Numerical issues for nonlinear MHD Jorek code

Emmanuel Franck
Max Planck Institute For Plasma Physics (Munich)
Seminar ITER, UPMC
23/05/2013

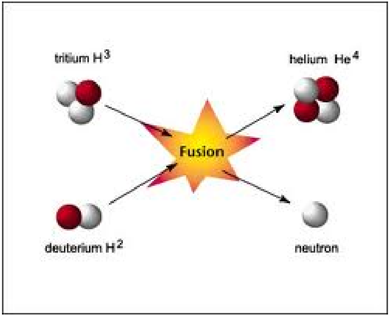
May 22, 2013

Table of contents

Physical context and models

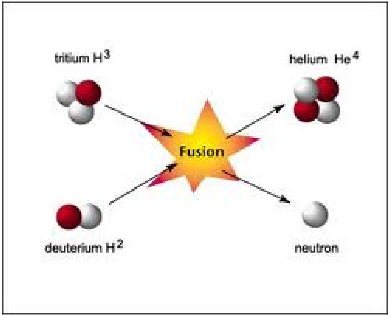
Iter

- **Fusion DT**: Reaction between Deuterium and tritium which product Helium and energy. The deuterium and tritium form a plasma (ionized gas).



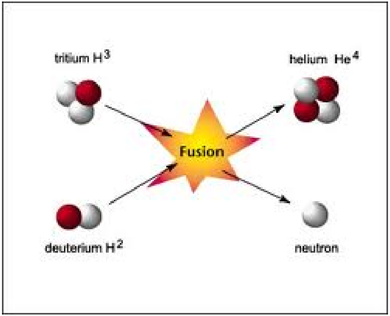
Iter

- **Fusion DT**: Reaction between Deuterium and tritium which product Helium and energy. The deuterium and tritium form a plasma (ionized gas).
- **Iter**: International project to prove the efficiency of controlled fusion as a power source. Iter is an experimental power plant using fusion.



Iter

- **Fusion DT**: Reaction between Deuterium and tritium which product Helium and energy. The deuterium and tritium form a plasma (ionized gas).
- **Iter**: International project to prove the efficiency of controlled fusion as a power source. Iter is an experimental power plant using fusion.
- **Magnetic confinement**: The plasma obtained by the reaction is confined in the center of the reactor (tokamak) using a powerful magnetic field.



Iter

- **Fusion DT**: Reaction between Deuterium and tritium which product Helium and energy. The deuterium and tritium form a plasma (ionized gas).
- **Iter**: International project to prove the efficiency of controlled fusion as a power source. Iter is an experimental power plant using fusion.
- **Magnetic confinement**: The plasma obtained by the reaction is confined in the center of the reactor (tokamak) using a powerful magnetic field.
- **Tokamak**: Toroidal room used for the plasma confinement.

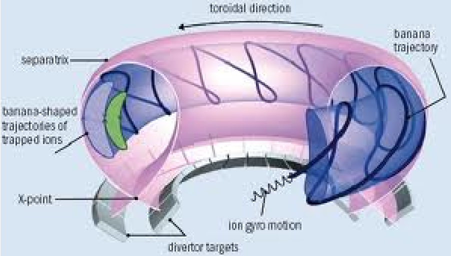


Figure: Tokamak

Models for Iter

- The dynamic of the plasmas in Iter is a very difficult multiscale problem.

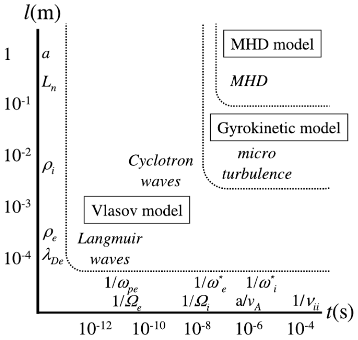


Figure: Spatial and time scales

Models for Iter

- The dynamic of the plasmas in Iter is a very difficult multiscale problem.
- We have different models for the different time and space scales :
  - Kinetic Vlasov-Maxwell equation not used in pratice (CPU cost very important).
  - Gyrokinetic approximation of the Vlasov-Maxwell equation used for the turbulence in the core Tokamak.
  - MagnetoHydrodynamics fluids models (resistive MHD, two fluids MHD) used to simulate the edge instabilities.

