ASDEX Upgrade

IPP

ASTRA-7

a state-of-the-art IPP transport code

Dr. E. Fable

Dedicated to the memory of Prof. G. V. Pereverzev

AUG seminar, 27 January 2012

Outline

- A brief **ASTRA** summary
  (and what distinguishes it from other transport codes)

- **ASTRA-7**: - what has been changed/added
  - benchmarks and results

- Future directions

ASTRA: a sophisticated transport analysis tool

* **ASTRA** [G. V. Pereverzev, Y. P. Yushmanov] is a 1D radial transport code that can solve for several profiles evolution:

- **Current profile**
- **Temperatures** (electrons, ions)
- **Densities** (electrons)

* Modular and very user friendly, GUI allows interactive simulations

* Transport equations: - **Finite volumes** in the spatial domain
- **Linearly implicit** scheme in the time domain

* **Advection-diffusion** terms are treated by either **central-differences** or with more sophisticated **power-law** schemes. Treatment of **stiff transport** has also been implemented [G. V. Pereverzev, M. Corrigan, CPC 2008]

ASTRA: a sophisticated transport analysis tool

* Recently implemented features:

  - Possibility of **parallelizing** several processes (IPC);

  - **TORBEAM** [E. Poli] has been coupled;

  - The equilibrium code **SPIDER** [A. A. Ivanov, S. Medvedev] has been coupled to perform **prescribed** and **free-boundary** computations of discharge evolution (the latter is still in progress though);

* **Latest released** version: **ASTRA-6.2.1**
  (available on the svn server at request).

Main points of discussion

* Nature of numerical grid: minor-radius-equivalent or normalized grid?

* A toroidal momentum transport equation is needed to extend the physics coverage of the code;

* Coupling with equilibrium has to be made robust, stable, and consistent;

- *Advances on different points have benefited from many discussions with colleagues.*

ASTRA-6.2.1 grid

* Radially-dependent variables are functions of the independent variable ρ, defined as:

$$\rho = \sqrt{\frac{\Phi}{\pi B_0}}$$

- Φ : toroidal magnetic flux;
- B_0 : reference vacuum magnetic field;

* ρ-step is defined and constant during the simulation → number of grid points in the plasma depends on the boundary value ρ_b ;

* Boundary conditions can be problematic. Also, what to do with portions of plasma that are either scraped off or created from the SOL ?

* Last 2 grid points have a different differential than the inner ones, renders derivatives and extrapolation cumbersome;

* Coupling with equilibrium becomes problematic, in particular when nested iteration cycles are present.

ASTRA-7 uses normalized $\rho \rightarrow x$

* Radially-dependent variables are now functions of the independent variable x, defined as:

$$x = \sqrt{\frac{\Phi}{\Phi_b}}$$

- Φ : toroidal magnetic flux;
- Φ_b : boundary toroidal magnetic flux;

* x-step is defined and constant during the simulation → number of grid points in the plasma is invariant. Total number of points varies;

* Boundary conditions are given at $x = 1$, which is a grid point. No need to treat plasma outside the LCFS;

* Every grid point has the same differential;

* Coupling with equilibrium becomes straightforward, as will be shown later on.

Grid compression/expansion

* Let us take as an example the continuity equation:

$$\frac{1}{V'}\frac{\partial(V'n)}{\partial t_\rho}+\frac{1}{V'}\frac{\partial}{\partial\rho}\left(V'\Gamma_{n,\rho}\right)=0$$

* And assume that the particle flux with respect to constant-ρ surfaces is zero:

$$\frac{\partial(V'n)}{\partial t_\rho}=0$$

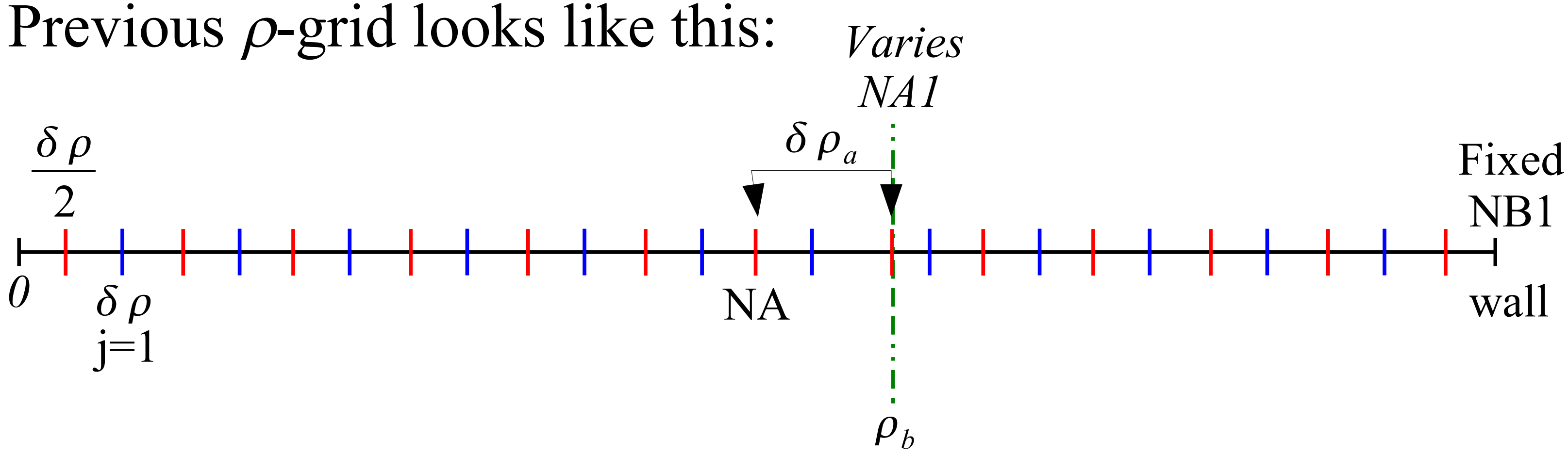
* We now go to the normalized grid x :

$$\frac{\partial(V'n)}{\partial t_x}-x\frac{\partial(V'n)}{\partial x}\frac{\dot{\Phi}_b}{\Phi_b}=0$$

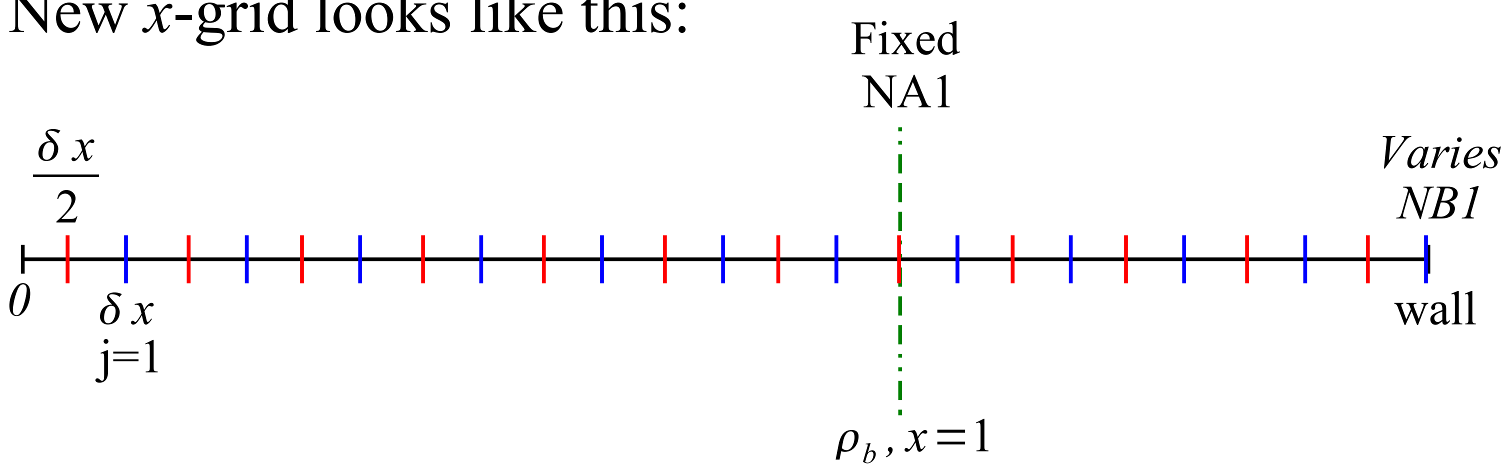
* This new term has been added to the equations to preserve the conservation properties on the normalized grid. Can be treated a **posteriori** (used now) or **iteratively**.

Grids comparison

* Previous ρ-grid looks like this:



* New x-grid looks like this:



Toroidal momentum transport equation

* The toroidal rotation is now solved for by using the quasi-neutrality condition:

$$\langle \nabla \cdot \boldsymbol{J} \rangle = 0 \rightarrow \langle J_r R B_p \rangle = 0$$

* The radial current can be obtained from the toroidal projection of the general fluid momentum equation:

$$\frac{\partial (V' \langle M_\phi \rangle)}{\partial t} + \frac{\partial}{\partial \Phi} \langle R |\nabla V| \Pi_{\phi r} \rangle - \langle J_r R B_p \rangle = \langle R \sum_s S_{\phi,s} \rangle$$

$= 0$

$$\begin{cases} M_\phi = R \sum_s m_s n_s U_{\phi,s} \quad ; \; U_{\phi,s} = R\,\Omega_s \\ \Pi_{\phi r} = \sum_s \left(\Pi_{\phi r,s} + m_s U_{\phi,s} \Gamma_s \right) \end{cases}$$

* So:

$$\frac{\partial (V' \langle M_\phi \rangle)}{\partial t} + \frac{\partial}{\partial \Phi} \langle R |\nabla V| \Pi_{\phi r} \rangle = \langle R \sum_s S_{\phi,s} \rangle$$

Parallel velocity is solved for ultimately

* Assume all ions have same toroidal velocity, no poloidal velocity:

$$\langle R^2 \Omega \rangle = \frac{\langle R^2 \rangle B_0}{I} u_{||} \quad ; \quad u_{||} = \frac{\langle U_{||} B \rangle}{B_0}$$

* We are lead to an equation for $u_{||}$:

$$\frac{1}{V'} \frac{\partial (V' Y_0 u_{||})}{\partial t} + \frac{1}{V'} \frac{\partial}{\partial \rho} \left[V' \left(P_1 \frac{\partial u_{||}}{\partial \rho} + P_2 u_{||} + P_3 \right) \right] = \left\langle R \sum_s S_{\phi,s} \right\rangle$$

$$\begin{cases} Y_0 = \left(\sum_s m_s n_s \right) \dfrac{\langle R^2 \rangle B_0}{I} \quad ; \quad P_1 = -\left(\sum_s m_s n_s \right) I B_0 \left\langle \dfrac{|\nabla \rho|^2}{B^2} \right\rangle \chi_{||} \\ P_2 = \left(\sum_s m_s n_s \right) I B_0 \left\langle \dfrac{|\nabla \rho|}{B^2} \right\rangle V_{||} \quad ; \quad P_3 = \left(\sum_s m_s n_s \right) I \left\langle \dfrac{|\nabla \rho|}{B} \right\rangle \dfrac{\Pi_{||r}}{m n} \end{cases}$$

Reasons for choosing $u_{||}$ over Ω

* Many present codes (JETTO, CHRONOS, TRANSP, …) solve for the toroidal angular velocity Ω;

* Historically this is due (I think) to the fact that experimental measurements are mostly available for that quantity at specific major radius location (LFS for example);

* However, I chose to solve for $u_{||}$ for the following reasons:
- "semantic" consistency with definitions of parallel current density;
- the parallel direction is the natural one for the plasma to flow;
- parallel momentum is dominated by neoclassical effects → differential rotation from neoclassical theory is given in terms of $u_{||}$;
- turbulent mechanisms are best given in terms of || and ⊥ directions;
- conversion from measured $\Omega \rightarrow u_{||}$ is done in ASTRA anyway.

