



irfm



RENAISSANCE
FUSION



M2P2



Advances in 3D ripple tokamak simulations with SolEdge-HDG: isothermal modelling and preconditioner development toward anisothermal simulations.

Marcello CAPASSO, Eric SERRE, Hugo BUFFERAND, Guido CIRAIOLO

TSVV-B annual workshop 2026

October 2, 2026



1. SolEdge-HDG
2. Objectives
3. Previous difficulties
4. WEST ripple simulations
5. Results: $n \nabla T_i T_e$
6. Iterative solvers
7. SolEdge-HDG: p -level multigrid (MG)
8. Conclusions & perspectives



SolEdge-HDG

SolEdge-HDG

What is SolEdge-HDG

SolEdge-HDG is a high-order finite element, mean-field, solving a 2D or 3D conservation equations for single-ion plasma and electrons, derived from **simplified Braginskii closure**. The dynamics are **split** between **perpendicular and parallel** to the magnetic field. **No drifts**.

It solves a set of equations for a vector **conservative values U**:

$$\partial_t \mathbf{U} + \nabla \cdot \mathcal{F}(\mathbf{U}, \nabla \mathbf{U}) = S(\mathbf{U}).$$

SolEdge-HDG can be used to solve **different models**:

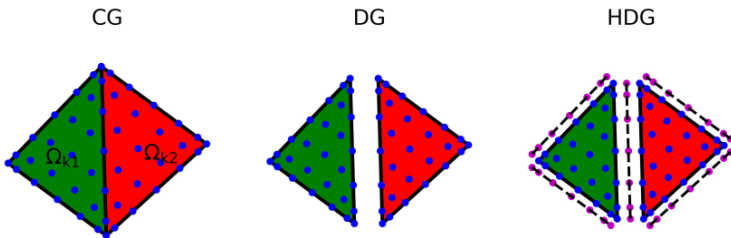
- $n\Gamma$
- $n\Gamma T_i T_e$
- above + **diffusive neutral** equation
- above + $k/k - \epsilon$

In this presentation we'll focus on $n\Gamma$ and $n\Gamma T_i T_e$ model.

SolEdge-HDG

What is HDG

HDG (**H**ybrid **D**iscontinuous **G**alerkin) solves for *traces* (dotted lines, right figures), from which the element values are recovered, reducing the number of DoFs in the linear system.



Discontinuous: the solution on the edge shared by two element **does not need** to be equal.

SolEdge-HDG

2025 Status and 2026 plan

The 2025 status:

- Able to run 3D axisymmetric circular limited simulations ($n\Gamma$) model
- Able to run 3D axisymmetric WEST simulations ($n\Gamma$)

The 2026 plan:

- at least 3D ripple simulation in circular geometry ($n\Gamma$ model)
- WEST 3D ripple simulation and compare with experiments ($n\Gamma$ model)
- investigate iterative solvers
- implement asymptotic preserving (AP) scheme to reduce condition number

I moved to **ripple simulations**:

- 3D circular limited was working fine (n Γ model)
- WEST case had **divergence of the Newton-Raphson** scheme, more resolution? $\rightarrow$ **too many nonzero entries** in the LU decomposition of the linear system $\rightarrow$ out of the **32 bits** limit for the direct solver (MUMPS)

Solution ideas:

- move to **iterative methods and preconditioners** to overcome the 32 bits barrier and reduce RAM consumption.

Iterative methods difficulties:

- classical preconditioners (Gauss-Seidel, BJacobi, ASM, Algebraic Multigrid ...) **fail to converge** or converge very slowly (thousands of iterations)

Solutions:

- implement a more robust preconditioner, like ***p*-level multigrid**.



SolEdge-HDG

The culprit

Is all this resolution for the WEST case actually needed?
Maybe there is a problem with the ripple implementation or configuration?

The was indeed a misconfiguration of the ripple. The coils, instead of being around the vacuum chamber, were partially inside. This caused the magnetic field to be badly defined at certain coordinates, which therefore required more resolution to overcome the problem.

Fixing the misconfiguration allowed to run WEST ripple simulations with a considerable less amount of elements.



WEST ripple simulations

The simulation settings for the $n\Gamma$ are the following:

- $2\pi/9$ toroidal extension (**2 coils**)
- ripple amplitude: up to **16%** wrt to the background magnetic field
- 2700 P4 **poloidal** elements, 4 **toroidal** P3 elements

The simulation settings for the $n\Gamma T_i T_e$ are the following:

- $1\pi/9$ toroidal extension (**1 coil**)
- ripple amplitude: up to **0.74%** wrt to the background magnetic field
- 3700 P4 **poloidal** elements, 3 **toroidal** P3 elements

General workflow:

- run 2D axisymmetric simulation to a prescribed value of perpendicular diffusion ($D_{\perp} = 1 \text{ m}^2\text{s}^{-1}$). **Use the 2D solution as the initial condition for the 3D ripple simulation.**

SolEdge-HDG: mesh generation

The meshes are **automatically optimized** by the implemented *h*-adaptivity procedure.

