Advanced Preconditioner for the Nonlinear MHD Code JOREK

I. Holod[1], M. Hoelzl[2], G. Huijsmans[3,4], JOREK Team[5]

[1] *Max Planck Computing and Data Facility, Garching, Germany*
[2] *Max Planck Institute for Plasma Physics, Garching, Germany*
[3] *Eindhoven University of Technology, Eindhoven, The Netherlands*
[4] *CEA, IRFM, Saint-Paul-lez-Durance, France*
[5]*M. Hoelzl, et. al., The JOREK non-linear extended MHD code and applications to large-scale instabilities and their control in magnetically confined fusion plasmas. Submitted to Nuclear Fusion. https://arxiv.org/abs/2011.09120*

Outline

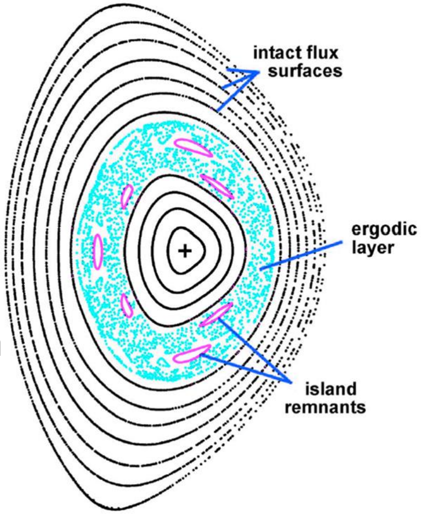
- Introduction
- JOREK solver
- Preconditioning
- Simulation benchmarks
- Summary
- References and Acknowledgments

MHD instabilities

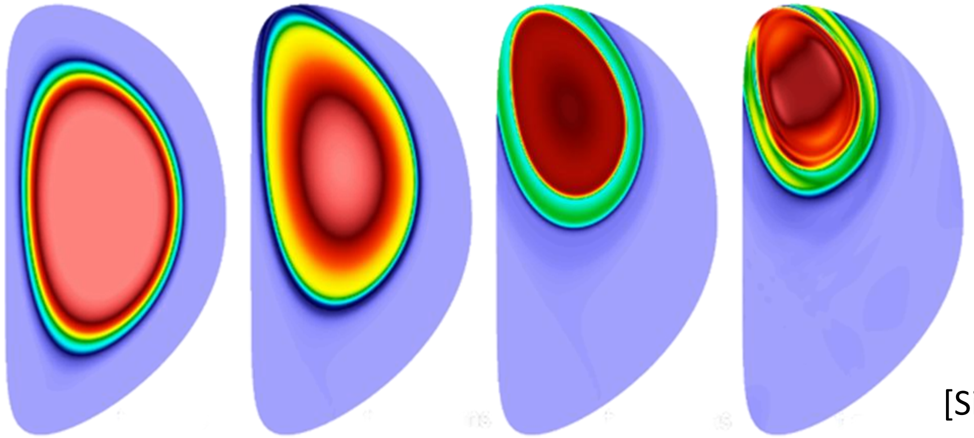
- Control and mitigation of MHD instabilities is crucial for large future tokamaks such as ITER and DEMO.
- Two major classes of MHD events: **Disruptions and Edge-localized modes (ELMs)**
- **Disruptions** cause a sudden loss of the plasma confinement and potentially damage the machine - biggest concern for tokamak.
- **Edge-localized modes** cause high transient heat loads onto material wall structures.
- Numerical simulations play a crucial role in improving physical understanding in order to avoid, control and mitigate MHD instabilities.

Dynamics of a Locked Mode Disruption

- One possible disruption mechanism:
  - Closed magnetic flux surfaces are broken leading to formation of quasistatic magnetic islands – **mode locking**
  - Stochastic field from overlapping islands causing **thermal quench** – a rapid loss of plasma kinetic energy
  - Plasma current rapidly decays in cold plasma – **current quench**
  - Generation of **runaway electrons** by large inductive electric field
  - Plasma drifts vertically into the wall – **vertical displacement event**