Coupling with 2D equilibrium on the *x*-grid

* Let us assume that we are solving the coupled current diffusion equation (CDE):

$$\left[\frac{\partial\psi}{\partial t}\right]_x - x\frac{\dot{\Phi}_b}{\Phi_b}\frac{\partial\psi}{\partial x} = C_0\frac{\partial}{\partial\rho}\left(\frac{g_3 g_2}{\rho}\frac{\partial\psi}{\partial\rho}\right)$$

$$\begin{cases} C_0 = \dfrac{I^2}{8\pi^3 B_0^2\mu_0\sigma\rho} \\ g_2 = \left\langle\dfrac{|\nabla V|^2}{R^2}\right\rangle ;\ g_3 = \left\langle\dfrac{1}{R^2}\right\rangle \end{cases}$$

* And the Grad-Shafranov equation:

$$\Delta^*\Psi = -4\pi^2 R^2\frac{\partial P}{\partial\Psi} - 4\pi^2 F\frac{\partial F}{\partial\Psi} + (\ldots)_{coils}$$

* … dynamically (in a time evolution, with evolving boundary, or else)

A novel form of the Grad-Shafranov equation

* We thus know $\psi(x)$ and $P(x)$, as such we also know $P(\psi)$;

* The flux ψ is already known everywhere in the plasma, but we have to find the plasma position, and also the boundary toroidal magnetic flux;

* To this purpose we use the relation between F and ψ :

$$F = 2x\Phi_b \frac{2\pi}{g_3}\frac{\partial\psi}{\partial V}\hat{q} \quad ; \quad \hat{q} = \frac{\partial x}{\partial \psi}$$

* To rewrite the Grad-Shafranov equation as:

$$\Delta^*\Psi = -4\pi^2 R^2 \frac{\partial P}{\partial \Psi} - 64\pi^4 \Phi_b^2 \frac{\hat{q}}{g_3}\frac{\partial\psi}{\partial V}\frac{\partial}{\partial\Psi}\left(\frac{\hat{q}}{g_3}\frac{\partial\psi}{\partial V}\right) + (\ldots)_{coils}$$

* The solution of this equation must satisfy the following constraints:

1) $\Psi = \psi$; 2) $R_0 B_0 = 2\Phi_b \frac{2\pi}{g_{3,b}}\left[\frac{\partial\psi}{\partial V}\right]_b \hat{q}_b$; 3) *Consistency of metric quantities*, *Consistency of* Φ_b

Numerical scheme

* We flux-surface-average the Grad-Shafranov equation:

$$\frac{\partial}{\partial V}\left(g_2\frac{\partial\psi}{\partial V}\right)=-4\pi^2\frac{\partial P}{\partial\psi}-64\pi^4\Phi_b^2\hat{q}\frac{\partial\psi}{\partial V}\frac{\partial}{\partial\psi}\left(\frac{\hat{q}}{g_3}\frac{\partial\psi}{\partial V}\right)$$

* And we manipulate to get the following form:

$$H\frac{\partial}{\partial x}\left(g_2 H\right)=-4\pi^2\frac{\partial P}{\partial x}-64\pi^4\Phi_b^2\hat{q}\,H\frac{\partial}{\partial x}\left(\frac{\hat{q}}{g_3}H\right)\quad ;\quad H=\frac{\partial\psi}{\partial V}$$

* We solve this 1st ODE iteratively with the boundary condition:

$$H_b=\frac{g_{3,b}}{\hat{q}_b}\frac{R_0 B_0}{4\pi\Phi_b}$$

* And the additional constraint that: $VOLUME=\int_{\psi_0}^{\psi_b}\frac{d\psi}{H(\Phi_b)}=\int_0^1\frac{1}{\hat{q}}\frac{dx}{H}$

* Where the plasma *VOLUME* is an input here, so it is known.

Numerical scheme

* Having found H, we now plug it into the Grad-Shafranov equation in this way:

$$\Delta^* \Psi = -4\pi^2 \left(R^2 - \frac{1}{g_3} \right) \frac{\partial P}{\partial \Psi} - 4\pi^2 \frac{1}{g_3} \frac{\partial}{\partial V} \left(g_2 \frac{\partial \psi}{\partial V} \right) + (\ldots)_{coils}$$

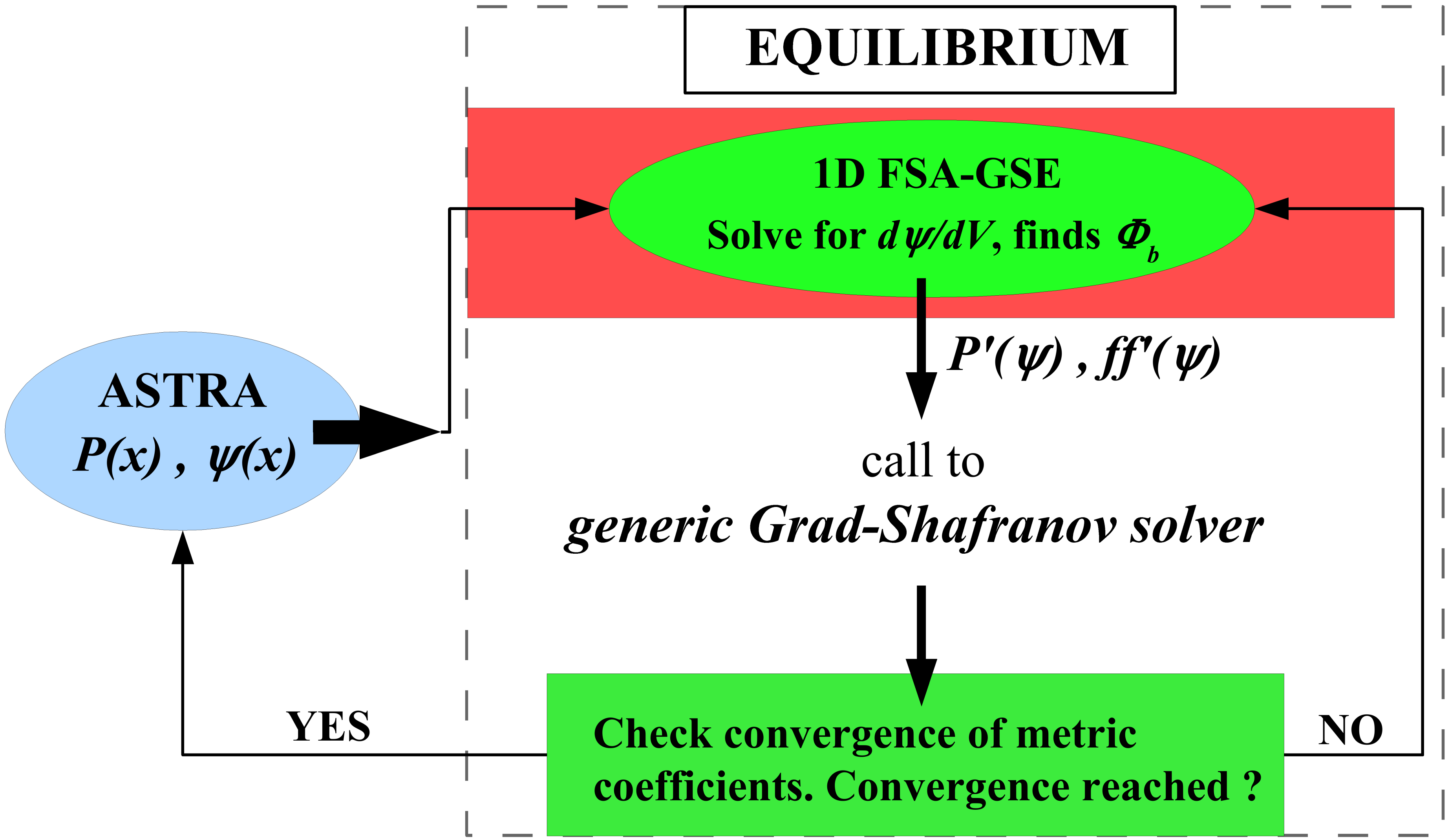
* This equation is solved on an adaptive grid that ends up having the exact same ψ points as the input from ASTRA → what this equation actually does is just find the shape of the flux surfaces but it does not change their values! (This is what we want in the end: $\Psi = \psi$);

- notice that the VOLUME can change as well in the free-boundary case;

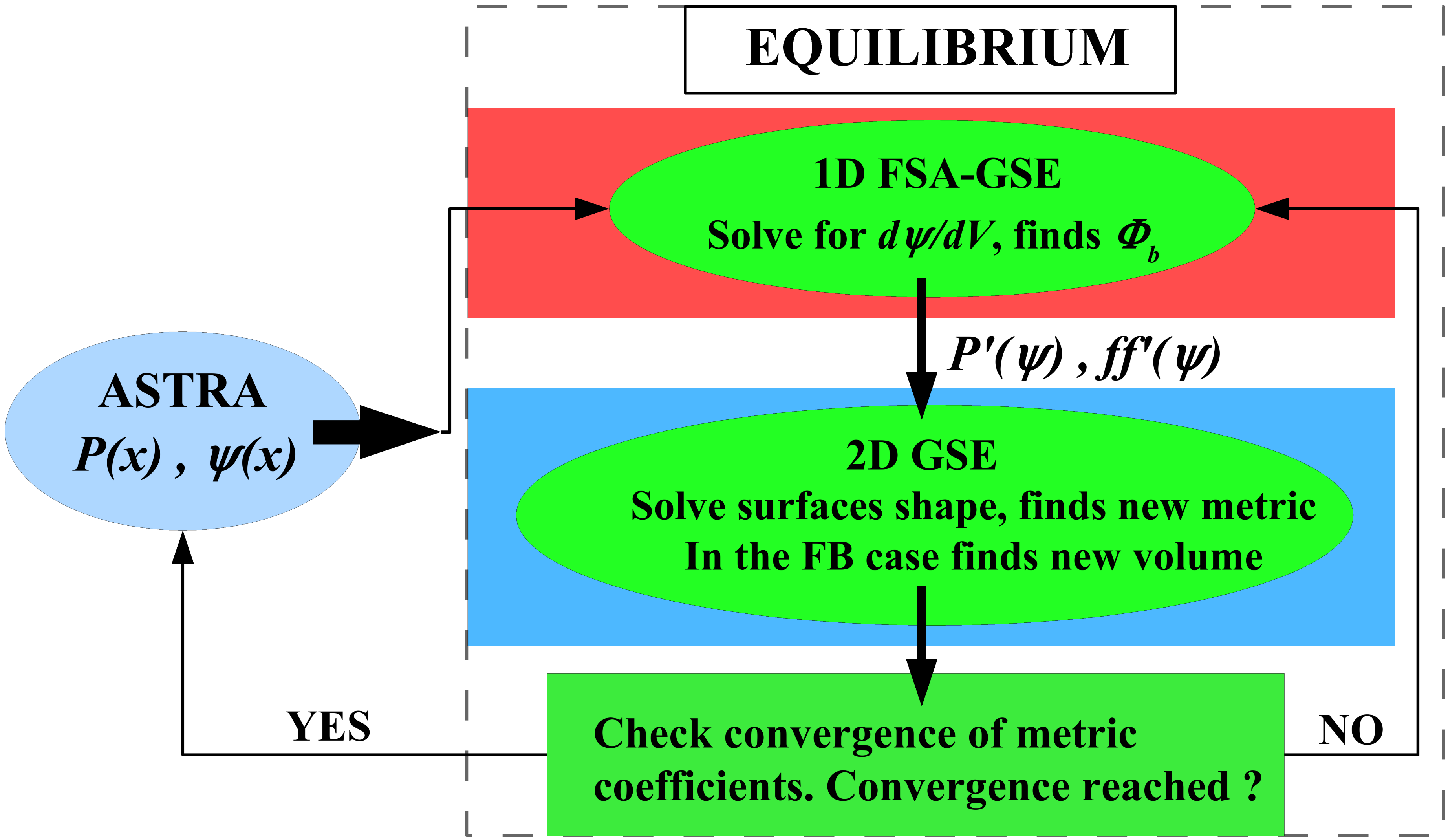
* After this equation is solved, we found the new metric quantities and go back to the evaluation of H, and so on until convergence;

* When convergence is reached, go back to ASTRA.