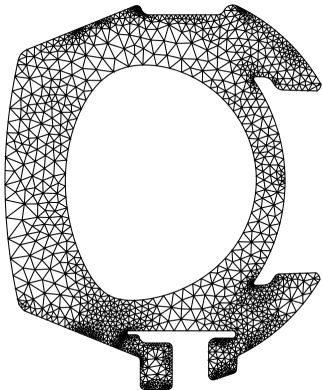


Figure: 2700 elements P4,
isothermal model.

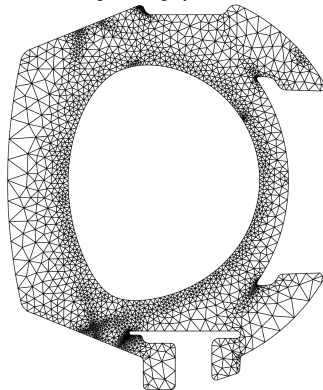


Figure: 3700 elements P4,
anisothermal model.

SolEdge-HDG

Performance

Was the 32 bits limit of the direct solver hit? **No**. The 3D WEST simulation could be run using the direct solver.

Performance $n\Gamma$ model:

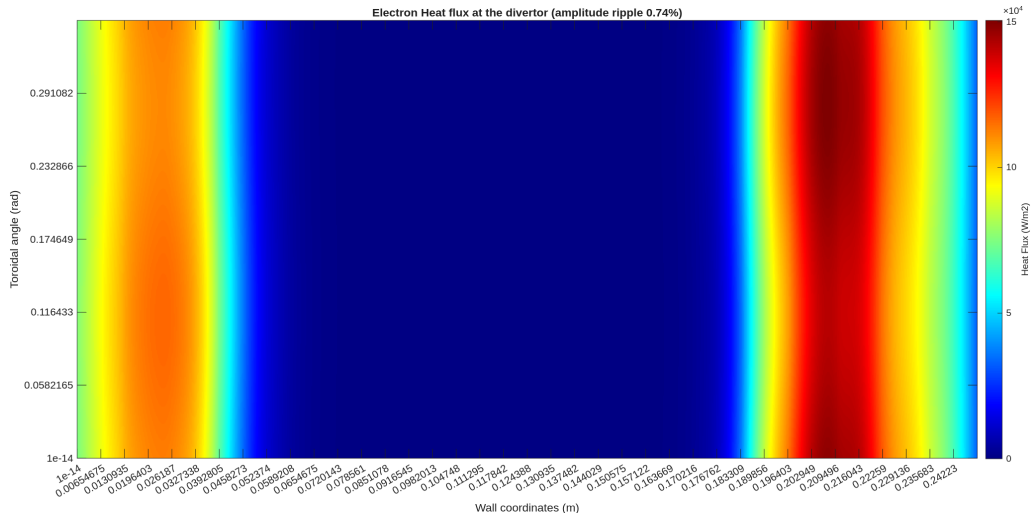
- 5 minutes run-time (starting from converged 2D simulation)
- 20 CPUs, 10 processes
- ≈ 128 GB of RAM memory

Performance $n\Gamma T_i T_e$ model:

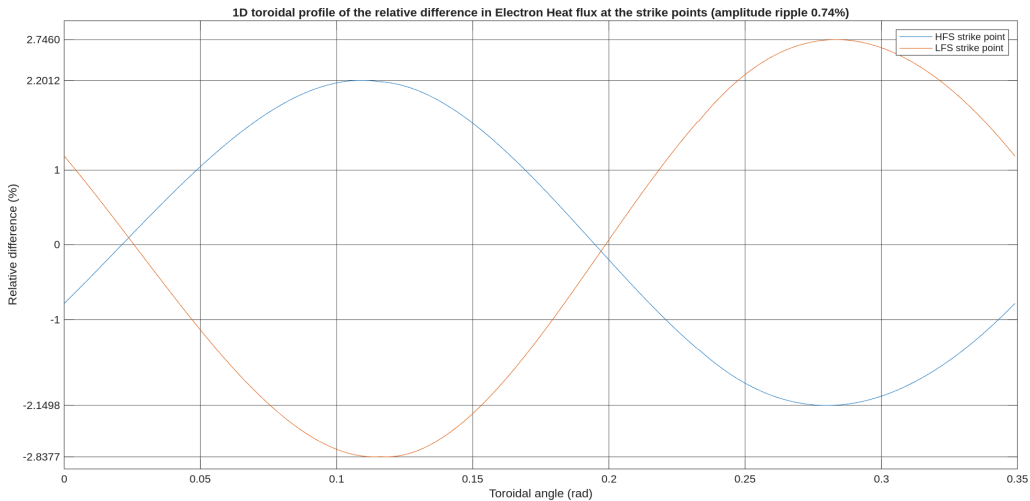
- 30 minutes run-time (starting from converged 2D simulation)
- 256 CPUs, 16 processes
- ≈ 756 GB of RAM memory



Results: $n \Gamma T_i T_e$









Iterative solvers



SolEdge-HDG

Iterative solvers

Since we were able to use the **direct solver**, can we just forget the **iterative ones**? **No** :(

At some point we're gonna hit the 32 bits MUMPS limit if we want to:

- increase computational domain
- lower the prescribed value of perpendicular diffusion
- simulate more complex phenomena
- simulate turbulence

Therefore we still have to think about iterative solvers. However, **for the more complex models and for higher number of degrees of freedom, classical iterative solvers and preconditioners do not converge.**

Iterative solvers

p-level multigrid (MG)

***p*-level multigrid** (MG): solves the problem on the same mesh but at lower polynomial orders and uses this information to "steer" the iterative solver in the right direction:

- each lower-order polynomial problem corresponds to a MG level.
- each lower-order polynomial has associated a matrix A_{coarse} , derived from the original matrix A_{fine} .

