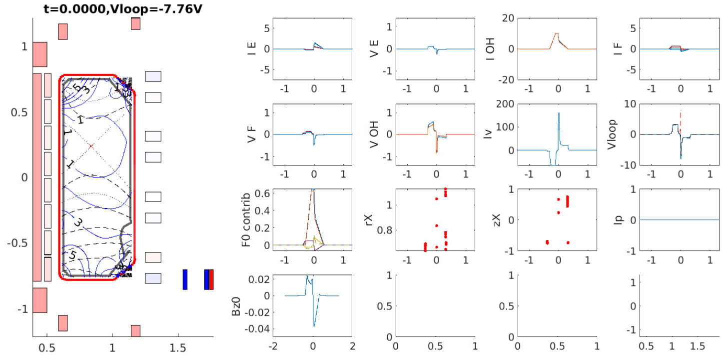
and

$$C = \begin{pmatrix} T_{I_a} \\ -T_{I_a} \\ \mathbb{I}_a \\ -\mathbb{I}_a \end{pmatrix}, \quad d = \begin{pmatrix} I_{a,\max} \\ -I_{a,\min} \\ V_{a,\max} \\ -V_{a,\min} \end{pmatrix}. \tag{44}$$

- The cost function terms correspond to:
  - Radial,Vertical field evolution target
  - Loop voltage evolution target
  - Regularization term minimizing coil currents
- Constraints: Circuit current constraints and Power supply voltage

Breakdown design and optimal control problem

- Quadratic constrained optimization problem:
  - Convex problem.
  - Fast solvers exist (e.g. Matlab: `quadprog`, Open-source `OSQP`).
- Example of optimized TCV breakdown.



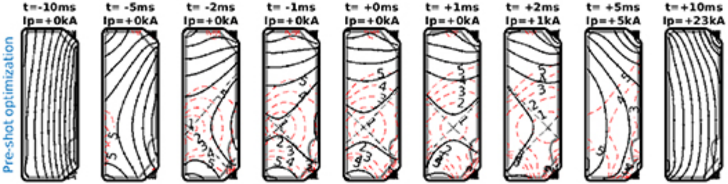
Automated shot-to-shot startup optimisation on TCV

- Write breakdown *+startup* evolution model: add a circuit equation for plasma assuming fixed location.
- Design target fields to have radially + vertically stable plasma in early ramp-up phase.
- Perform *nominal* design of optimal breakdown based on machine model. Yields nominal $V_a(t), I_a(t)$, and field/flux evolutions.
- Run experiment, harvest error $e(t)$
- Write relation between error and input trajectory using linear relations (40)
- Adjust trajectories in the direction to *reduce* this error.
- Related to control technique called *Iterative Learning Control*

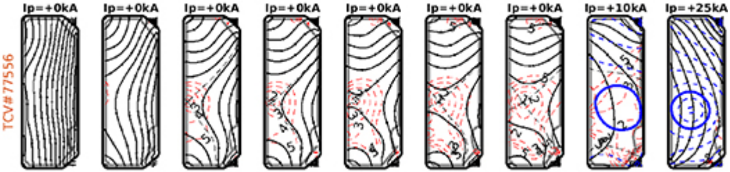
Automated shot-to-shot optimisation of startup phase

From [3]

- Nominal breakdown:



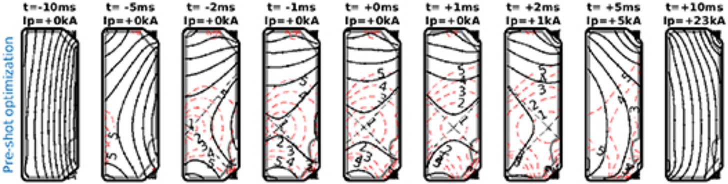
- First experimental breakdown:



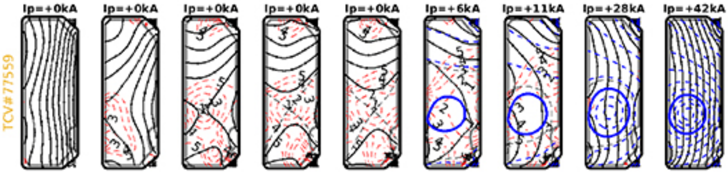
Automated shot-to-shot optimisation of startup phase

From [3]

- Nominal breakdown:



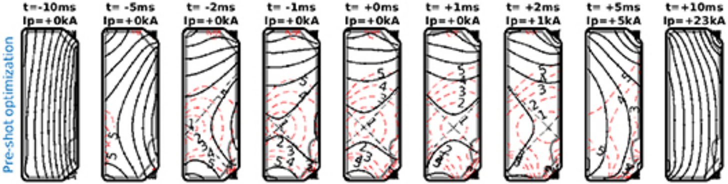
- 2rd attempt in the sequence:



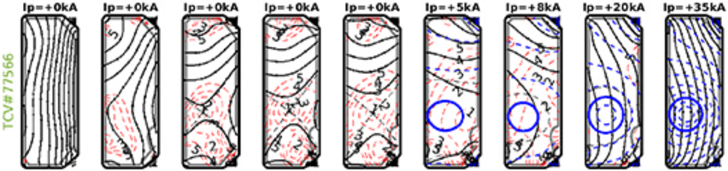
Automated shot-to-shot optimisation of startup phase

From [3]

- Nominal breakdown:



- Final (4th) attempt:



Subsection 2

Ramp-up phase

Example: EAST ramp-up phase

- Start with low-current, limited plasma sitting against the wall
- Ramp up current, increase shape and create x-points
- Fully developed shape at start of flat-top.

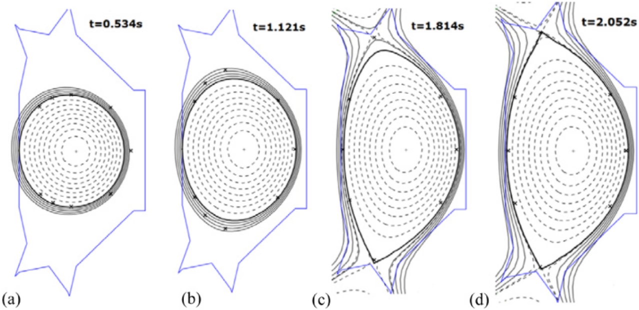


Figure: EAST ramp-up shape evolution, from [4]

Control issues for plasma ramp-up

- Switching from feedfoward-controlled breakdown to feedback control of plasma position and current
- Switching from R, Z control only to full shape control
- Well-timed formation of x-point.
- Obtain desired q profile, β, ℓ_i at the start of the flat-top
- Remain within engineering and physics constraints all the time.

Subsection 3

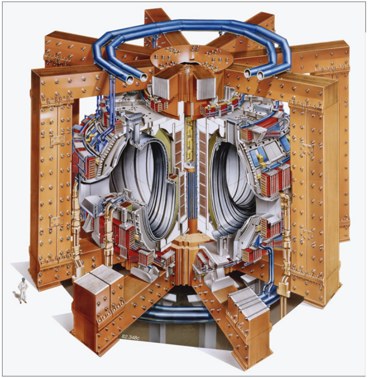
Flat-top

Magnetic control issues for plasma flat-top

- Maintain required position, shape and I_p.
- Compensate from disturbances due to change in β, ℓ_i, I_p.
- Compensate for changing stray field due to central solenoid ramp.

Air core vs. iron core tokamaks

- Iron core: iron *transformer yoke* around tokamak, 'guides' field generated by Central Solenoid.
- Air core: no iron, OH gives 'stray' vertical field. OH coils designed to minimize this field.



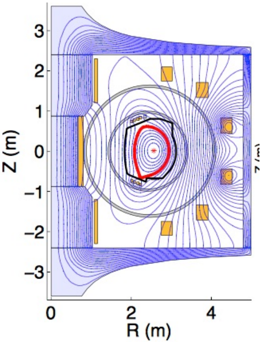


Figure: Left: JET with iron transformer yokes. Right: Flux map for WEST from CEDRES code

Air core vs. Iron core tokamaks

	Air core	**Iron core**
Adv:	Circuit equations are LTI, easier reconstruction of fields	Smaller stray field
Disadv:	Need to compensate OH stray field during shot	Field depends on iron magnetization: nonlinear and time-dependent equations.
Examples:	ITER, TCV, AUG, DIII-D	JET, Tore Supra

Subsection 4

Ramp-down

Ramp-down: open research questions

- Ramp down I_p, decrease shape, etc in a controlled way.
- Current profile evolution, timing of H-L transition during ramp-down play a key role [5]
- Much more to say, but outside the scope of this class.
- Open research field, but very important for ITER and DEMO: safe plasma termination.