To summarize with a graph



To summarize with a graph



To test this, a new GS solver has been written

* Solves GS inside a fix boundary using curvilinear, non-orthogonal grid $(\hat{\Psi}, \theta)$;

* Values of $\hat{\Psi}$ are given as an input and are fixed;

* *(R,Z)* grid is built at each iteration such that grid points fall on the flux surfaces of constant $\hat{\Psi}$ and the magnetic axis satisfies $|\nabla \Psi| = 0$;

* Δ^* is solved for using an optimized *SOR* iterative scheme (or with *LAPACK* routines for banded matrices);

* Differential operators: finite volume method and tensor calculus → conservation properties and accuracy of the result are assured.

* Called **EQUIL2D** in the following. There is also **EQUIL1D**, which assumes a *circular*, *large aspect ratio* tokamak.

Benchmarks with Astra-6.2.1

* Several benchmarks are carried out to check correctness of the new features implemented in ASTRA-7, in particular:

- Benchmark n_e, T_e, T_i, ψ evolution separately and together using EMEQ (3 moments) equilibrium;

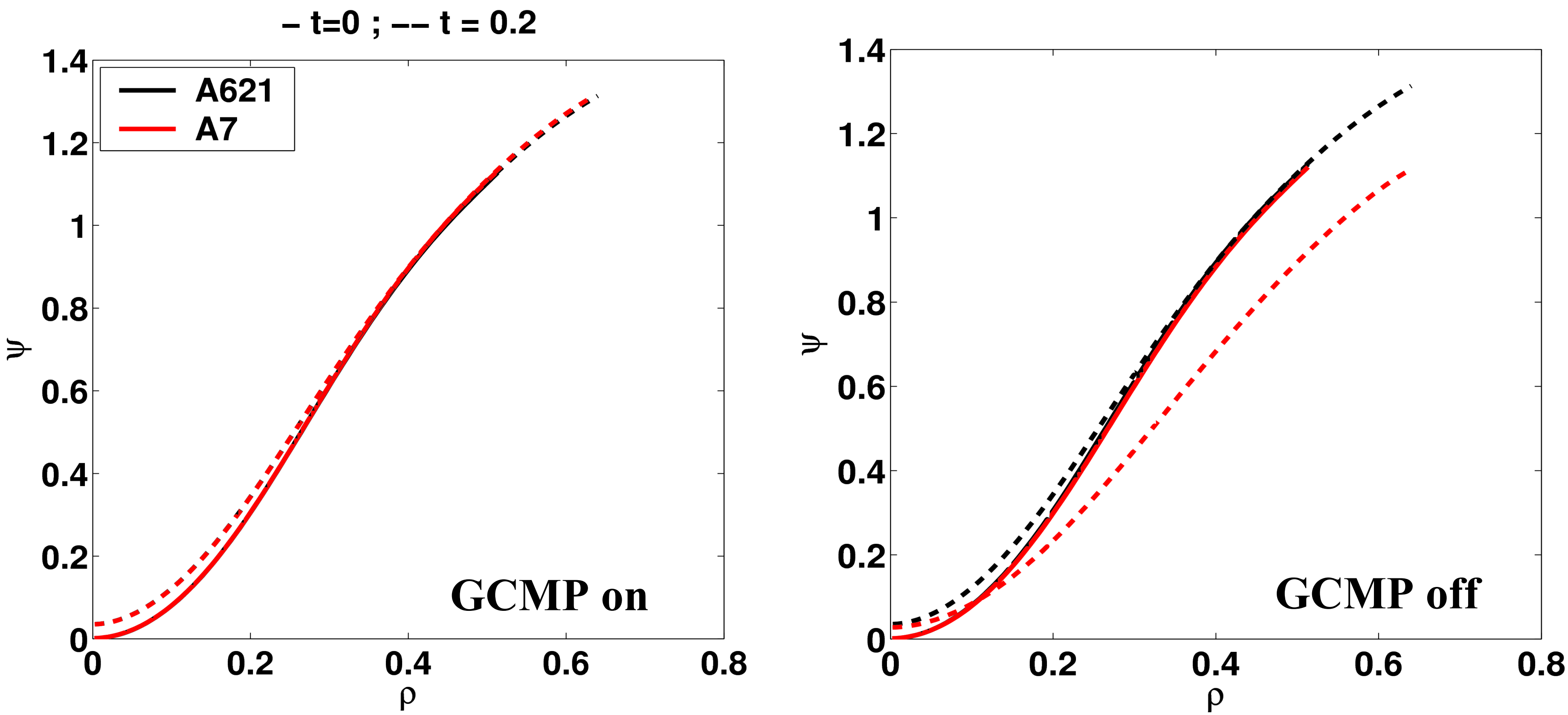
- Benchmark $u_{\|}$ evolution against analytical predictions for simple cases;

- Benchmark stiff transport cases for T_e and T_i;

- Benchmark between equilibrium codes with evolving ψ: EMEQ, SPIDER, EQUIL2D, SPID2 (i.e. SPIDER included in the proposed iterative scheme);

- First example of free-boundary computation of an AUG case.

Effect of grid expansion on ψ evolution



- This case has $k = 1$ @ $t = 0$ and $k = 1.6$ @ $t = 0.2\ s$. Only ψ evolves, other kinetic profiles are flat. Initial equilibrium is circular. EMEQ solver is used;

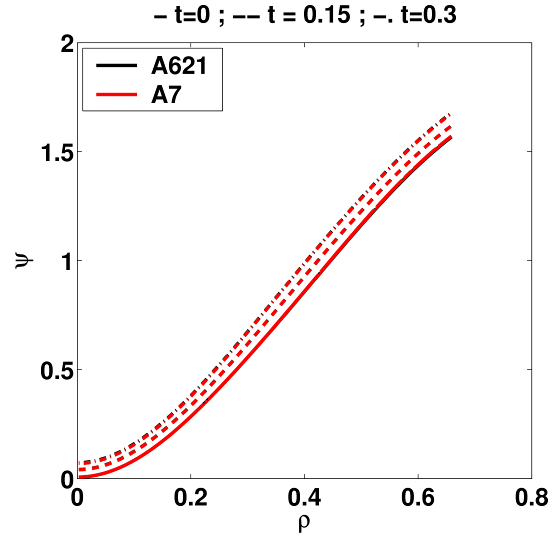
- Agreement between new and old version is excellent when grid compression (**GCMP**) is included in the equations solved on normalized grid.

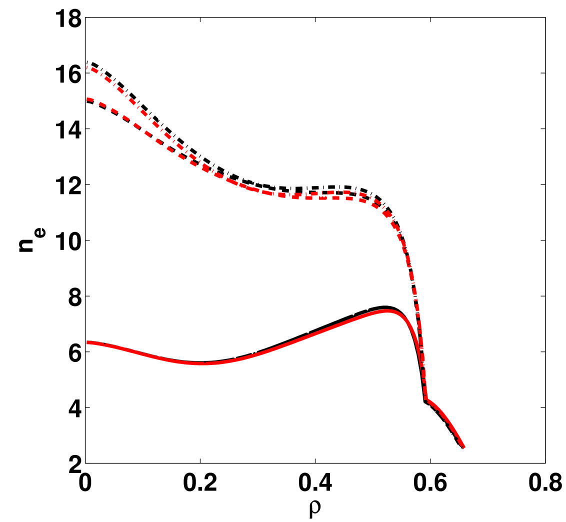
Benchmark over full profiles evolution

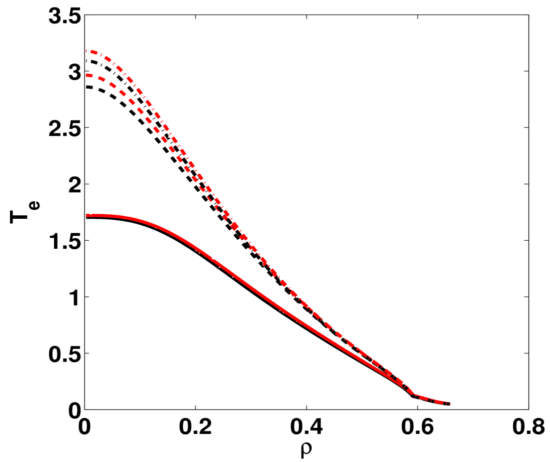
- Here AUG #26328 @ t = 1.1 s (L-mode, diverted) is used. EMEQ provides equilibrium;

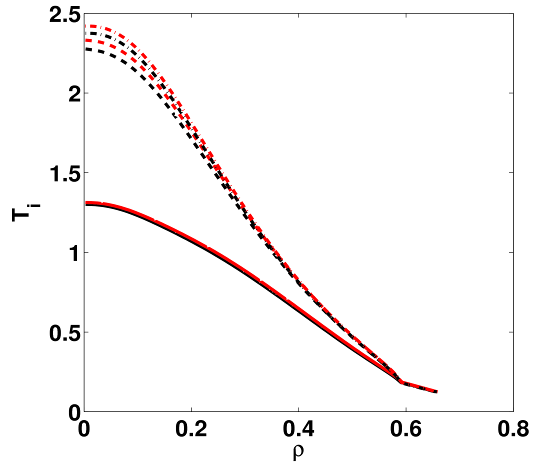
- T_e, T_i, n_e, ψ are let evolve. Transport coefficients are given by:

$$\begin{cases} \chi_i = 0.1 + 4x^2 + \chi_{neo} \\ \chi_e = 0.1 + 4x^2 \\ D = 0.3 + 4x^2 \\ V = -W_p - D\left(0.5\dfrac{R}{L_{Te}} - 5\mathrm{s}\right) \end{cases}$$