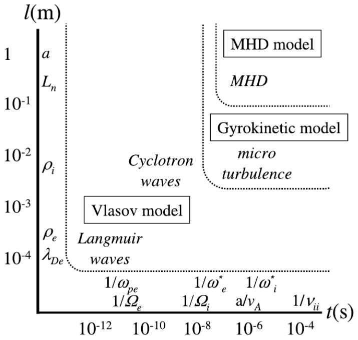


Figure: Spatial and time scales

ELMs and instabilities

- An edge-localized mode ("ELM's") is a disruptive instability occurring in the edge region of a tokamak plasma.
- The development of edge-localized modes poses an important challenge in magnetic fusion research with tokamaks. Instabilities can damage wall components due to their extremely high energy transfer rate.
- **Aim**: simulate the ELM's to estimate the amplitude of these instabilities and understand how control these.

- *MHD stability in X-point Geometry: simulation of ELMs*, G. Huysmans, O. Czarny, Nuclear fusion, 2007.
- *Reduced magnetohydrodynamic simulation of toroidally and poloidally localized edge localized modes*, M. Hölzl and co-workers, Phys. of Plasmas, 2012.

MHD model

- The full - resistive MHD model is given by

$$\begin{cases} \partial_t \rho + \nabla.(\rho \mathbf{v}) = \nabla.(D_{||}\nabla_{||}\rho + D_{\perp}\nabla_{\perp}\rho) + S_p \\ \rho\partial_t \mathbf{v} + \rho\mathbf{v}.\nabla\mathbf{v} + \nabla(\rho T) = \mathbf{J}\times\mathbf{B} + \nu\triangle\mathbf{v} \\ \rho\partial_t T + \rho\mathbf{v}.\nabla T + (\gamma - 1)\rho T\nabla\mathbf{v} = \nabla.(K_{||}\nabla_{||}T + K_{\perp}\nabla_{\perp}T) + S_h \\ \partial_t \mathbf{B} = \nabla\times(\mathbf{v}\times\mathbf{B}) - \nabla\times\eta\mathbf{J} + S_c \\ \nabla.\mathbf{B} = \mathbf{0} \end{cases} \tag{1}$$

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.
- We add source terms. S_c correspond to the current source, S_h correspond to the heat source, S_p correspond to the particle source.

Reduced MHD: assumption and derivation

- We consider the cylindric coordinate $(R, Z, \phi) \in \Omega \times [0, 2\pi]$.
- (R, Z) correspond 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.

- For the reduced MHD the quantities are ρ, T, Ψ, u,$v_{||}$ the parallel velocity, w the vorticity and z_j the toroidal current.
- Derivation: Plug $\mathbf{B}$ and $\mathbf{v}$ in the density, magnetic and energy equations. For the equations on u and $v_{||}$ we use

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

and

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

Basic Reduced MHD: model 199

- With $v_{||} = 0$ we obtain the model 199 considered in this talk.
- We solve $\partial_t A(\mathbf{U}) = B(\mathbf{U}, t)$ with

$$B(\mathbf{U}) = \begin{pmatrix} [\Psi, u] - \epsilon \frac{F_0}{R}\partial_\phi u + \frac{\eta(T)}{R}(z_j - S_c(\Psi)) - \eta_n \nabla.(\nabla z_j) \\ \frac{1}{2}[R^2||\nabla u||^2, \hat{\rho}] + [R^2\hat{\rho} w, u] + [\Psi, z_j] - \epsilon \frac{F_0}{R}\partial_\phi z_j - [R^2, p] \\ +\nabla.(R\nu(T)\nabla w) - \nu_n \nabla.(\nabla w) \\ \frac{1}{R^2} z_j - \nabla.(\frac{1}{R^2}\nabla \Phi) \\ w - \nabla.(\nabla u) \\ R^2[\rho, u] + 2R\rho\partial_Z u + \nabla.(D_{||}\nabla_{||}\rho + D_\perp \nabla_\perp \rho) + S_p(\Psi) \\ R^2[T, u] + 2(\gamma - 1)RT\partial_Z u + \nabla.(K_{||}\nabla_{||} T + K_\perp \nabla_\perp T) + S_h(\Psi) \end{pmatrix}$$

with $\hat{\rho} = R^2\rho$ and $\partial_t A(\mathbf{U}) = \left(\frac{1}{R}\partial_t \Psi,\ R\nabla.(\hat{\rho}\nabla(\partial_t u)),\ 0,\ 0,\ R\partial_t \rho,\ R\partial_t T\right)$.