- Simulation needed for understanding, avoidance and mitigation

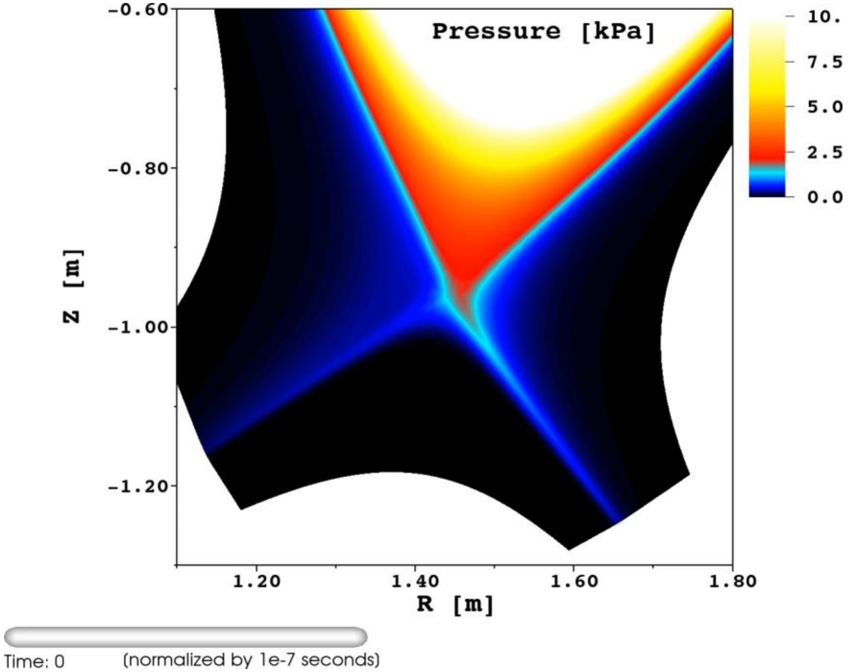


[Simulation by F.J. Artola]

Edge Localized Modes

- ELMs are leading to periodic crashes of plasma pedestal
  - Magnetic reconnection creates an edge stochastic layer causing conductive losses along field lines
  - Development of interchanging filaments leads to convective transport
- Simulation of ELMs provides
  - Estimate of the heat load onto divertor targets
  - Estimate the amount of generated impurities
  - Developing **suppression** & **mitigation** mechanisms

Expulsion of filaments during an ELM crash in ASDEX Upgrade

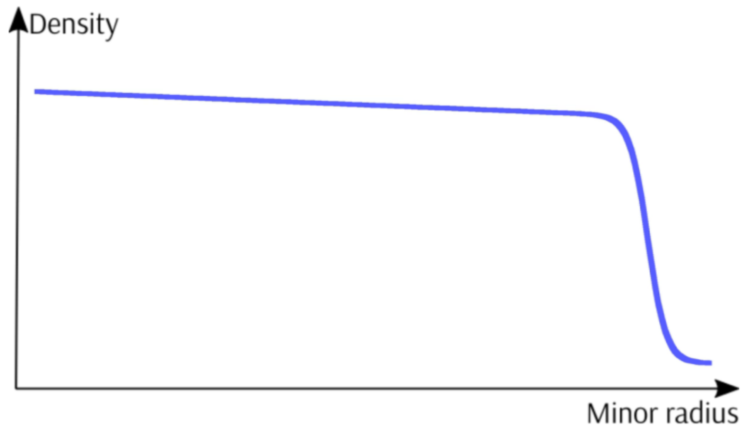


[M Hoelzl et al, PoP 19, 082505 (2012)]

JOREK Code Overview

- Comprehensive simulations need to cover fast MHD events, their onset, non-linear evolution and longer time scale effects.

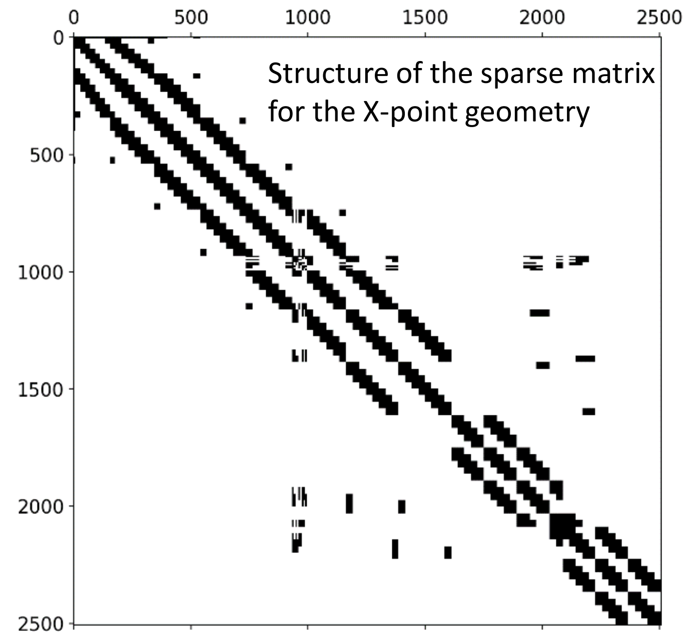
Evolution of H-mode plasma pedestal during the ELM crash