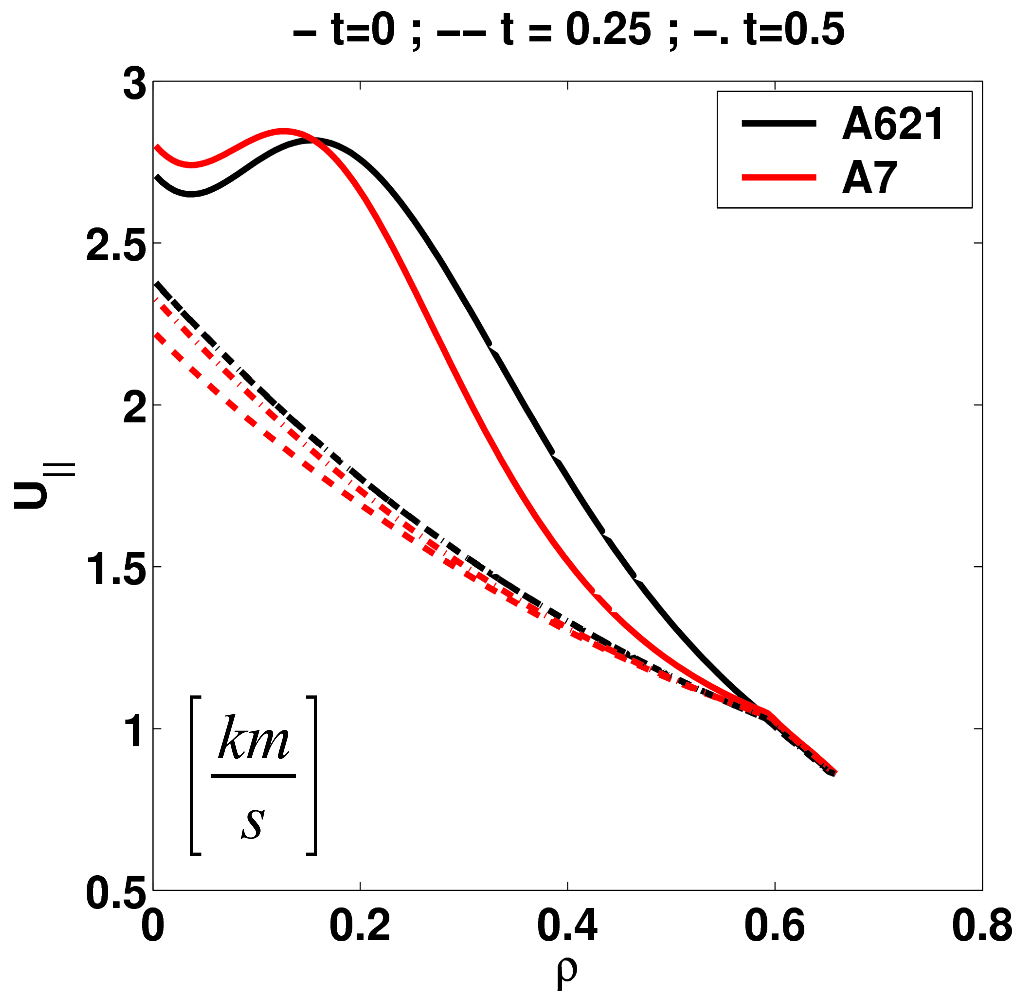
- Almost perfect agreement (not exactly the same curves because of slightly different differentiation of metric quantities is used);

- This discrepancy is < *3%* .

Testing parallel momentum transport equation

- Same case as before. In A621 the auxiliary equation for *F1* is run, mocking rotation and assuming constant density (which is not). So one expects agreement only at steady-state!

$$
\begin{cases}
\chi_i = 0.1 + 4x^2 + \chi_{neo} \\
\chi_e = 0.1 + 4x^2 \\
D = 0.3 + 4x^2 \\
V = -W_p - D\left(0.5\frac{R}{L_{Te}} - 5s\right) \\
\chi_{||} = 0.3 + 4x^2 \\
V_{||} = -1.5\chi_{||} \\
R_{||} = R_{\perp} = T = 0
\end{cases}
$$



Stiff transport computation

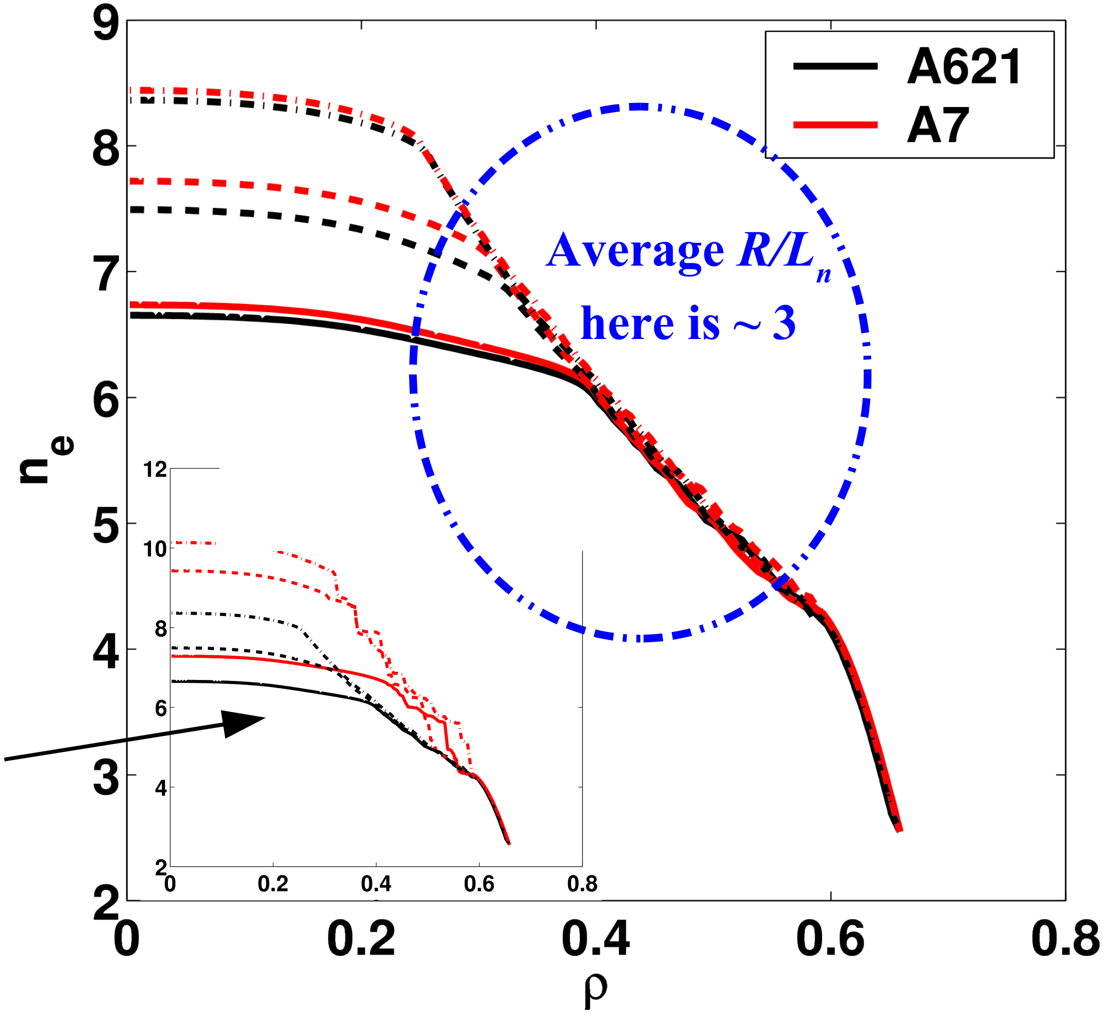
- Same case as before. Only n_e evolves (I have still to implement stiff transport terms into temperature equation).

$$\begin{cases} D = 0.3 + \left(\frac{R}{L_n} - 3\right) \\ V = -W_p - D\left(0.5\frac{R}{L_{Te}} - 5s\right) \\ D_{stiff} = 10 \end{cases}$$

Stiff term

[- If $D_{stiff} = 0$, the result is this:]

– t=0 ; –– t = 0.04 ; –. t=0.1



Benchmark on equilibrium computations

- Again, we use AUG #26328 @ t = 1.1 s (L-mode, diverted);

- T_e, T_i, n_e are fixed and do not evolve. ψ is let to evolve;

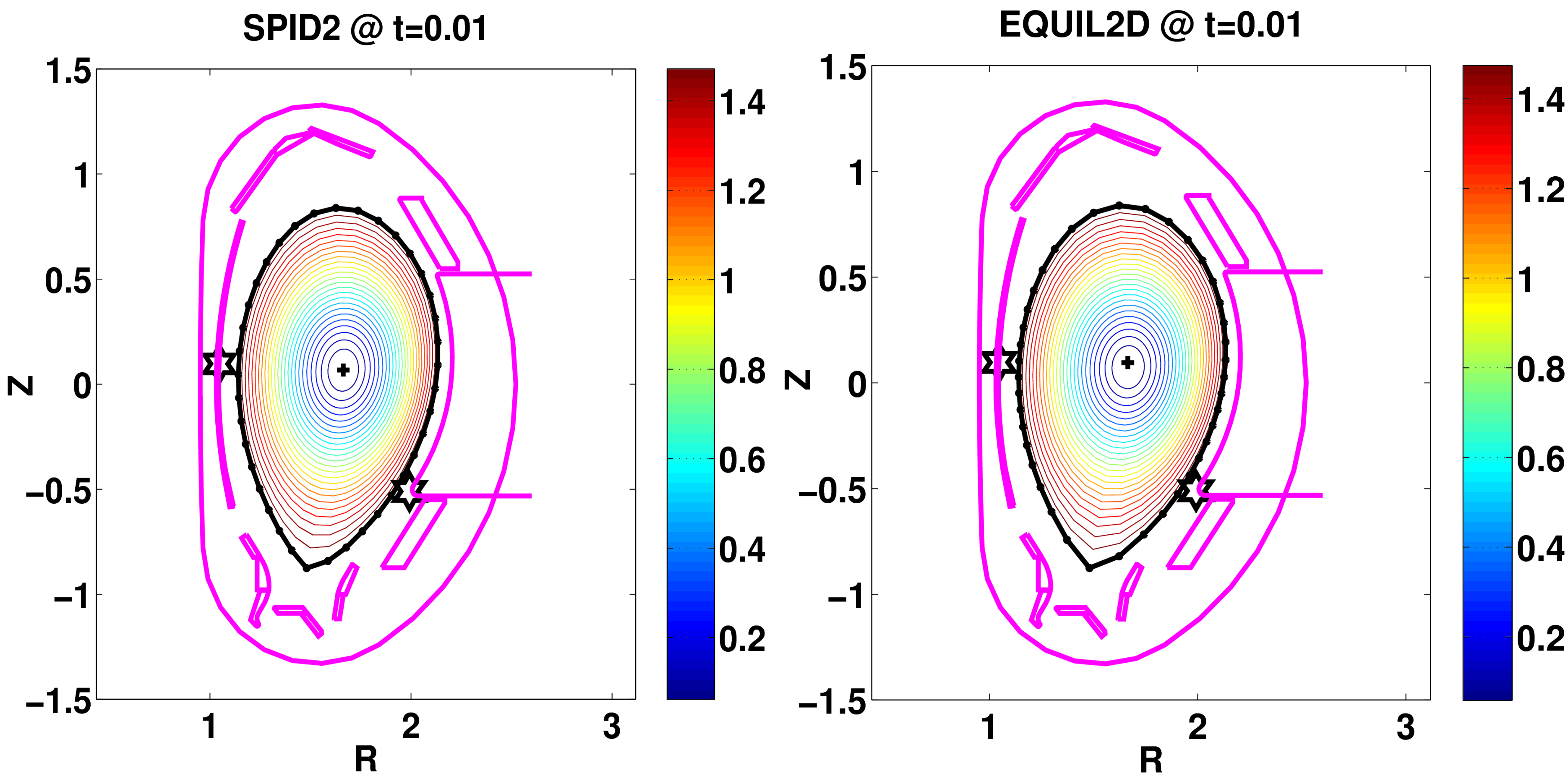
- We compare outcome using only ASTRA-7 and different equilibrium solver:

  * EMEQ
  * SPIDER
  * SPID2 (SPIDER included in the new iteration cycle)
  * EQUIL2D (my solver, still in progress though)

