EUROfusion

EUROfusion ACH Annual Meeting, November 27-28, 2024, Barcelona, Spain

Response matrix compression in the free-boundary and resistive wall extension of JOREK: Current Status and Progresses

F. Cipolletta

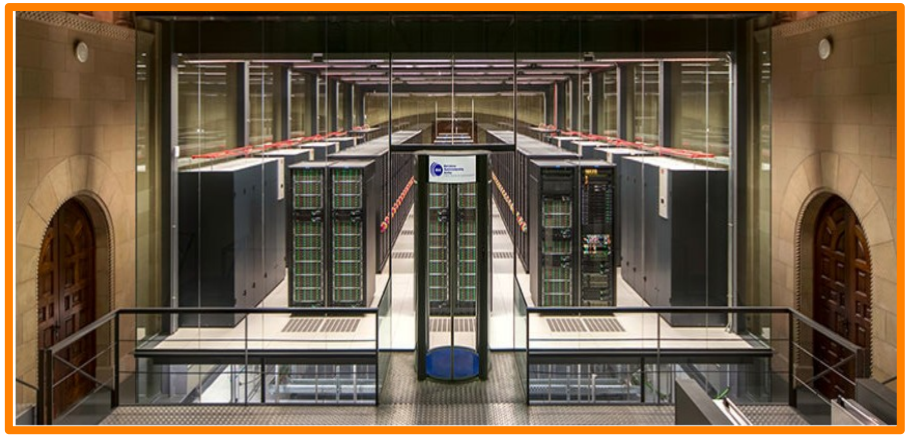
N. Schwarz, M. Hoelzl, S. Ventre, N. Isernia, G. Rubinacci, A. Soba, M. Mantsinen, and The JOREK Team

...

This work has been carried out within the framework of the EUROfusion Consortium, funded by the European Union via the Euratom Research and Training Programme (Grant Agreement No 101052200 — EUROfusion). Views and opinions expressed are however those of the author(s) only and do not necessarily reflect those of the European Union or the European Commission. Neither the European Union nor the European Commission can be held responsible for them.

Context

MareNostrum 4



MareNostrum 5

- This work has been carried on in the framework of the *CIEMAT-BSC Advanced Computing Hub* (ACH)
- All the developments related to this presentation where published on PPCF
- The documentation is available in the JOREK wiki (*requires an account*)

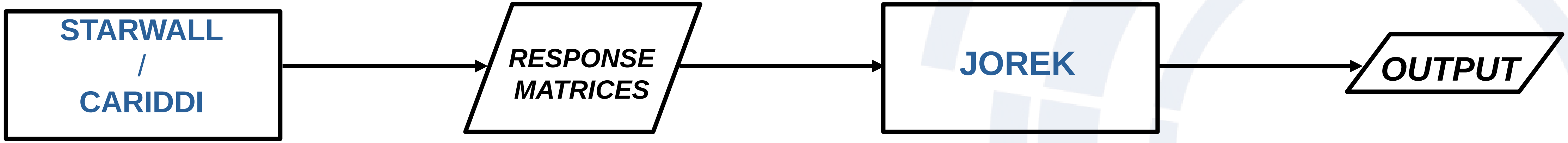
Outline

Free boundary and resistive-wall extension

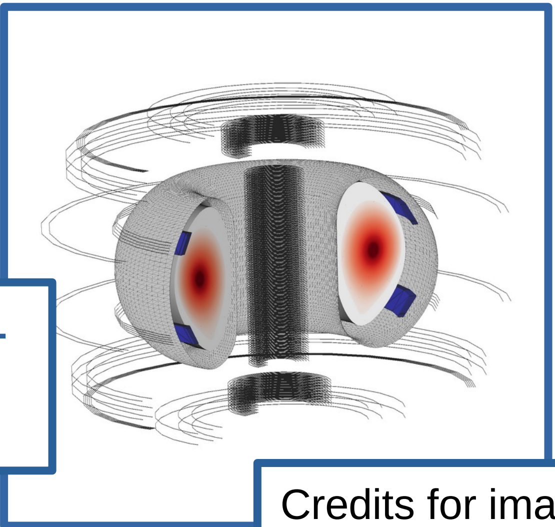
1.1 What is

- JOREK is an HPC code to simulate **MHD instabilities** during Nuclear Fusion
- It adopts 2D *Bezier finite elements* to discretize the poloidal cut and a *Fourier expansion* in the toroidal direction
- In the *free-boundary and resistive wall extension*, the **interactions** between the conducting structures and the plasma are considered to **improve the accuracy of the modeling**. Calculations are done via **STARWALL** or **CARIDDI**