- To address large time scale difference it is essential to use **implicit** time integration method
- JOREK is a massively parallel non-linear MHD code developed for simulations of large-scale instabilities in magnetically confined fusion plasmas
  - Development is coordinated in a proposed EUROfusion TSVV project
  - JOREK has extended worldwide developer/user community

JOREK Solver

- JOREK implements implicit solver for extended MHD equations in 3D realistic geometry including X-point
  - 2D finite element formulation based on Bezier elements (continuous in the values and derivatives, G1)
  - Fourier decomposition in toroidal direction
- Fully implicit time integration requires solving a linear system of equations Ax=b at every time step
  - A is a **sparse matrix**, typically huge and badly conditioned
  - Example: 30K nodes; 8 physical variables; 4 dof per node; 21 toroidal harmonics: matrix dimension 40 million with 500 billion non-zero elements – requires 8 TB of memory for storage

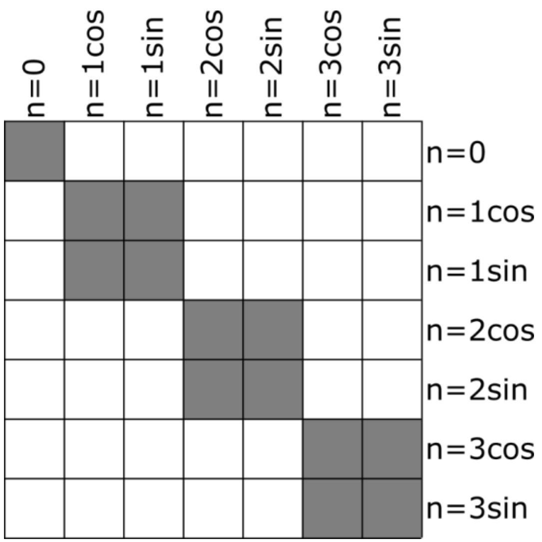


JOREK Solver

- **Direct LU factorization** is (usually) prohibitively expensive.
  - **Iterative GMRES** method with (left) preconditioning is used.
- Preconditioned system to be solved: $P^{-1}Ax=P^{-1}b$
  - Preconditioner matrix P should be easily invertible
  - Product $P^{-1}A$ should have low condition number
    - Ideally $P^{-1}A$ should be close to I
- Solving algorithm:
  - Construct global matrix and RHS – *every time step*
  - Construct/distribute preconditioner matrix – *once per several steps*
  - Analyze/build elimination graph – *once per simulation run*
  - Perform LU factorization – *once per several steps*
  - Perform GMRES iterations – *every step*
    - Find solution for preconditioner matrix – *every iteration*

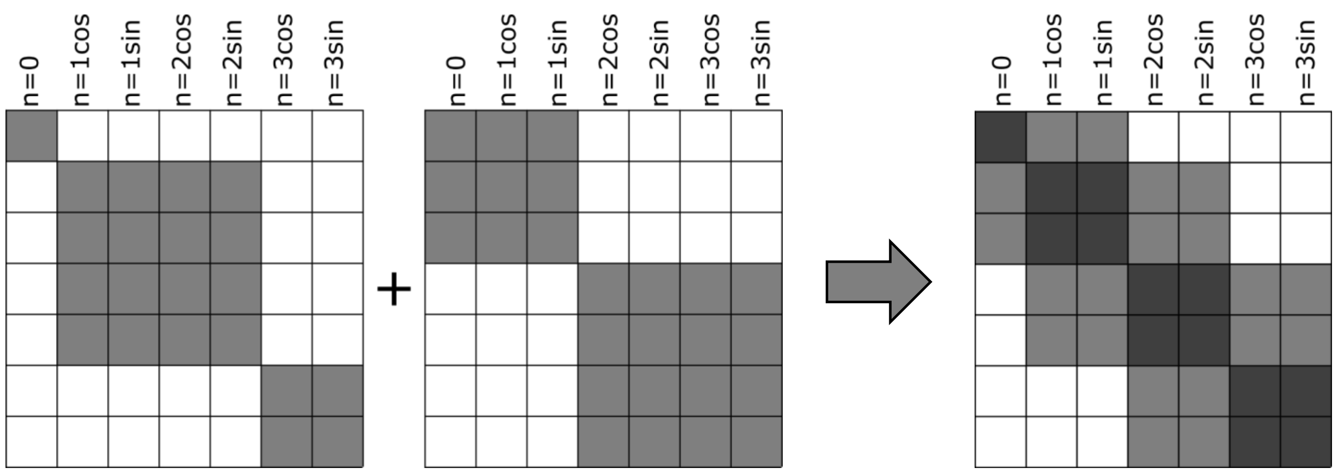
Preconditioning Method I

- In current approach (implemented prior to this work) preconditioner matrix is constructed from the diagonal blocks of individual Fourier harmonics
  - Preconditioner matrix resembles the original matrix A with **omitted mode coupling**
- Each diagonal block can be inverted **independently**
- The full solution is used in GMRES iterations
- The method is fast and scalable with the number of modes
- **In the nonlinear regime where mode coupling is strong, convergence can deteriorate significantly**
- *Can we bring back some of the couplings?*