- For SPIDER, SPID2, EQUIL2D we use real numerical boundary from CLISTE-EQH, with 40 boundary points, and a radial grid of 24 points for SPIDER, SPID2, and EQUIL2D (run with *DGBSV* solver);

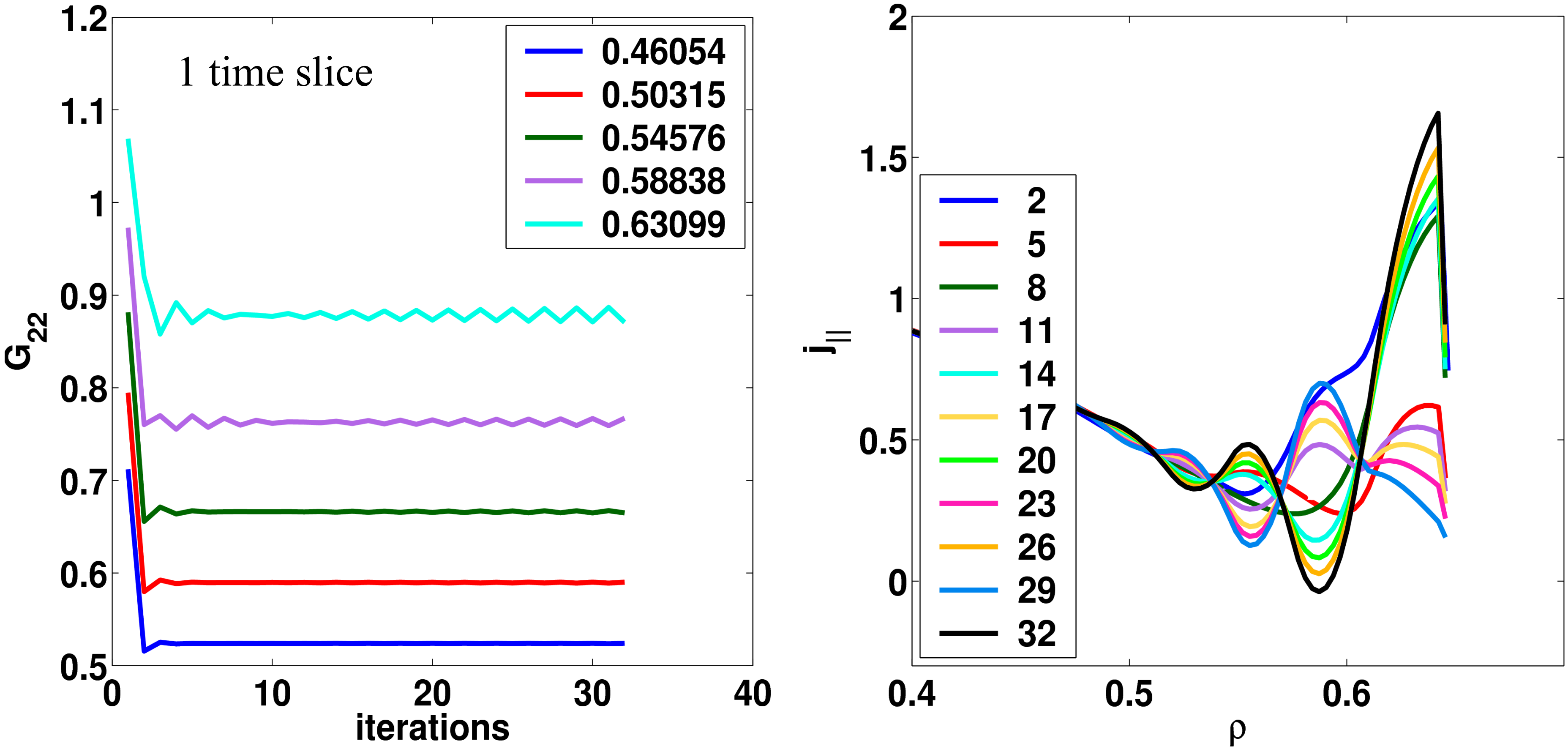
- EMEQ uses 41 radial points and analytical boundary, which looks like the CLISTE one, but is up-down symmetric, with no X-point.

Comparison of shape and poloidal flux



→ EQUIL2D can be used as a reliable cross-check tool for SPID2

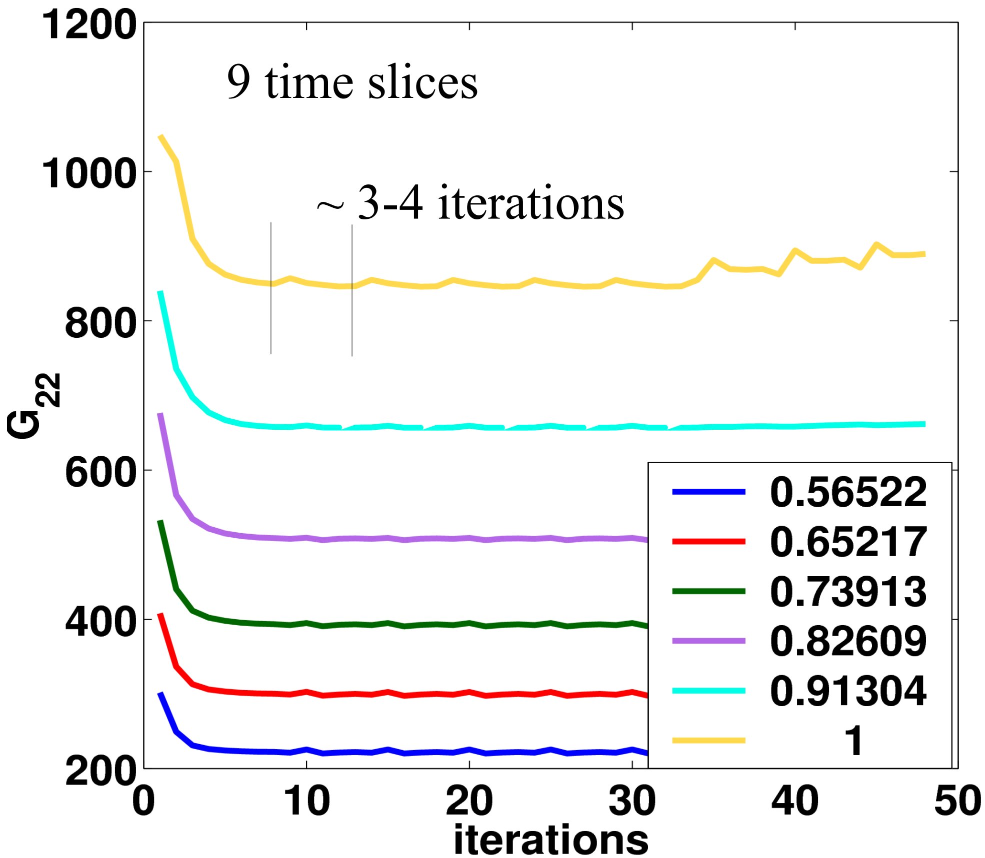
Convergence of SPIDER in the old scheme



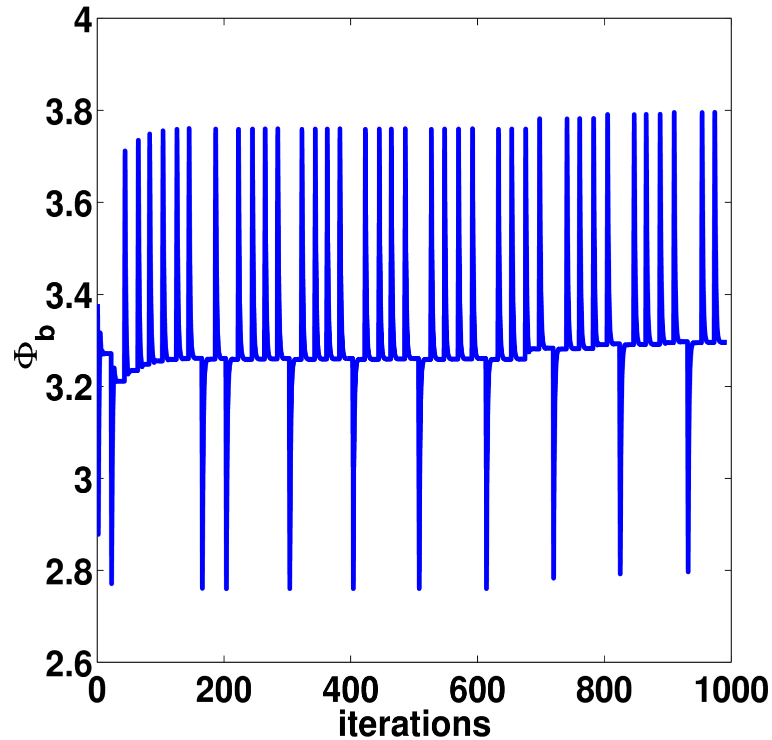
- Numerical instability associated to the non-consistency of the scheme;

- G22 is the most critical metric coefficients (enters in $j_{||}$ definition).

Convergence of SPID2 in the new scheme



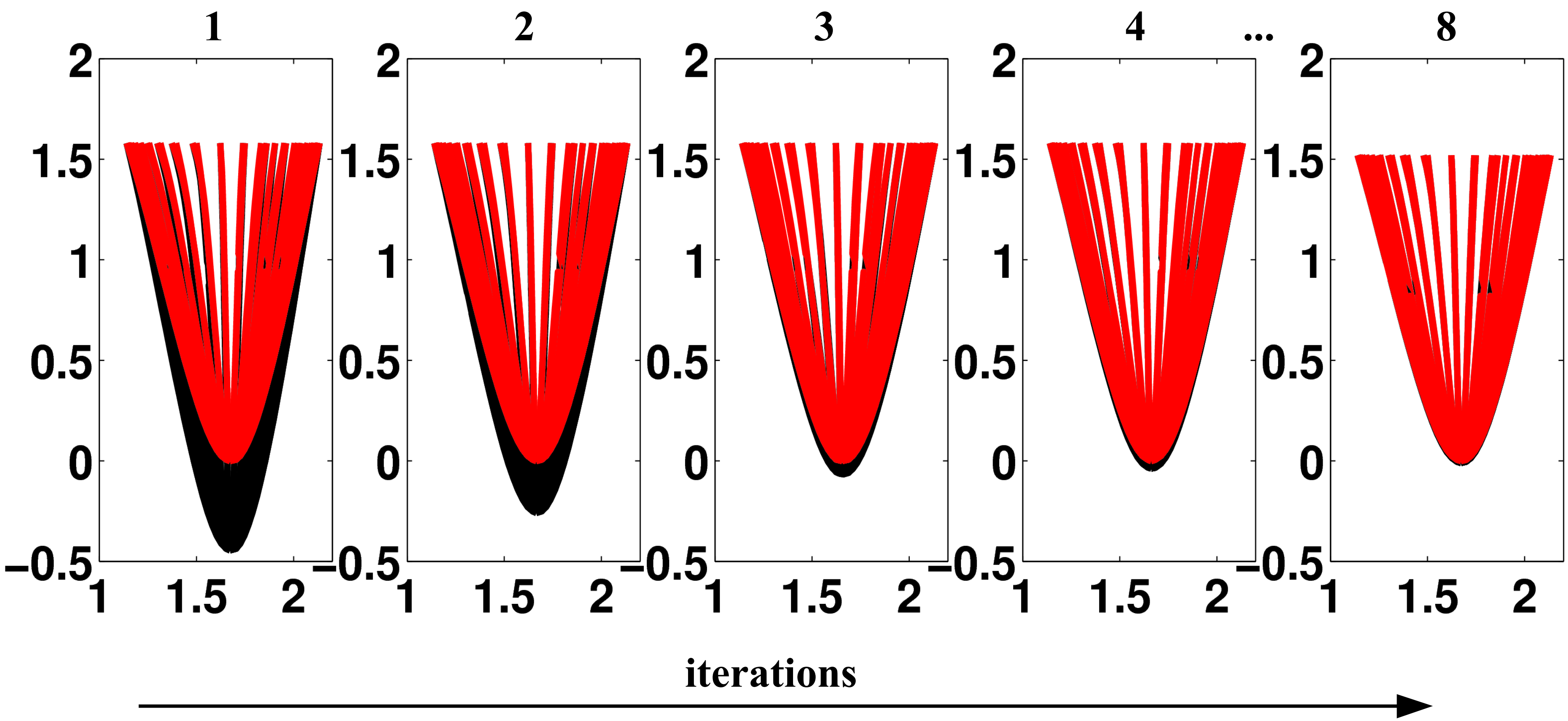
Requested accuracy on G_{22} is 10^{-3} (corresponding to 0.1%), while on Φ_b is 10^{-7}