Mathematical formulation: **start with the linear system matrix A_{fine} and construct two operators R and P such that:**

$$\begin{aligned}A_{coarse} &= RA_{fine}P \\ A_{fine} &= PA_{coarse}R\end{aligned}$$

R and P are two rectangular matrices, the former is called restriction matrix while the latter is the prolongation one. **They restrict from high order to low and interpolate from low to high.**

Iterative solvers

Performance

Does p -level MG work, is performing well and is it robust wrt nDoF, number processes etc?
Partially.

The iterative method in anisothermal simulations does not converge, even in 2D, for low values of diffusion.

Possible solutions:

- rearrange linear system entries to help the efficiency of the preconditioner
 - rearrange according to **geometrical locality**
 - rearrange according to **strongest coupling** (following **b** for example)
- **reduce anisotropy** → reduce condition number

Iterative solvers

Matrix reordering: geometrical locality 2D

Reverse Cuthill–McKee algorithm (RCM) is applied to the mesh **element** connectivity, but it's the **face** nodes that enter the linear system. The problem is in the external faces.

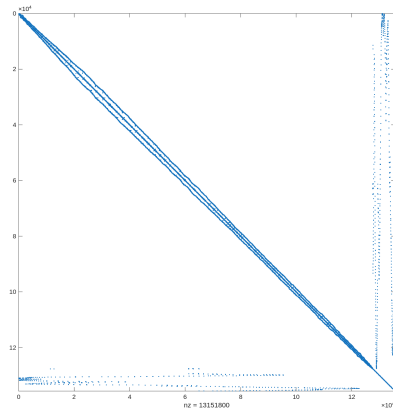
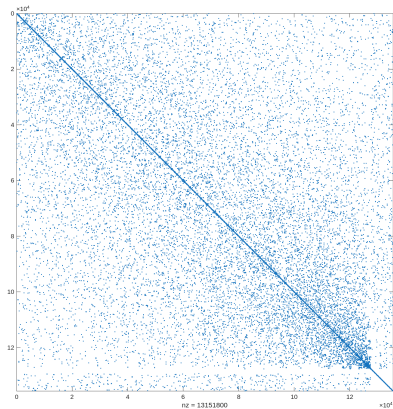


Figure: Original matrix entries pattern.

Figure: Reordered matrix entries pattern.

Iterative solvers

Matrix reordering: geometrical locality 3D

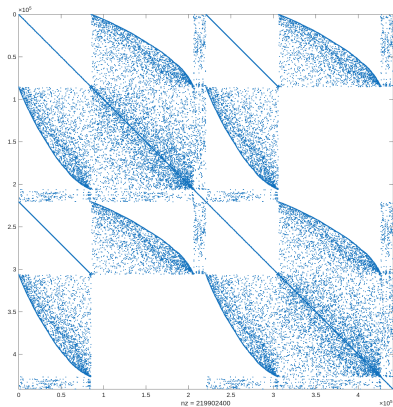


Figure: Original matrix entries pattern.

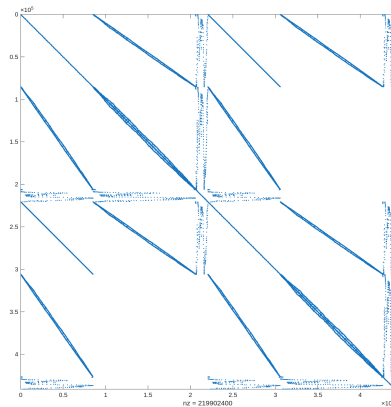


Figure: Reordered matrix entries pattern.



Conclusions & perspectives

Work done:

- $n\Gamma$ 3D WEST ripple simulation, for variable mesh sizes and ripple intensity ✓
- $n\Gamma T_i T_e$ 3D WEST ripple simulation for reasonable mesh sizes and ripple intensity ✓
- implemented p -level preconditioner, 2D and 3D ✓
- comparison with experiments ✗
- implementation of the AP scheme ✗

Perspectives:

- implement matrix reordering based on face locality and/or up-wind coupling
- comparison with experiments and/or SolEdge-3X
- implement AP scheme to reduce condition number due to anisotropy



irfm



RENAISSANCE
FUSION



M2P2



Thank you !

CEA CADARACHE

Cadarache,
Saint-Paul-lès-Durance,
13108 France
marcello.capasso@cea.fr