Preconditioning Method II

- Coupling of **neighboring** modes is taken into account by constructing larger overlapping diagonal blocks (n-1)
- Contribution to solution from each block is taken with a factor ½
- Additional contributions from the first and last modes are calculated
- Total number of diagonal blocks is n + 1
- Performance is limited while solving the largest block

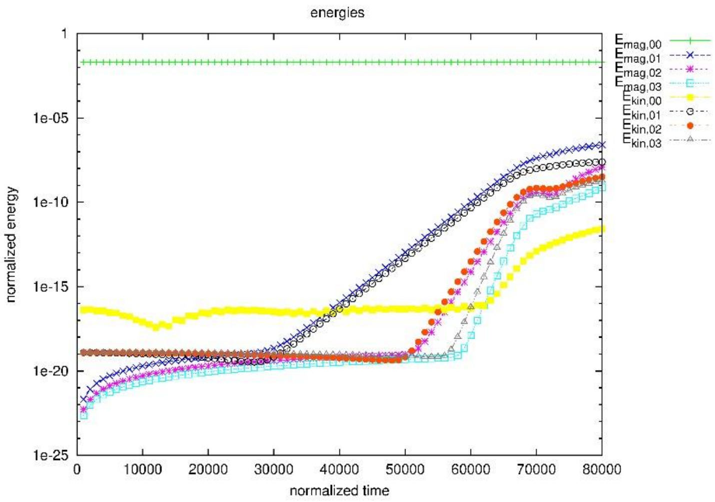


Performance Test in Generic Case

- Simulation of a tearing mode in simplified geometry
  - Number of grid points 2400
  - Global matrix size for n=(0,1,2,3) case: 400,722
  - Global number of non-zeros: 596,198,484, sparsity 0.996
- Number of GMRES iterations for the last 20 simulation steps is reduced significantly with Method II compared to standard Method I

Total number of GMRES iterations

	Method I	Method II
n = (0,…,3)	930	**427**
n = (0,…,6)	1261	**470**
n = (0,…,10)	1419	**483**



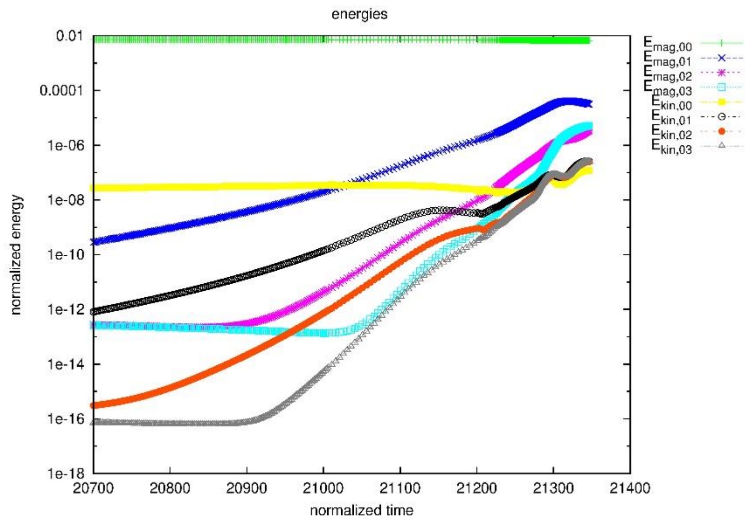
Flexible Workload Distribution

- Ability to arbitrary distribute MPI tasks among mode families is needed for load balancing
- Factorization time scales differently than GMRES time
  - Example: n=(0,1,2,3)
  - 5 mode families: (1,2),(2,3),(0,1),(3),(0)
  - 40 MPI tasks with 6 threads/task
- Preconditioner can usually be reused for many time step
- Minimizing GMRES time is most important

	N = (8,8,8,8,8)	N = (16,16,4,2,2)	N = (18,18,2,1,1)
Factorization (s)	12.1	**8.1**	8.5
GMRES (s)	2.82	**2.67**	3.64

Improved Convergence in VDE Simulation

- Nonlinear simulation of Vertical Displacement Event (VDE) [provided by F.J. Artola]
  - Strong mode coupling leading to poor solver convergence in the nonlinear phase
- Number of iterations reduced by up to a factor of 3 with new preconditioner!