- The new scheme is intrinsically stable and convergence is rather fast (could be improved with some more thinking);

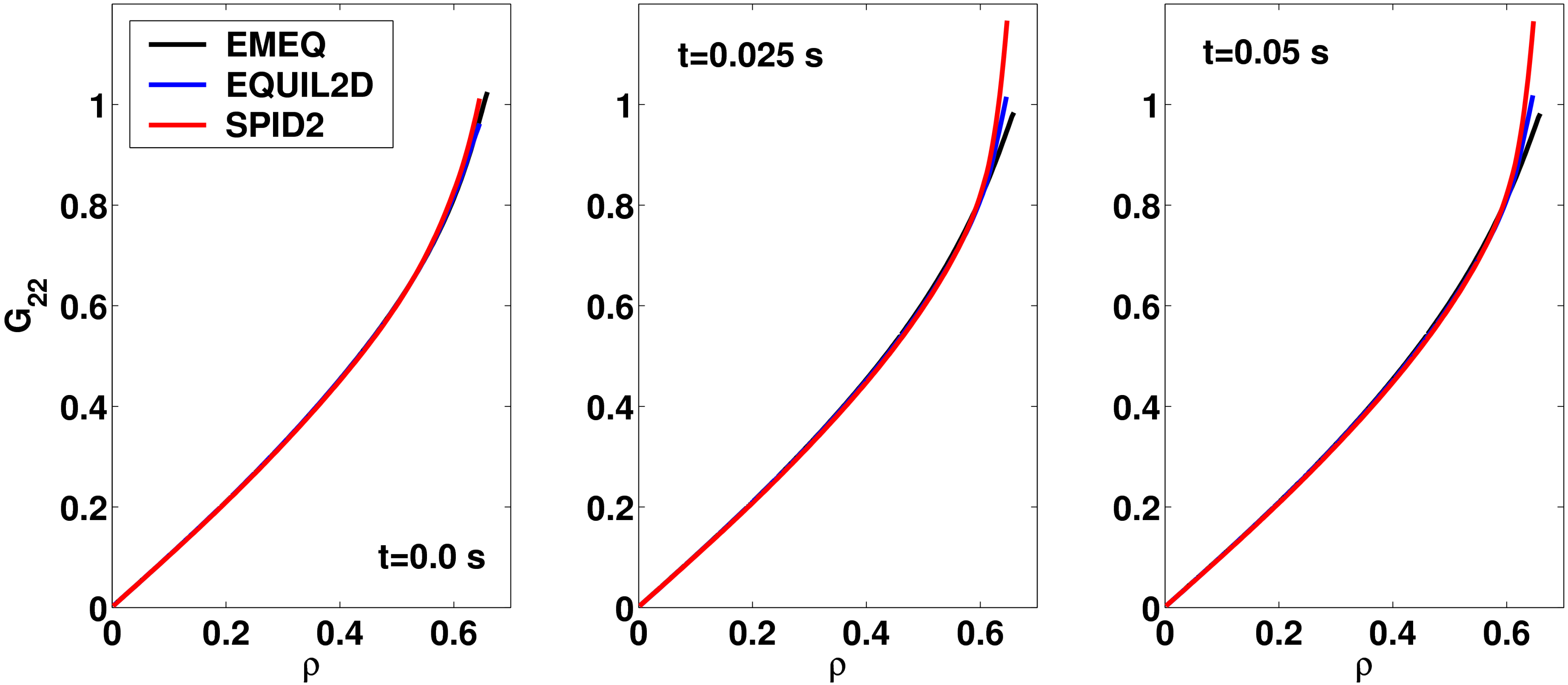
- After convergence, poloidal flux, metrics, and boundary fluxes are self-consistent and the poloidal flux is invariant, as shown in the next figure →

Convergence of poloidal flux



- Red : poloidal flux from current diffusion (mapped onto 2D grid);
- **Black : poloidal flux from EQUIL2D;**
- 1D and 2D poloidal fluxes are one and the same once convergence is reached

Comparison between codes: G_{22}

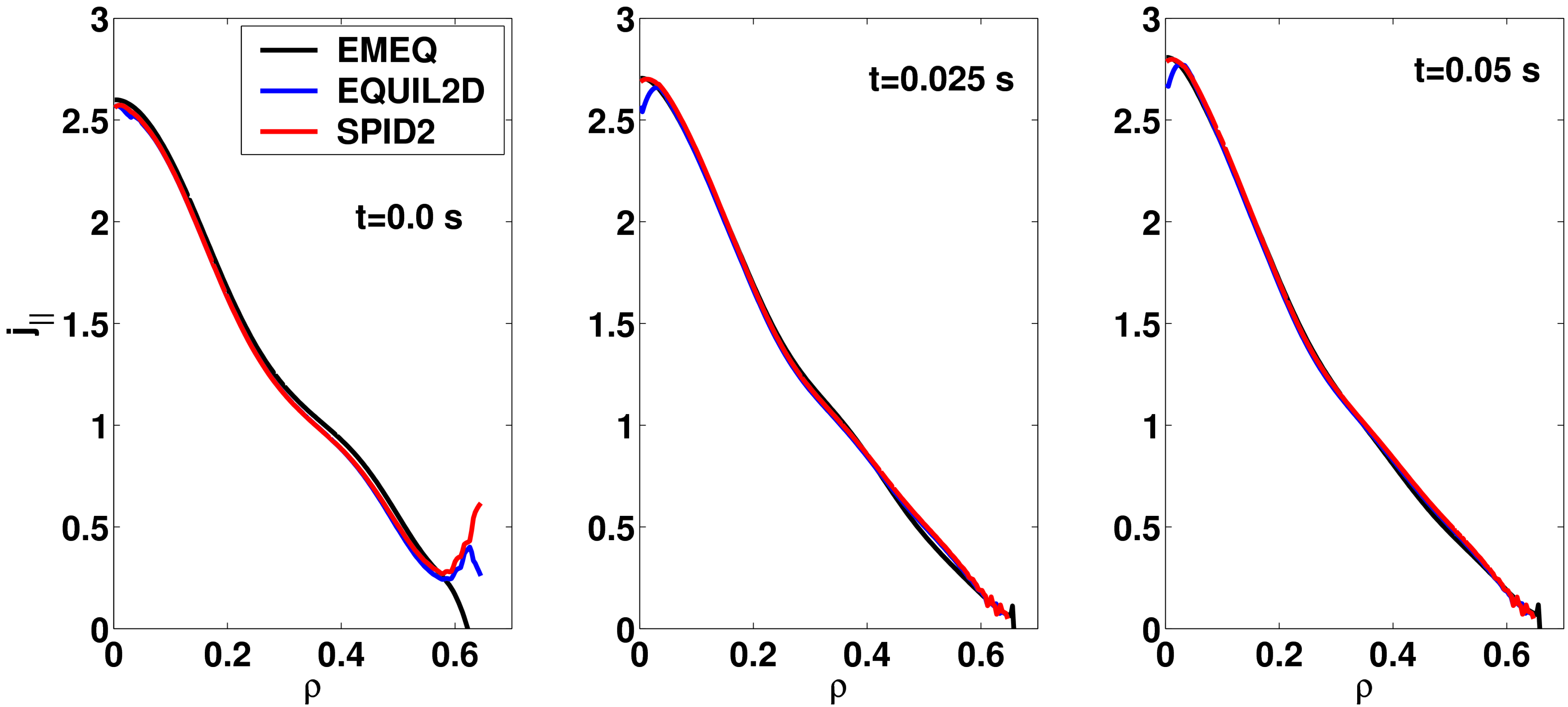


- Very good agreement between equilibrium codes, except at the very edge;

- EMEQ is up-down symmetric, however there is a large discrepancy between SPID2 and EQUIL2D;

- Computation of metric quantities has to be rechecked in both EQUIL2D and SPID2 (asap).

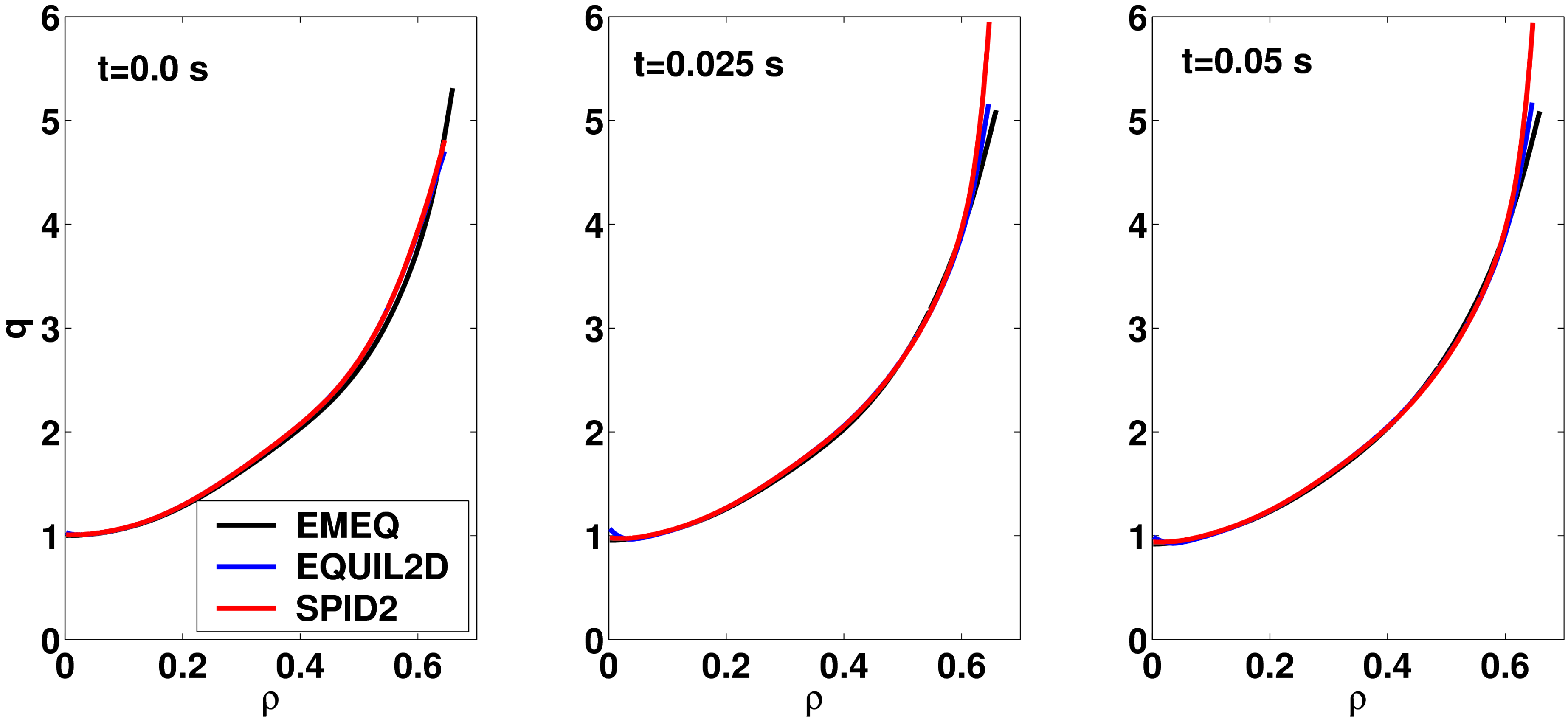
Comparison between codes: $j_{||}$



- Some problem in the very center due to interpolation routine, this also will be rechecked asap;

- In any case, $j_{||}$ is a derived quantity and does not enter in the iteration cycle (other schemes use it as the fundamental quantity in the iterations, that is why they are unstable).

