Preconditioning and nonlinear time solvers for the Jorek MHD code

E. Franck, M. Hölzl, E. Sonnendrücker

Max Planck Institute for Plasma physics, Garching, Germany

Theorieseminar of the IPP, Usedom, 20 November 2013

Outline

Physical context and models

MHD model

- **Context**: simulate the ELM's (edge-localized mode) to estimate the amplitude of these instabilities and understand how to control them.
- **Model**: The full- resistive MHD model given by

$$\begin{cases} \partial_t \rho + \nabla.(\rho \mathbf{v}) = \nabla.(D_{||}\nabla_{||}\rho + D_{\perp}\nabla_{\perp}\rho) + S_p \\ \rho\partial_t \mathbf{v} + \rho\mathbf{v}.\nabla\mathbf{v} + \nabla(\rho T) = \mathbf{J}\times\mathbf{B} + \nu\triangle\mathbf{v} \\ \rho\partial_t T + \rho\mathbf{v}.\nabla T + (\gamma-1)\rho T\nabla\mathbf{v} = \nabla.(K_{||}\nabla_{||}T + K_{\perp}\nabla_{\perp}T) + S_h \\ \partial_t \mathbf{B} = \nabla\times(\mathbf{v}\times\mathbf{B}) - \nabla\times\eta\mathbf{J} \end{cases} \quad (1)$$

with ρ the density, $\mathbf{v}$ the velocity, T the temperature, $\mathbf{B}$ the magnetic field and $\mathbf{J} = \nabla\times\mathbf{B}$ the current.

- The terms $D_{||}$, $D_{\perp}$, $K_{||}$, $K_{\perp}$ are anisotropic diffusion tensors.
- Source terms: S_h is the heat source, S_p is the particle source.

- *Reduced magnetohydrodynamic simulation of toroidally and poloidally localized edge localized modes*, M. Hölzl and co-workers, Phys. of Plasmas, 2012.

Reduced MHD: assumptions and derivation

- We consider the cylindrical coordinates $(R, Z, \phi) \in \Omega \times [0, 2\pi]$.
- (R, Z) corresponds to the poloidal plan and ϕ the toroidal direction.

Reduced MHD: assumptions

$$\mathbf{B} = \frac{F_0}{R}\mathbf{e}_\phi + \frac{1}{R}\nabla\psi \times \mathbf{e}_\phi \quad \mathbf{v} = -R\nabla u \times \mathbf{e}_\phi + v_{||}\mathbf{B}$$

with u the electrical potential and ψ the poloidal magnetic flux.

- We add the vorticity $w = \triangle_\perp u$ and the toroidal current $z_j = \triangle^*\psi$
- Derivation: Plugging **B** and **v** in the density, magnetic and energy equations + simplifications. For the equations on u and $v_{||}$ we use the following projections

$$\mathbf{e}_\phi.\nabla \times R^2 \left(\rho\partial_t\mathbf{v} + \rho\mathbf{v}.\nabla\mathbf{v} + \nabla(\rho T) = \mathbf{J} \times \mathbf{B} + \nu\triangle\mathbf{v}\right)$$

and

$$\mathbf{B}.\left(\rho\partial_t\mathbf{v} + \rho\mathbf{v}.\nabla\mathbf{v} + \nabla(\rho T) = \mathbf{J} \times \mathbf{B} + \nu\triangle\mathbf{v}\right).$$

Theoritical results

- To ensure numerical stability it is essential to obtain well-posed models
- Example of criterion for well posed models: Conservation of total energy

Model without parallel velocity

We assume that the boundary conditions are correctly chosen. We obtain the following energy estimate

$$\frac{d}{dt}\int_\Omega \left(\frac{|\nabla_\perp \psi|^2}{2R^2} + \widehat{\rho}\frac{|\nabla_\perp u|^2}{2} + \frac{1}{\gamma-1}p\right) = -\int_\Omega \eta(T)\frac{|\triangle^* \psi|^2}{R^2} - \int_\Omega \nu |\triangle_\perp u|^2$$

with $E = \rho\frac{|\mathbf{v}|^2}{2} + \frac{|\mathbf{B}|^2}{2} + \frac{1}{\gamma-1}p$ the total energy.