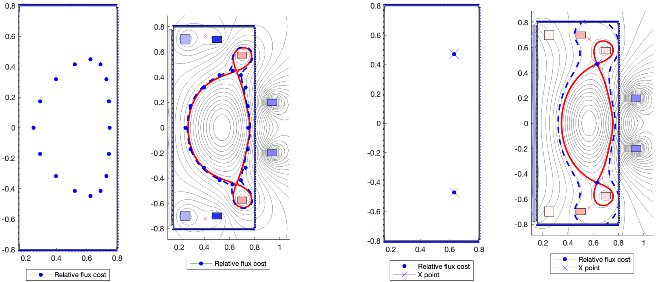
Section 3

Equilibrium calculation workflows for tokamak operation

Static inverse problem

Desired equilibrium snapshot → FBT → I_a +Nominal plasma equilibrium

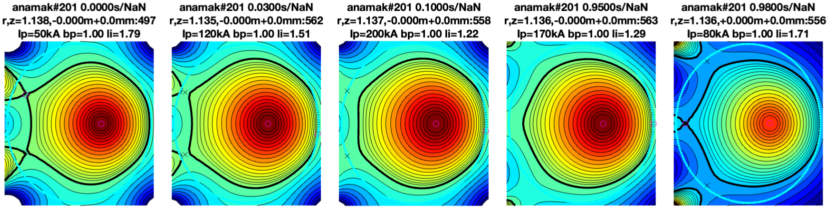
- Picard Iterations used in FBT [6]:
  - For fixed I_y, solve constrained QP problem to get I_a.
  - For fixed I_a and given plasma constraints, solve GS equation for I_y.



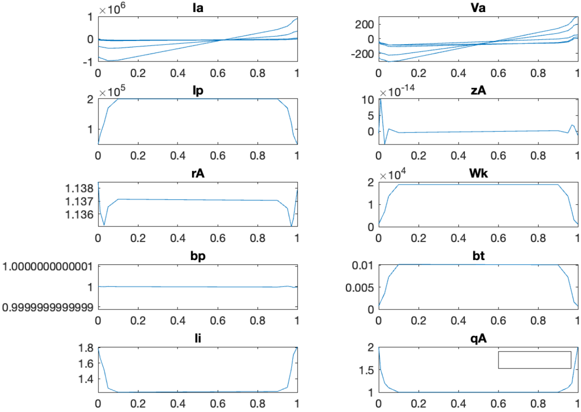
Dynamic inverse problem (Pulse Planning)

Desired equilibrium evolution → FBT-evo/GSPD → $I_a(t)$, $V_a(t)$ +Nominal plasma equilibrium *evolution*

- Picard Iterations used in GSPulse [7]:
  - For fixed $I_y(t)$, solve constrained QP problem to get $I_e(t)$, $V_a(t)$ using *lifted* circuit equations like (40)
  - For fixed $I_e(t)$ and given plasma internal constraints, solve GS equation to get $I_y(t)$.



Dynamic inverse problem (Pulse Planning)

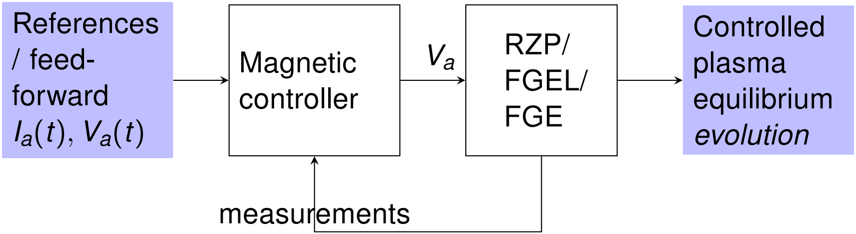


Equilibrium reconstruction

Magnetic measurements (t) → LIUQE / LIH → Reconstructed plasma equilibrium evolution (t)

- Choice between:
    - Full GS model (LIUQE) - accurate but nonlinear
    - Simplified Finite-Element model (See Part II) (LIH) - inaccurate but linear and fast
- Post-shot or real-time (LIUQE $<$ 1ms on TCV).

Controller design and testing



- Solve time-dependent plasma + circuit evolution equations to test feedback controllers. Choice between:
  - RZP/FGEL: linear rigid / deformable plasma model
  - FGE: nonlinear plasma model
- Simulate magnetic controller in the loop
  - Receive references and feedforward voltages from pulse planning
  - Step the controller together with the plant, including any delays etc.
  - Might include RT-equilibrium reconstruction!
- Basically acting as a 'tokamak simulator' for the magnetics, same inputs/outputs as real tokamak.

Magnetic control - summary

- Dynamics of PF coils + vessel, controllers for PF coils
- Model for plasma as rigid conductor
    - Plasma current control
    - Plasma radial force balance and vertical field
    - Plasma vertical stability and control
    - Plasma shape control
- Magnetic measurements
- Plasma MHD equilibrium and Grad-Shafranov equation
- Equilibrium preparation, simulation and reconstruction workflows
- Plasma discharge phases

Bibliography I

- Carpanese, F. 2021 *Development of free-boundary equilibrium and transport solvers for simulation and real-time interpretation of tokamak experiments* Ph.D. thesis
- di Grazia, L. *et al.* 2022 *Fusion Engineering and Design* **176** 113027
- di Grazia, L.E. *et al.* 2024 *Nuclear Fusion* **64** 096032
- Yuan, Q.P. *et al.* 2013 *Nuclear Fusion* **53** 43009
- Van Mulders, S. *et al.* 2024 *Plasma Physics and Controlled Fusion* **66** 025007
- Hofmann, F. *et al.* 1998 *Nuclear Fusion* **38** 399
- Wai, J.T. *et al.* 2023