Comparison between codes: *q*



First "dynamical" free-boundary computations

- Again, we use AUG #26328 @ t = 1.1 s (L-mode, diverted);

- T_e, T_i, n_e are fixed and do not evolve. ψ is let to evolve;

- SPIDER is used in free-boundary mode, coupled to ASTRA through the iteration procedure shown previously, however now *metric* is not converged;

- A rectangular grid of 129 x 129 points is used, which includes part of the vacuum region;

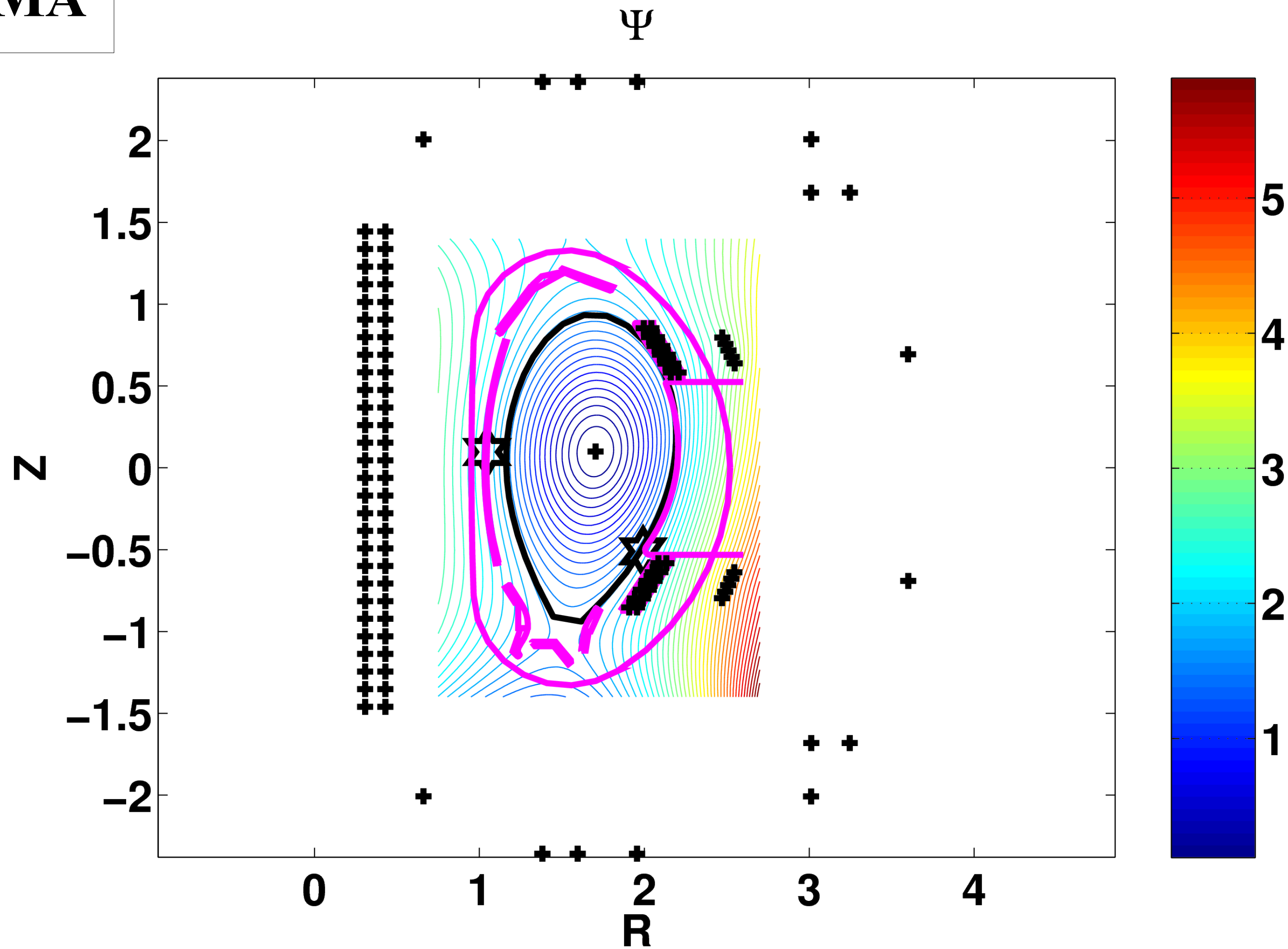
- The PF coils currents are taken from the MBI and MAI diagnostics (no halo, no vessel currents);

- The geometrical infos are taken from the ASDEX *xml* machine description file;

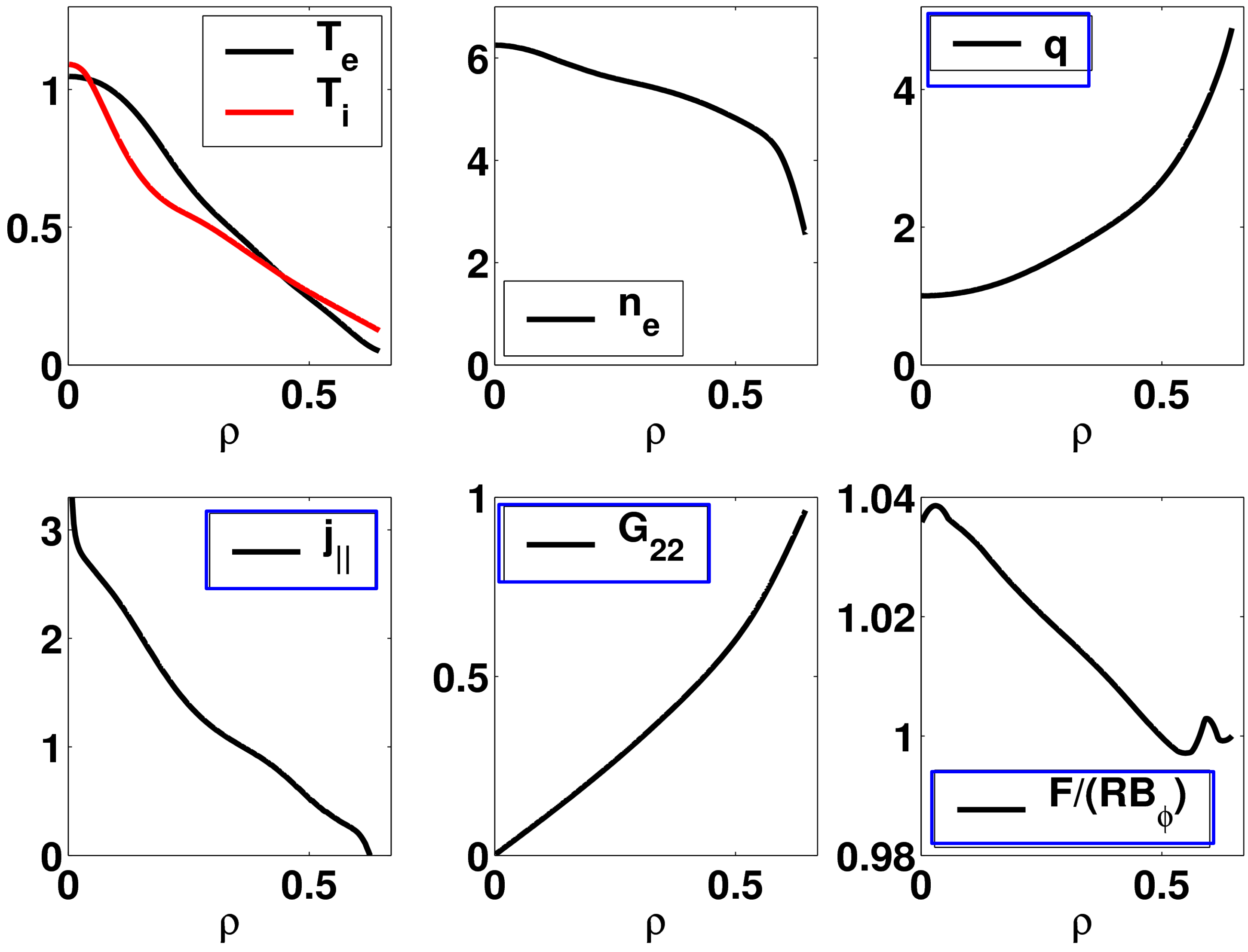
- There is no feedback on the coils (i.e. no circuit equations are solved), so changes in plasma boundary and positions at fixed coils currents are due to internal plasma profiles evolution.