Average number of GMRES iterations per time step

	Method I	Method II
n = (0,…,3)	130	**43**
n = (0,…,10)	102	**48**

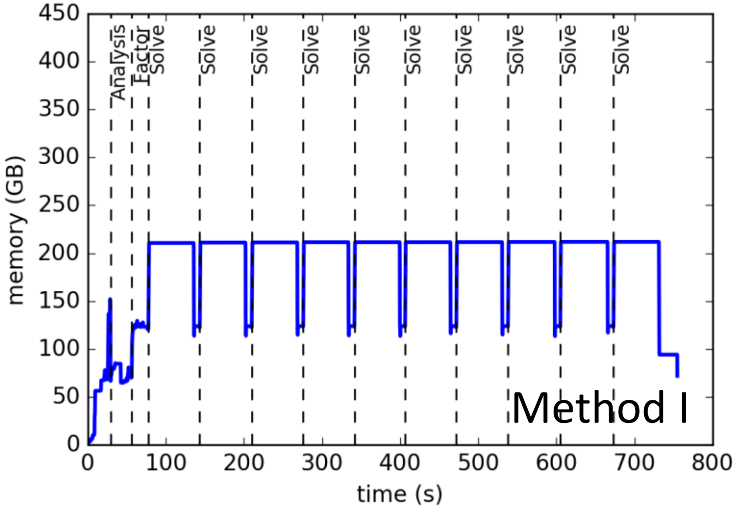
VDE Simulation: Overall Performance Gain

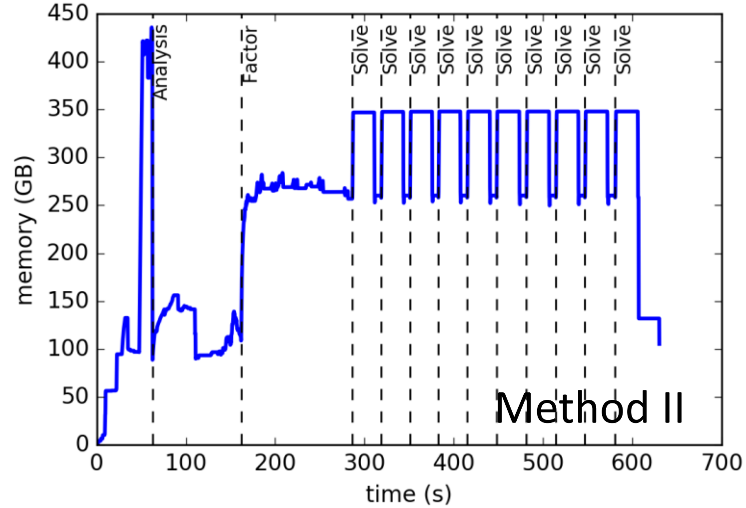
- Improved convergence with new preconditioner allows using longer time step
  - Example shows improvement by factor three in computational costs
  - Further production simulations are ongoing with similar speed-up

	Method I, dt=0.05	Method II, dt=0.15
Factorization, total (s)	21	122x7=857
GMRES, total (s)	5332	4164
Run time (100 steps)	01:37:33	01:44:00

Increased Memory Utilization

- Peak memory utilization with new preconditioner is higher





- Ways to reduce memory usage
  - Optimize matrix construction/conversion
  - Complex representation of preconditioner matrix [P.S. Verma]
  - Compression techniques via Block-Low-Rank (BLR) and Hierarchically Semi-Separable (HSS) matrix representation

Summary

- Large-scale MHD instabilities need detailed understanding for a successful operation of the ITER experiment.
- Due to scale separations, implicit time stepping is used posing a challenging sparse matrix problem.
- New preconditioner based on families of overlapping toroidal harmonics is developed and implemented in the nonlinear MHD code JOREK.
- Flexible workload distribution can mitigate the increased numerical factorization time.
- Solver iterative convergence increases dramatically due to better approximation of a global matrix.
- Overall speedup of a factor of 3 is demonstrated in challenging nonlinear VDE simulations.

References and Acknowledgements

- JOREK non-linear MHD code [https://www.jorek.eu]
- This work will be submitted soon as [I Holod, M Hoelzl, P Singh Verma, G Huijsmans, et al, Journal of Computational Physics]

- The authors would like to acknowledge test cases provided by J Artola and collaborations with P Singh Verma from the EUROfusion HLST team.
- Some of the simulations shown here were performed using the Marconi-Fusion supercomputer.