- Physical and numerical resistivity: η and η_n, viscosity coefficients: ν and ν_n.

Jorek Code: description

Description of the jorek code I

- Jorek: code Fortran 90, parallel (MPI+OpenMP) + algebraic libraries (Pastix, MUMPS ...)
- Initialization
- 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

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

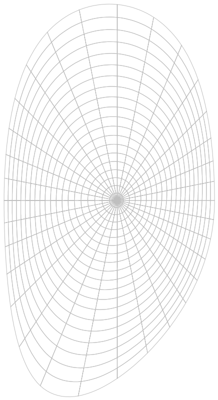


Figure: unaligned grid

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
  - Plot kinetic magnetic energies and restart files.

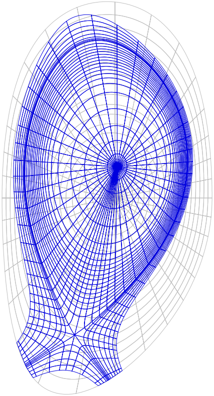


Figure: Aligned grid

Spatial discretization

- The equation in the poloidal plane are discretized using finite element method. For the toroidal direction: Fourier expansion.
- Basis functions: Cubic Bezier elements
  - Generalization of cubic Hermite elements.
  - The generalization allows the local refinement of each element essential for adaptive mesh refinement.
  - 4 degrees of freedom by node to describe a function (9 for Lagrange cubic finite element).
  - With the isoparametric formulation (discretization of (R, Z) using the Bezier elements) the finite elements can be accurately aligned with the equilibrium flux surfaces.
  - The Cubic Bezier elements assure a C^1 polynomial reconstruction.

- *Bezier surfaces and finite elements for MHD simulations*, O. Czarny, G. Huysmans, JCP 2088.

Time scheme in Jorek code

- We recall the model $\partial_t A(\mathbf{U}) = B(\mathbf{U}, t)$
- For time stepping we use a Crank Nicholson or BDF2 scheme :

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

- Defining $G(\mathbf{U}) = (1+\zeta)A(\mathbf{U}) - \theta\Delta t B(\mathbf{U})$ and

$$b(\mathbf{U}^n, \mathbf{U}^{n-1}) = (1+2\zeta)A(\mathbf{U}^n) - \zeta A(\mathbf{U}^{n-1}) + (1-\theta)\Delta t B(\mathbf{U}^n)$$

we obtain the non linear problem

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

- First order linearization

$$\left(\frac{\partial G(\mathbf{U}^n)}{\partial \mathbf{U}^n}\right)\delta\mathbf{U}^n = -G(\mathbf{U}^n) + b(\mathbf{U}^n, \mathbf{U}^{n-1})$$

with $\delta\mathbf{U}^n = \mathbf{U}^{n+1} - \mathbf{U}^n$ and $\frac{\partial G(\mathbf{U}^n)}{\partial \mathbf{U}^n}$ the Jacobian of $G(\mathbf{U}^n)$.

Time scheme in Jorek code

- Linear solver in Jorek:
  - Case 1: Direct solver using Pastix (using when $n_{tor} = 1$)
  - Case 2: Iterative solver
- Iterative Solver step 1: Preconditioning
  - Extraction of submatrices associated to each toroidal harmonics.
  - Factorization of each submatrix
  - We solve exactly (with Pastix) each subsystems.
  - We construct the initial vector of GMRES using the solutions of these systems.
- Iterative solver step 2: GMRES solver for the global matrix.
  - The matrix product vector is preconditioned with the solutions of each subsystems.
- **Principle**: Construction of initial GMRES data + right preconditioning with an approximation of the Jacobian where the coupling between the Fourier modes are neglected.
- In practice for some test cases this coupling is strongly nonlinear.