- If $\eta = \nu = 0$ the total energy is conserved.
- **Model with** $v_{||}$: total energy conservation not clear because some terms missing.
- These terms can explain the convergence problems in the nonlinear phase with small dissipation terms.

Jorek Code: description

Description of the jorek code I

- Jorek: Fortran 90 code, parallel (MPI+OpenMP)
- Determinate the equilibrium
  - Define the boundary of the computational domain
  - Create a first grid which is used to compute the aligned grid
  - Compute $\psi(R, Z)$ in the new grid.
- Compute equilibrium
  - Solve the Grad-Shafranov equation

$$R\frac{\partial}{\partial R}\left(\frac{1}{R}\frac{\partial \psi}{\partial R}\right) + \frac{\partial^2 \psi}{\partial Z^2} = -R^2\frac{\partial p}{\partial \psi} - F\frac{\partial F}{\partial \psi}$$

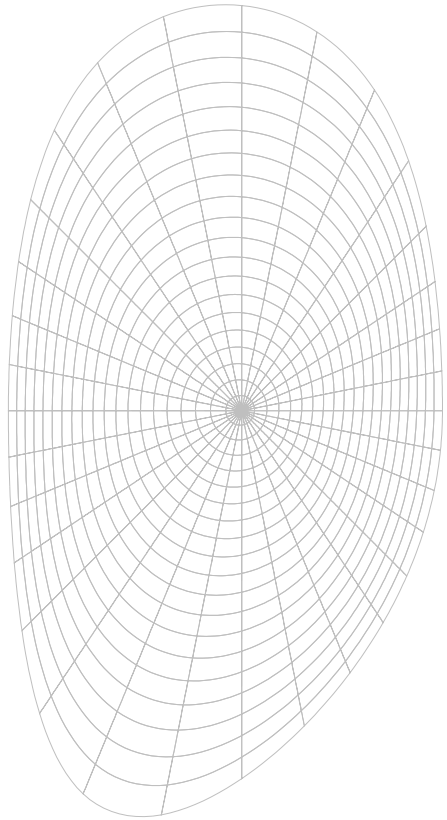

Figure: unaligned grid

Description of the jorek code II

- Computation of aligned grid
  - Identification of the magnetic flux surfaces
  - Create the aligned grid (with x-point)
  - Interpolate $\psi(R, Z)$ in the new grid.
- Recompute equilibrium of the new grid.
- Time-stepping (restart)
  - Construction of the matrix and some profiles (diffusion tensors, sources terms)
  - Solve linear system
  - Update solutions
- Spatial discretization:
  - For the poloidal plan: finite element method.
  - For the toroidal direction: Fourier expansion.

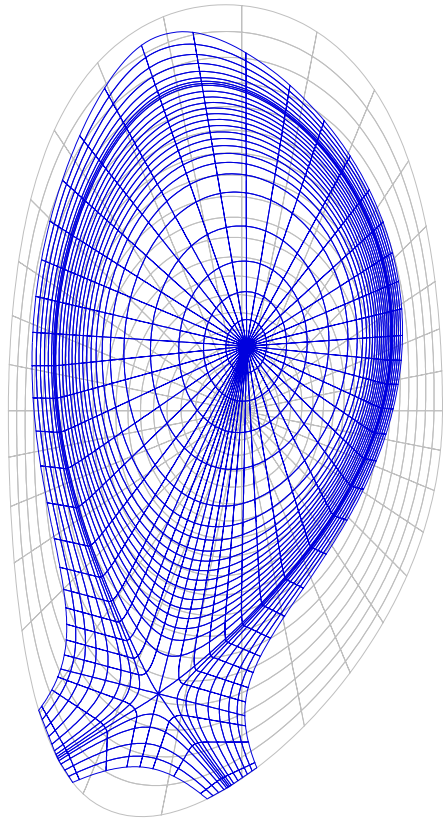

Figure: Aligned grid

Time scheme in Jorek code

- We recall the model $A(\partial_t \mathbf{U}) = B(\mathbf{U}, t)$
- For time stepping we use a Crank Nicholson :

$$A(\mathbf{U}^{n+1}) - A(\mathbf{U}^n) = \theta \Delta t B(\mathbf{U}^{n+1}) + (1-\theta)\Delta t B(\mathbf{U}^n)$$