Results after first time slice: global structure

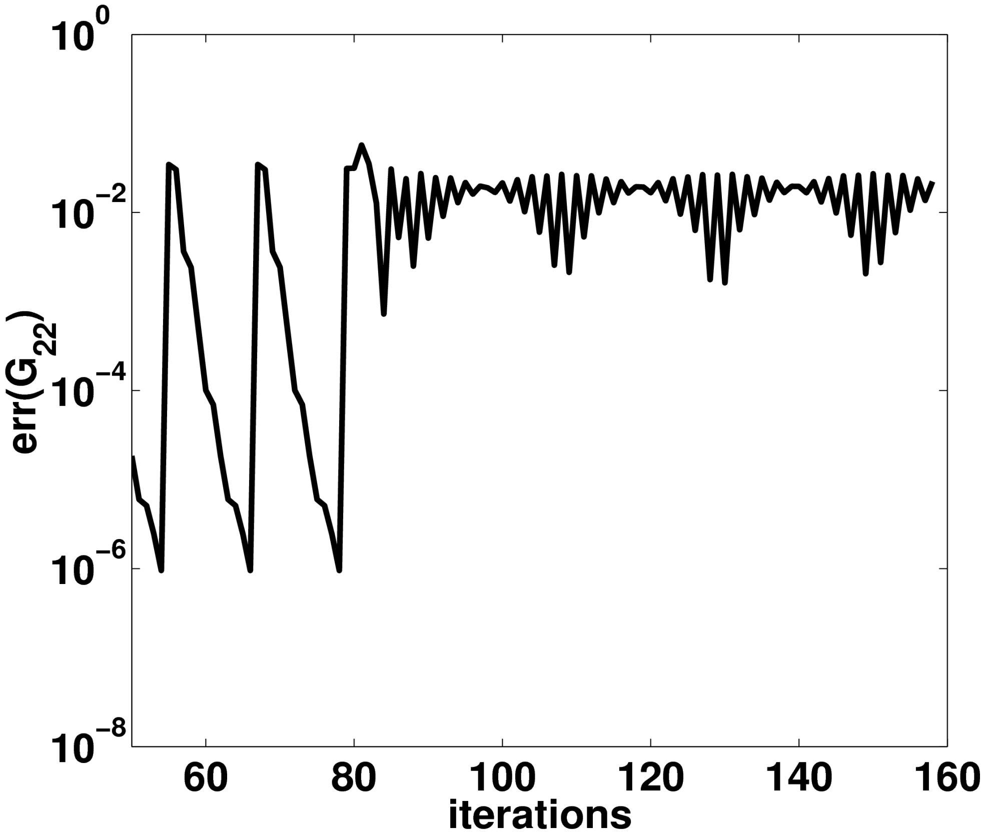
$I_p = 0.96$ MA



Results after first time slice: internal profiles



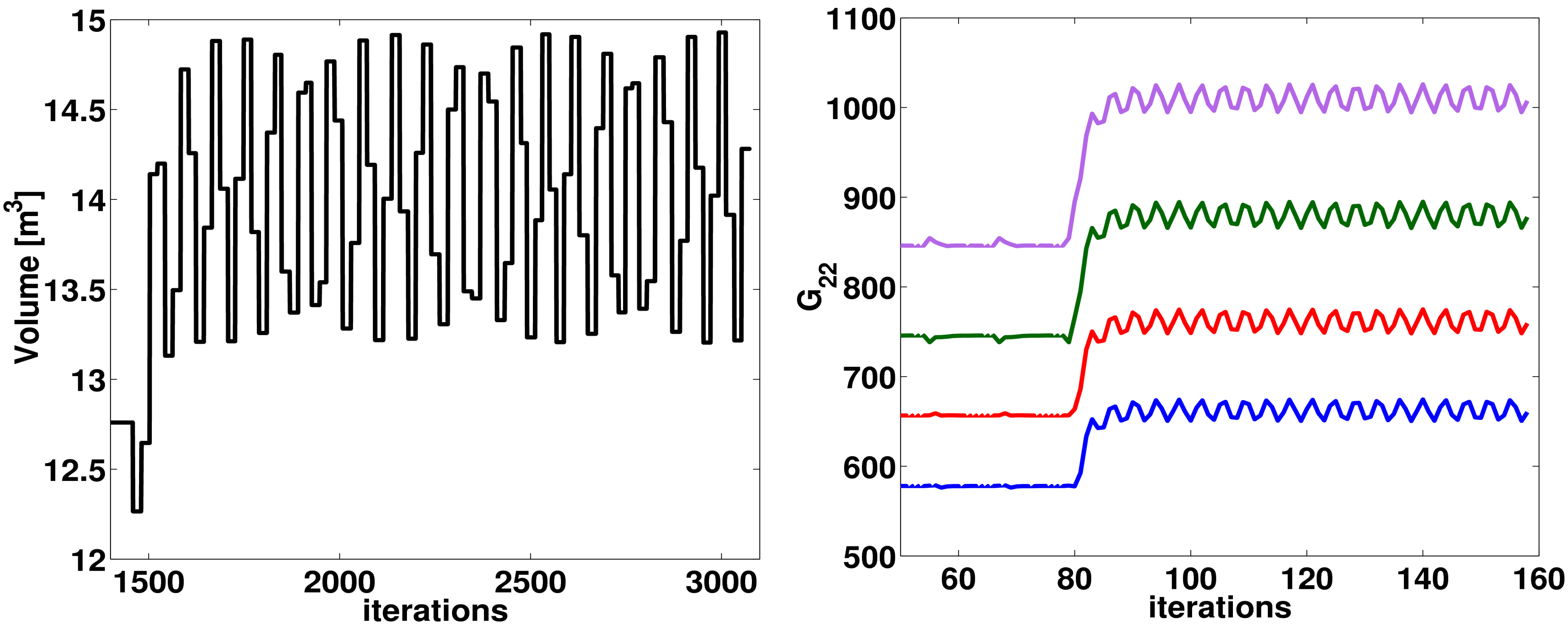
Problems with convergence due to volume oscillations?



- The free-boundary code in the iteration scheme enters in a neverending loop during which the relative error in the metric has some rather regular and beat-wave like oscillations, the nature of which is under investigations;

- Without metric coefficients convergence, the code would however run fine.

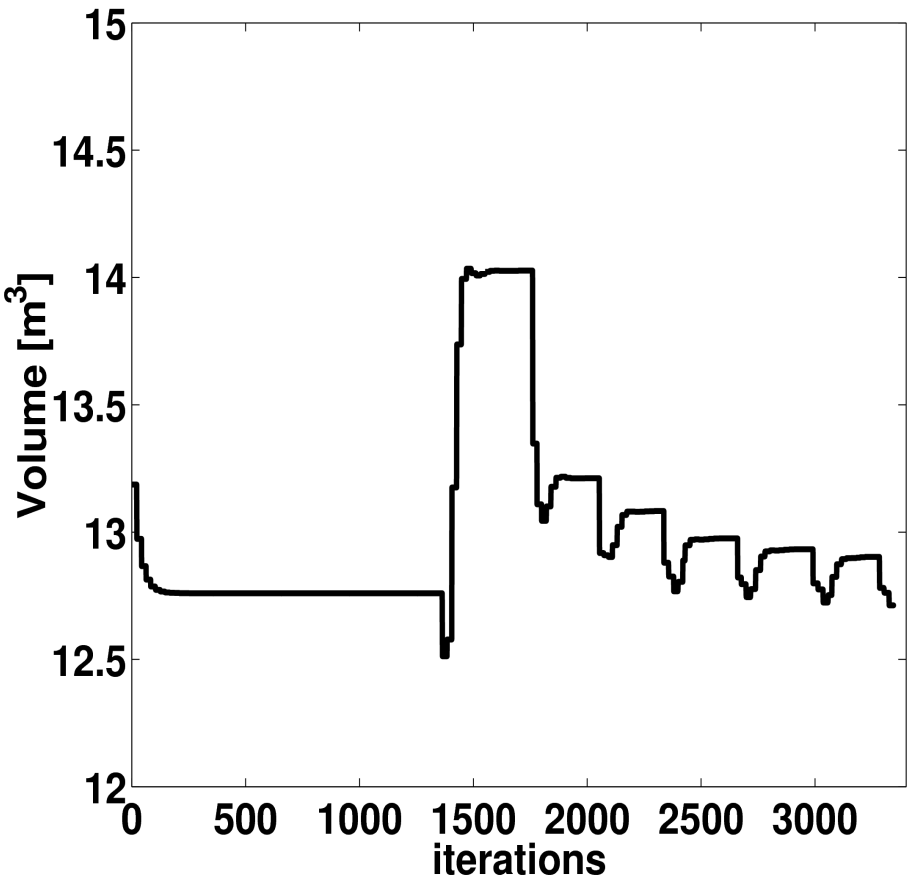
Problems with convergence due to volume oscillations?

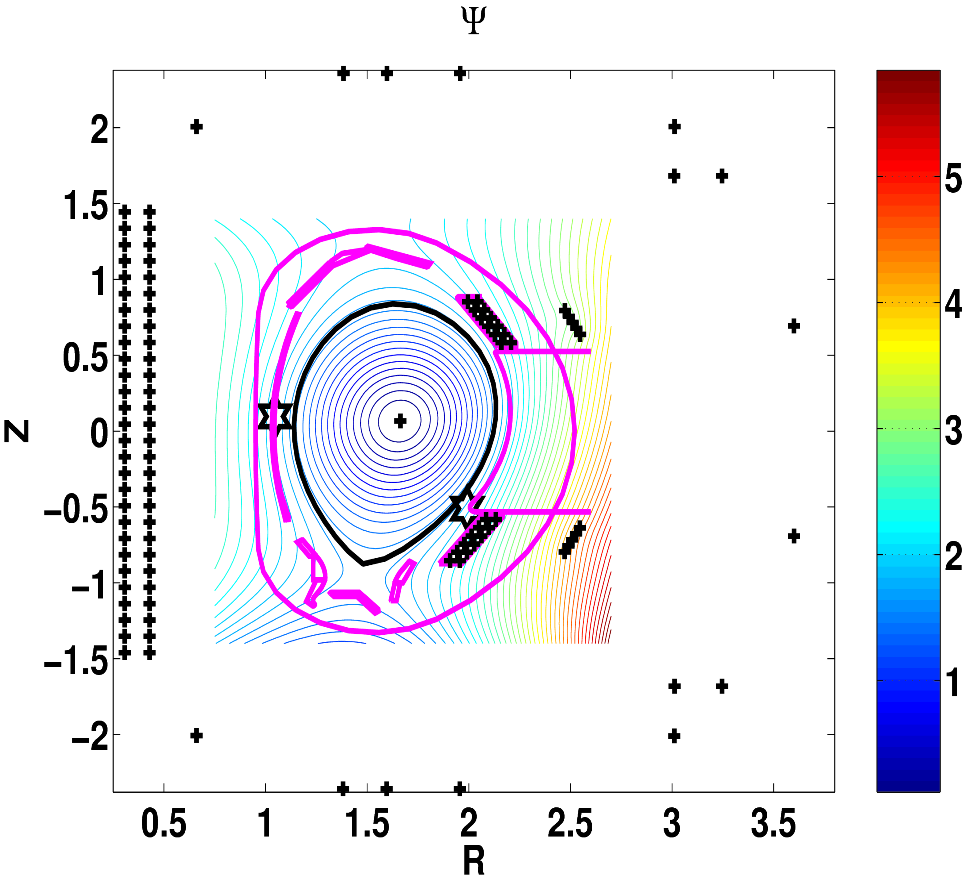


- It looks like an interplay between changes in metric and plasma 'breathing' during which the volume undergoes oscillations of the same nature of the relative error shown previously;

- Analysis of this numerical phenomenon and its solution are underway.

25 Jan 2012: problem solved!





- Under-relaxion is applied (factor 0.5), which causes automatic damping of pure imaginary Fourier modes;

- Now convergence is achieved in both the iteration scheme and also with ASTRA!

- Exponential decay of volume is due to internal plasma current diffusion.

Summary of ASTRA-7 main features

- Equations have been rewritten in normalized grid x, by adding grid convective terms to assure conservation properties are still satisfied;

- A theory-based toroidal momentum transport equation has been added;

- Dynamical coupling with equilibrium (either prescribed or free-boundary) has been approached in a novel fashion, to preserve consistency and stability. This has been accomplished by writing a dedicated Grad-Shafranov solver with exactly the requested properties. As a result, coupling with SPIDER has followed, and is succesfully working;

- Can also include TORBEAM with real equilibrium, and the recently developed turbulence model TGLF [G. Staebler et al.].

Future directions

- Free-boundary equilibrium computations with dynamical circuit equations;

- Include poloidal rotation (Pfirsch-Schlueter terms) in the momentum transport equation;

- Coupling with TORBEAM using numerical equilibrium under way;

- Coupling with TORIC (collaboration with R. Bilato, M. Brambilla);

- Coupling with STRAHL [R. Dux] will be taken care of soon.

* Far prospects: - SOL 1D mockup transport ?;
- Assess feasibility of breakdown simulations;
- GUI for free free boundary computations.

For ASTRA users

I suggest all ASTRA users to check out

this new version and try it!

As soon as final tests and adjustments are done, I will set up a user-interfaceable version in ~*astra* .

Feedback is very welcome for improvement.

Many thanks!