Jorek code: Non convergence

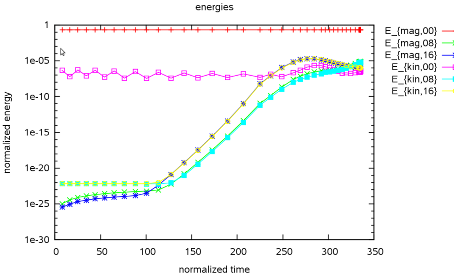
Problem:

- For some test cases the GMRES method does not converge in the nonlinear phase for large time step.

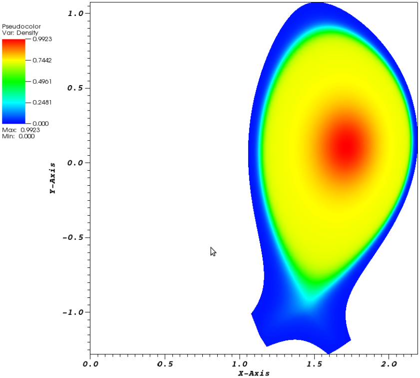
- Why ?
  - The preconditioning is not adapted to obtain a robust GMRES method ?
  - The spatial poloidal and toroidal discretizations is not adapted ?
  - The mesh is not adapted ?
  - The models are not stables ?

Numerical example

- evolution of energy in time

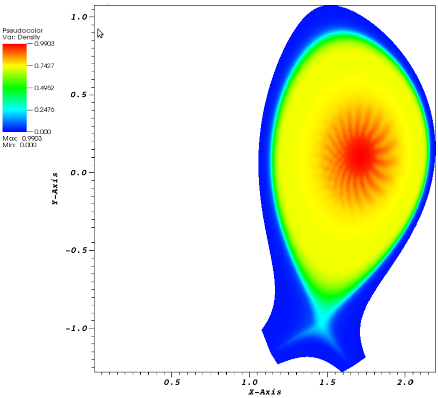


- Density

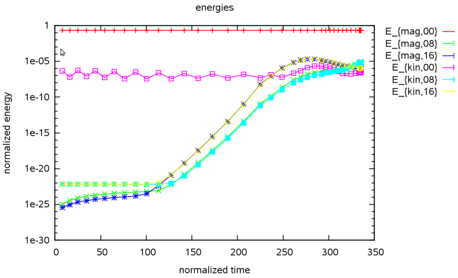


Numerical example

- Density

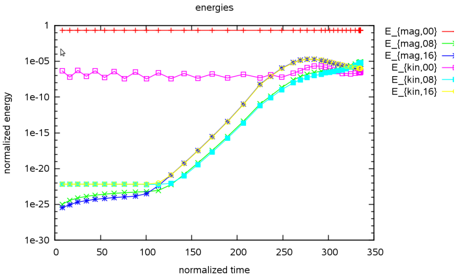


- evolution of energy in time

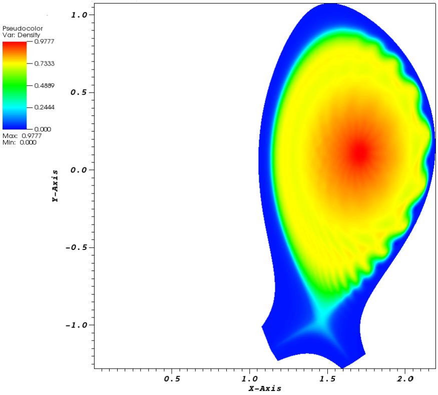


Numerical example

- evolution of energy in time



- Density



Current works on the time discretization

Inexact Newton scheme

- At the time step n, we compute $b(\mathbf{U}^n, \mathbf{U}^{n-1})$, $G(\mathbf{U}^n)$
- We choose $\mathbf{U}_0 = \mathbf{U}^n$ 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

$$\left(\frac{\partial G(\mathbf{U}_k)}{\partial \mathbf{U}_k}\right) \delta \mathbf{U}_k = \tilde{G}(\mathbf{U}_k) = b(\mathbf{U}^n, \mathbf{U}^{n-1}) - G(\mathbf{U}_k)$$

  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 $\mathbf{U}_{k+1} = \mathbf{U}_k + \delta \mathbf{U}_k$.
  - We apply the convergence test (for example $||\tilde{G}(\mathbf{U}_k)|| < \varepsilon_a + \varepsilon_r ||\tilde{G}(\mathbf{U}^n)||$)
- If the newton procedure stop we define $\mathbf{U}^{n+1} = \mathbf{U}_{k+1}$.