- Defining $G(\mathbf{U}) = A(\mathbf{U}) - \theta \Delta t B(\mathbf{U})$ and

$$b(\mathbf{U}^n) = A(\mathbf{U}^n) + (1-\theta)\Delta t B(\mathbf{U}^n)$$

we obtain the nonlinear problem

$$G(\mathbf{U}^{n+1}) = -G(\mathbf{U}^n) + b(\mathbf{U}^n)$$

- First order linearization

$$\left(\frac{\partial G(\mathbf{U}^n)}{\partial \mathbf{U}^n}\right) \delta \mathbf{U}^n = \tilde{G}(\mathbf{U}^n)$$

with $\delta \mathbf{U}^n = \mathbf{U}^{n+1} - \mathbf{U}^n$, $\tilde{G}(\mathbf{U}^n) = -G(\mathbf{U}^n) + b(\mathbf{U}^n)$ and $J = \frac{\partial G(\mathbf{U}^n)}{\partial \mathbf{U}^n}$ the jacobian matrix of $G(\mathbf{U}^n)$.

Time scheme in Jorek code

- Linear solver in Jorek: Left Preconditioning + GMRES iterative solver.
- Principle of preconditioning step:
  - Replace the problem $J\delta\mathbf{U}_n = \tilde{G}(\mathbf{U}^n)$ by $P(P^{-1}J)\delta\mathbf{U}_k = \tilde{G}(\mathbf{U}^n)$.
  - Solve the new system with two steps $P\delta\mathbf{U}_n^* = \tilde{G}(\mathbf{U}^n)$ and $(P^{-1}J)\delta\mathbf{U}_n = \delta\mathbf{U}_k^*$
- If P is easier to invert than J and $P \approx J$ the linear solving step is more robust and efficient.
- Construction and inversion of P
  - P: diagonal block matrix where the submatrices are associated to each toroidal mode.
  - Inversion of P: We factorize and invert exactly each subsystem.
- This preconditioning is based on the assumption that the coupling between toroidal modes is weak.
- In practice, for the nonlinear phase this coupling can be strong.

Jorek code: convergence issues

Problem :

- For some test cases the GMRES method does not converge in the nonlinear phase even with small time steps.

- Why ?
- The preconditioning is not sufficient to obtain a robust GMRES method ?
- Numerical instabilities are generated ?
  - The spatial poloidal and time discretization are not adapted ? Problem of positivity ?
  - The models are not stable ?

Nonlinear solvers and time stepping

Inexact Newton scheme

- At the time step n, we compute $b(\mathbf{U}^n)$, $G(\mathbf{U}^n)$ and $\mathbf{U}_0 = \mathbf{U}^n$ and ε_0.
- Step k of the Newton procedure
  - We compute $G(\mathbf{U}_k)$ and $\left(\frac{\partial G}{\partial \mathbf{U}_k}\right)$
  - We solve the linear system with GMRES

$$\left(\frac{\partial G(\mathbf{U}_k)}{\partial \mathbf{U}_k}\right) \delta \mathbf{U}_k = \tilde{G}(\mathbf{U}_k) = b(\mathbf{U}^n) - G(\mathbf{U}_k)$$

and the following convergence criterion

$$\frac{||\left(\frac{\partial G}{\partial \mathbf{U}_k}\right) \delta \mathbf{U}_k + \tilde{G}(\mathbf{U}_k)||}{||\tilde{G}(\mathbf{U}_k)||} \leq \varepsilon_k, \quad \varepsilon_k = \gamma \left(\frac{||\tilde{G}(\mathbf{U}_k)||}{||\tilde{G}(\mathbf{U}_{k-1})||}\right)^{\alpha}$$

  - We iterate with $\mathbf{U}_{k+1} = \mathbf{U}_k + \delta \mathbf{U}_k$ and apply the convergence test.
- If the Newton procedure stops we define $\mathbf{U}^{n+1} = \mathbf{U}_{k+1}$.
- For the Newton procedure it not necessary to solve GMRES step with high accuracy.
- Inexact Newton procedure minimize the number of GMRES iteration.