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.

- **Aim**: Construct an algorithm which give a good prediction of the solution and which is easy to solve.
  - The algorithm must give a solution of $A\delta\mathbf{U}^n = -G(\mathbf{U}^n) + b(\mathbf{U}^n, \mathbf{U}^{n-1})$ with $A \approx \frac{\partial G(\mathbf{U}^n)}{\partial \mathbf{U}^n}$.
  - A must be well-conditioned. **Idea**: parabolization of the coupled hyperbolic equations.
- Example

$$\left\{ \begin{array}{l} \partial_t u = \partial_x v \\ \partial_t v = \partial_x u \end{array} \right. \longrightarrow \left\{ \begin{array}{l} u^{n+1} = u^n + \Delta t \partial_x v^{n+1} \\ v^{n+1} = v^n + \Delta t \partial_x u^{n+1} \end{array} \right.$$

- 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 matrix.

Preconditioning idea II

- To apply easily this method for more complicated equations, we propose a other interpretation.
- We assume that the matrix associated to the previous linear system is

$$\begin{pmatrix} D_1 & U \\ L & D_2 \end{pmatrix}$$

- Using a Schur decomposition we obtain

$$\begin{pmatrix} D_1 & U \\ L & D_2 \end{pmatrix} = \begin{pmatrix} I & UD_2^{-1} \\ 0 & I \end{pmatrix} \begin{pmatrix} P_{schur} & 0 \\ 0 & D_2 \end{pmatrix} \begin{pmatrix} I & 0 \\ D_2^{-1}L & I \end{pmatrix}$$

$$\begin{pmatrix} I & -\Delta t \partial_x \\ -\Delta t \partial_x & I \end{pmatrix} = \begin{pmatrix} I & -\Delta t \partial_x \\ 0 & I \end{pmatrix} \begin{pmatrix} P_{schur} & 0 \\ 0 & I \end{pmatrix} \begin{pmatrix} I & 0 \\ -\Delta t \partial_x & I \end{pmatrix}$$

- The first and third matrices are triangular and easily to invert.
- $P_{schur} = D_1 - UD_2^{-1}L = (1 - \Delta t^2 \partial_{xx})$ is diagonal dominant matrix.

Preconditioning with Schur decomposition for MHD

- We apply the Schur decomposition to the model 199. The system solved is

$$\frac{\partial G(\mathbf{U}^n)}{\partial \mathbf{U}^n}\delta \mathbf{U}^n = \begin{pmatrix} D_\psi & 0 & D_{\psi,T} & D_{\psi,z_j} & 0 & U_{\psi,u} \\ 0 & D_\rho & 0 & 0 & 0 & U_{\rho,u} \\ 0 & 0 & D_T & 0 & 0 & U_{T,u} \\ D_{z_j,\psi} & 0 & 0 & D_{z_j} & 0 & 0 \\ 0 & 0 & 0 & 0 & D_w & D_{w,u} \\ L_{u,\psi} & L_{u,\rho} & L_{u,T} & L_{u,z} & L_{u,w} & D_u \end{pmatrix} \delta \mathbf{U}^n = \tilde{G}(\mathbf{U}^n)$$

with $\delta \mathbf{U}^n = (\delta\psi, \delta\rho, \delta T, \delta z_j, \delta w, \delta u)$ and $\tilde{G}(\mathbf{U}^n) = -G(\mathbf{U}^n) + b(\mathbf{U}^n, \mathbf{U}^{n-1})$.

- The terms D contains advection and diffusion operators.
- The terms L and U contains non linear coupling hyperbolic operators.
- We reduce the number on variable using the definition of w and z_j.

$$\frac{\partial G(\mathbf{U}^n)}{\partial \mathbf{U}^n}\delta \mathbf{U}^* = \begin{pmatrix} D^*_\psi & 0 & D^*_{\psi,T} & U_{\psi,u} \\ 0 & D_\rho & 0 & U_{\rho,u} \\ 0 & 0 & D_T & U_{T,u} \\ L^*_{u,\psi} & L^*_{u,\rho} & L^*_{u,T} & D^*_u \end{pmatrix} \delta \mathbf{U}^*$$

with $\delta \mathbf{U} = (\delta\psi, \delta\rho, \delta T, \delta u)$

Preconditioning : Algorithm

- The final system with Schur decomposition is given by

$$\delta \mathbf{U}^n = \frac{\partial G(\mathbf{U}^n)}{\partial \mathbf{U}^n}^{-1} \tilde{G}(\mathbf{U}^n) = \begin{pmatrix} M & U \\ L & D_u \end{pmatrix}^{-1} \tilde{G}(\mathbf{U}^n)$$

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

with $P_{schur} = D_u^* - LM^{-1}U$.
- M, D_u^* are associated to the advection and diffusion operators. L, U are associated to the hyperbolic coupling operators.

Final PC-Algorithm

$$\begin{cases} \text{Predictor :} \quad M\delta \mathbf{v}_p^n = (-G_{\mathbf{v}}^n + B_{\mathbf{v}}^n) \\ \text{potential update :} \quad P_{schur}\delta u^n = (-L\delta \mathbf{v}_p^n - G_u^n + B_u^n)) \\ \text{Corrector :} \quad M\delta \mathbf{v}^n = M\delta \mathbf{v}_p^n - U\delta u^n \\ \text{diffusion, update :} \quad D_{z_j}\delta z_j^n = D_{z_j,\psi}\delta\psi^n \quad D_w\delta w^n = D_{w,u}\delta u^n \end{cases}$$

with $\delta v_p = (\delta\Psi, \delta\rho, \delta T)$, $G_{\mathbf{v}}$ and $B_{\mathbf{v}}$ the right hand side associated to the equations on Ψ, ρ and T.

Preconditioning : Approximation of the Schur complement

- The Schur complement $P_{schur} = D_u^* - LM^{-1}U$ necessity to known the matrix M^{-1}.
- Consequently we must approximate P_{schur}. Two approximations:
- Small flow approximation (L. Chacon)
  - In P_{schur} we assume that $M^{-1} \approx \Delta t$
  - **Mathematical problem**: estimate the operator LU.
- Arbitrary flow approximation (L. Chacon).
  - We introduce a operator M_* (in u-space) with $UM_* \approx MU$.
  - Consequently $P_{Schur} = (D_u M_* - LU)M_*^{-1}$ with LU given by the small flow approximation.
  - In this case the Potential udapte step in given by

$$\begin{cases} \text{potential update I :} & (D_u M_* - LU)\delta u^{*,n} = \left(-L\delta \mathbf{v}_p^n - G_u^n + B_u^n)\right) \\ \text{potential update II :} & \delta u^n = M_* \delta u^{*,n} \end{cases}$$

  - **Mathematical problem**: estimate the operator M_*.
- Other choices are possible to approximate the Schur complement.

Preconditioning V: Conclusion

- The PC preconditioning method use a prediction of the solution based on the approximation of the Schur complement.
- It is probable that this prediction of the solution is better than the previous method used in Jorek.
- However it possible that each step of the PC algorithm admit also a problem of conditioning. But since we have a parabolization of the equations and diagonal dominant matrices, add algebraic preconditioning as mutigrid methods can be performing.
- For the step where we solve diffusion and advection operator the previous preconditioning method can be used.

Other way for the future

Extension for others reduced MHD and full MHD

- After the model 199, it will be important to extend the PC-algorithm for the models with parallel velocity and full MHD.
- For the models with parallel velocity the operators U and P_{Schur} are applied on u and $v_{||}$.
- For the full-MHD the operators U and P_{Schur} are applied on the complete velocity field.
- For the full-MHD we have

Helmholtz decomposition

$$\mathbf{v} = R\nabla u \times \mathbf{e}_\phi + Rw\mathbf{e}_\phi + \frac{1}{R^2}\nabla_\perp \chi$$

with u, w, χ scalar fluxes.

- u is associated mainly with the Alfven wave.
- w is associated mainly with the slow wave.
- χ is associated mainly with the fast wave.

Extension for others reduced MHD and full MHD

- In the model 199, the choice of the velocity field show the Alfven wave dominate.
- In the reduced MHD with parallel velocity and the full-MHD, the different types of waves are present.
- The ratio between the different waves can be very important. Consequently the conditioning is impacted by the ratio.
- If this problem impact the efficiency of the PC-algorithm we can use a method proposed by S. Jardin coupled with the previous algorithm.