The continuation method

- The Newton algorithm converges if the initial solution is not too far from the good one.
- **Aim**: to give a good initial solution for the Newton solver with a continuation method.
  - Nonlinear problem: $R(\mathbf{U}) = 0$ not easy to solve.
  - **Idea**: replace the initial problem by the homotopy mapping $F(\mathbf{U}, d) = 0$ easier to solve.
  - Dissipation continuation: $F(\mathbf{U}, d) = 0 = R(\mathbf{U}) + dD(\mathbf{U})$ with D a diffusion operator.
- **Algorithm**: we solve $F(\mathbf{U}, d_i) = 0$ for a decreasing set of d_i and use the previous solution to initialize the Newton procedure of the current step.

Conclusion about continuation and Newton methods

- **Inexact Newton procedure** :
  - For difficult cases, the convergence problem is not solved by the Newton procedure.
  - In other cases the Newton method allows to use bigger time step and avoid the accumulation of time error which can generate instabilities
- **Adaptive time stepping**:
  - The Newton procedure is coupled with an adaptive time stepping based on the nonlinear residue.
- **Conclusion**: The Inexact Newton procedure with adaptive time stepping is more robust then the previous time scheme.
- Additionnal test cases and numerical studies are necessary.

- **Continuation method** :
  - For now, the continuation method is not helpful for solving the difficult cases.
  - The continuation method can help explain convergence problems.
- **Other way** : Find a more efficient preconditioning for reduced and full MHD.

Preconditioning

Preconditioning idea I

- *An optimal, parallel fully implicit Newton-Krylov solver for 3D viscoresistive Magnetohydrodynamics*, L. Chacon, Phys. of plasma, 2008.
- *Scalable parallel implicit solvers for 3D magnetohydrodynamics*, L. Chacon, Journal of Phys. 2009.

- **Right preconditioning**: We solve $JP^{-1}P\delta\mathbf{U}_k = G(\tilde{\mathbf{U}}_k)$.
- **Aim**: Find P easy to invert with $P \approx J^{-1}$ and more efficient in the nonlinear phase as the current preconditioning.

- **Idea**: Operator splitting + parabolic formulation of the MHD + multigrid methods.
- Example

$$\begin{cases} \partial_t u = \partial_x v \\ \partial_t v = \partial_x u \end{cases} \longrightarrow \begin{cases} u^{n+1} = u^n + \Delta t \partial_x v^{n+1} \\ v^{n+1} = v^n + \Delta t \partial_x u^{n+1} \end{cases}$$

- We obtain $(1 - \Delta t^2 \partial_{xx}) u^{n+1} = u^n + \Delta t \partial_x v^n$.
- The matrix associated to $(1 - \Delta t^2 \partial_{xx})$ is diagonal dominant and well conditioned.

Preconditioning : simple example I

- We assume T constant, $\rho = \frac{1}{R^2}$, $\mathbf{B} = \frac{F_0}{R} e_\phi + \frac{1}{R} \nabla\psi \times e_\phi$ and $\mathbf{v} = -R\nabla u \times e_\phi$
- The model obtained is

$$\begin{cases} \partial_t \psi = R[\psi, u] + \eta \triangle^* \psi - F_0 \partial_\phi u \\ \\ \partial_t \triangle_\perp u = \frac{1}{R}[R^2 \triangle_\perp u, u] + \frac{1}{R}[\psi, \triangle^* \psi] - \frac{F_0}{R^2} \partial_\phi \triangle^* \psi + \nu \triangle_\perp (\triangle_\perp u) \end{cases}$$

  with $w = \triangle_\perp u$ and $z_j = \triangle^* \psi$.
- In this formulation the evolution equations and elliptic equations are non coupled.
- The Jacobian associated to the evolution equations is

$$\frac{\partial G(\mathbf{U}^k)}{\partial \mathbf{U}^k} \delta\mathbf{U}^k = J\delta\mathbf{U}^k = \begin{pmatrix} M & U \\ L & D \end{pmatrix} \delta\mathbf{U}^n$$

  with $\delta\mathbf{U}^k = (\delta\psi, \delta u)$
- M and D the matrices associated to advection and diffusion operators on ψ and u.
- L and U associated to the coupling operators between ψ and u.