Jardin method for Schur matrix

- This technic use projection operator to isolate the physics associated with the different wave types in different blocks in the matrix weakly coupled.
- Each submatrix are corrected conditioned.

AP schemes for anisotropic diffusion in Jorek

- Anisotropic diffusion

$$\partial_t \rho - \nabla.(D_{||}\nabla_{||}\rho + D_{\perp}\nabla_{\perp}\rho) = 0 \quad \text{with } D_{\perp}/D_{||} << 1$$

- It is known that the anisotropic diffusion operators are ill-conditioned.
- For instance the big problem of non convergence come from hyperbolic coupling. But it is possible the anisotropic diffusion can be give problem for some test case.
- In the PC-algorithm the anisotropic diffusion operators are contained in the matrix M
- The initial Preconditioning algorithm of Jorek is efficient to treat these terms but the CPU time associated with this algorithm is important.
- We propose:
  - Determinate if the conditioning of M (advection and diffusion terms) is mainly impacted by the anisotropic diffusion.
  - Use AP scheme for these terms to avoid to use a preconditioning and decrease the CPU time.

Anisotropic diffusion in jorek

- Application in the jorek code. Diffusion operator :

$$-\nabla.\left((D_{||}-D_{\perp})\frac{\mathbf{B}\otimes\mathbf{B}}{||\mathbf{B}||^2}\nabla\rho+D_{\perp}\nabla\rho\right)=0$$

  with for example the constants $D_{||}=O(1)$, $D^1_{\perp}=O(\varepsilon)$, $D^2_{\perp}=O(1)$, $D^3_{\perp}=O(1)$ and

$$D_{\perp}=D^1_{\perp}\left(1-D^2_{\perp}+D^2_{\perp}\left(0.5-0.5\tanh(f(\Psi)-D^3_{\perp}\right)\right)$$

- We define $\varepsilon=D^1_{\perp}$, $\nabla_{||}=\frac{\mathbf{B}}{||\mathbf{B}||}.(\frac{\mathbf{B}}{||\mathbf{B}||}.\nabla T)$ to obtain

$$-\nabla.\left(\frac{1}{\varepsilon}A_{||}\nabla_{||}\rho+A_{\perp}\nabla_{\perp}\rho\right)=0$$

  with $A_{||}=\varepsilon D_{||}$ and $A_{\perp}=D_{\perp}=O(\varepsilon)$.
- In this formulation we can apply the AP scheme.

- *Asymptotic-Preserving schemes. Modeling, simulation and mathematical analysis of magnetically confined plasmas*, C. Negulescu.

Stabilization of reduced MHD Models

- Recently B. Després and R. Sart have proposed a more rigorous method to deduce the reduced MHD models (the moment method).
- The authors show that to obtain an energy estimate we must had a term on the poloidal magnetic flux ψ equation.
- For the model 199 the equation ψ come from

$$\partial_t \frac{\Psi}{R} = [\Psi, u] - \epsilon \frac{F_0}{R} \partial_\phi u + \frac{\eta(T)}{R}(z_j - S_c(\Psi)) - \eta_n \nabla.(\nabla z_j) + Q$$

  with $\triangle Q = 0$.
- For the model 199 the stabilization term depends to the boundary conditions.
- For the models with parallel velocity the term Q satisfy $\triangle Q = b(F_0, v_{||}, \Psi)$.
- It will be interesting to add this term in jorek and study the stability of time schemes

- *Derivation of hierarchies of reduced MHD models in Tokamak geometry*, B. Després, Rémy Sart, 2013.

Conclusion

Ongoing and future works

- Determinate the approximations of P_{Schur}
- Finish the code of PC-algorithm for the different approximations of P_{Schur}.

- Analyze the conditioning of the matrix M.
- If the conditioning is impacted by the anisotropic diffusion operators, try to reduced the computational cost using AP schemes.

- Analyze the conditioning of the matrix P_{schur}.
- If this matrix is ill-conditioned, use classical method as ILU method or multigrid method in the "update velocity" step.

- Add the stabilization terms in the reduced MHD models with parallel velocity.
- Adapt the PC-algorithm for the reduced MHD models with parallel velocity.