Preconditioning : Algorithm

- The final system with Schur decomposition is given by

$$\delta \mathbf{U}^k = J^{-1}\tilde{G}(\mathbf{U}^k) = \begin{pmatrix} M & U \\ L & D \end{pmatrix}^{-1} \tilde{G}(\mathbf{U}^k)$$

$$= \begin{pmatrix} I & M^{-1}U \\ 0 & I \end{pmatrix} \begin{pmatrix} M^{-1} & 0 \\ 0 & P_{schur}^{-1} \end{pmatrix} \begin{pmatrix} I & 0 \\ -LM^{-1} & I \end{pmatrix} \tilde{G}(\mathbf{U}^k)$$

  with $P_{schur} = D - L_a M^{-1} U$ ($L_a \approx L$).
- Algorithm to solve $J\delta\mathbf{U}_k = \tilde{G}(\mathbf{U}^k)$ + elliptic equations:

$$\begin{cases} \text{Predictor :} & M\delta\psi_p^k = \tilde{G}_\psi \\ \text{potential update :} & D_\perp \tilde{P}_{schur}\delta u^k = \left(-L_a\delta\psi_p^k + \tilde{G}_u\right) \\ \text{Corrector :} & M\delta\psi^k = M\delta\psi_p^k - U\delta u^k \\ \text{Current update :} & \delta z_j^k = D_*\delta\psi^k \\ \text{Vorticity update :} & \delta w^k = D_\perp \delta u^k \end{cases}$$

  with $\tilde{G}_\psi$ and $\tilde{G}_u$ the rhs for equations on ψ and u. $D_\perp$ is the elliptic operator.
- In the potential update step we have factorized the system by $\triangle_\perp$.

Preconditioning : Approximation of the Schur complement

- To define $\tilde{P}_{schur} = D - L_a M^{-1} U$ we must know the matrix M^{-1}.
- The previous algorithm with a Schur complement approximation gives the preconditionning P.
- **Small flow approximation**
  - In $\tilde{P}_{schur}$ we assume that $M^{-1} \approx \Delta t$

$$\tilde{P}_{schur} = \frac{\delta u}{\Delta t} - \frac{\theta}{R}[R^2 \delta u, u^k] - \theta \nu \triangle_\perp \delta u - \theta^2 \Delta t \left(\mathbf{B}^k . \nabla (R^2 \mathbf{B}^k . \nabla \delta u) \right)$$

- $\mathbf{B}^n . \nabla (R^2 \mathbf{B}^n . \nabla \delta u)$ is a positive self-ajoint second order wave operator.
- **Arbitrary flow approximation**
  - We introduce an operator M_* with $UM_* \approx MU$. Consequently $P_{Schur} = (DM_* - L_a U) M_*^{-1}$.
  - In this case the Potential udapte step in given by

$$\begin{cases} \text{potential update I :} & (DM_* - L_a U)\delta u^{*,k} = \left(-L_a \delta \psi_p^k + \tilde{G}_u \right) \\ \text{potential update II :} & \delta u^k = M_* \delta u^{*,k} \end{cases}$$

  - The operator M_* is the advection operator.

Preconditioning: Remarks and future work

- With more difficult calculus and additional tools we can extend the algorithm for the model with temperature arbitrary density and parallel velocity.
- The Schur preconditioning method uses an approximation of the Jacobian based on the approximation of the Schur complement and the coupling hyperbolic terms.
- Contrary to the previous preconditioning the coupling terms between the Fourier modes are not neglected.
- This preconditioning is easily compatible with free jacobian method.

- **Future work**
  - Justify the approximations of the operators with a spectral analysis.
  - Complete the derivation and the study of the algorithm for the different models.
  - Validate the Algorithm and implement the method in Jorek with multigrid method and Free-Jacobian Newton solver.

Thanks

Thanks for your attention