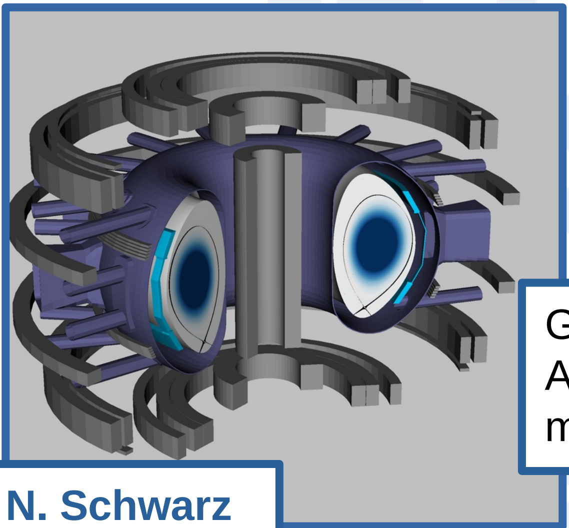
General Flow Chart



Visual Insights



Geometry used in **STARWALL** adopting a 3D thin wall modeling of the response



Geometry used in **CARIDDI** Adopting a 3D volumetric modeling of the response

Credits for images to **N. Schwarz**

1.2 Response matrices and their meaning

All the interactions between the conducting structures and the plasma are provided via the **response matrices**

Dimensionality

bez = *DoF on the boundary elements on the plasma side*
w = *DoF on the wall*

Notes

- Usually **w > bez**
- **w** is fixed by the geometrical model of the device
- **bez** depends of the number n_tor considered in JOREK

Matrix	Rows	Cols	Description
a_ee	bez	bez	describes the interaction of the plasma boundary with itself
s_ww	w	w	describes the wall-wall interaction
s_ww_inv	w	w	the inverse of s_ww
a_ey	bez	w	relates the tangent magnetic field to the JOREK boundary to external currents
a_ye	w	bez	relates the variations of the magnetic flux at the JOREK boundary to ones of the “wall” currents

The response matrices are used in algebraic matrix-vector products inside the JOREK code

Matrix compression

2.1 Objectives and the method adopted (1)

Goal

Enable modeling capabilities of realistic and accurate 3D wall structures within MHD simulations of plasma instabilities inside a Tokamak

But

Modeling the walls and the conductive structures accurately can be very expensive in terms of memory

Objective

Reduce Memory required by response matrices

How?

Apply factorization and compression techniques to matrices provided by **STARWALL** or **CARIDDI** in the free-boundary and resistive wall extension of JOREK

2.1 Objectives and the method adopted (2)

We choose the Singular Value Decomposition (**SVD**) method:

- k = rank($\boldsymbol{\Sigma}$) $\leq$ min(m,n)
- **A** dense
- ***U***, ***V*** orthogonal
- $\boldsymbol{\Sigma}$ diagonal

$$\boldsymbol{A}(m,n) = \boldsymbol{U}(m,k) \times \boldsymbol{\Sigma}(k,k) \times \boldsymbol{V}^T(k,n)$$

$$\begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & & \vdots \\ a_{(m-1)1} & a_{(m-1)2} & \cdots & a_{(m-1)n} \\ a_{m1} & a_{m2} & \cdots & a_{mn} \end{bmatrix} = \begin{bmatrix} u_{11} & \cdots & u_{1k} \\ u_{21} & \cdots & u_{2k} \\ \vdots & & \vdots \\ u_{(m-1)1} & \cdots & u_{(m-1)k} \\ u_{m1} & \cdots & u_{mk} \end{bmatrix} \times \begin{bmatrix} \sigma_{11} & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & \sigma_{kk} \end{bmatrix} \times \begin{bmatrix} v_{11} & v_{12} & \cdots & v_{1n} \\ \vdots & \vdots & & \vdots \\ v_{k1} & v_{k2} & \cdots & v_{kn} \end{bmatrix}$$

Required Dimensions for the factorized representation:

storing (**UΣ**) and **V**^{T} instead of **A** => *(mk+nk)* elements instead of *(mn)* (size scales linearly with k)

Powerful Features:

1) The SVD can be always performed
2) An SVD with singular values in descending order always exists
3) The SVD is an optimal approximation with respect to the residual computed via Frobenius norm

Implementation:

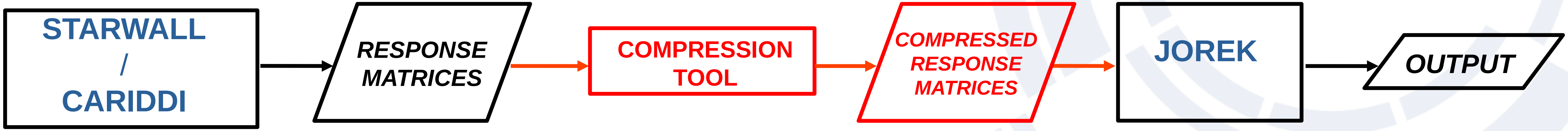
The Scalable Linear Algebra PACKage (ScaLAPACK) parallelized library offers the routine pdgesvd to compute the SVD of a given matrix (https://netlib.org/scalapack/)

2.2 Notes on the current implementation

$$\mathbf{A}(m,n) = \mathbf{U}(m,k) \times \mathbf{\Sigma}(k,k) \times \mathbf{V}^T(k,n)$$

$$\begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & & \vdots \\ a_{(m-1)1} & a_{(m-1)2} & \cdots & a_{(m-1)n} \\ a_{m1} & a_{m2} & \cdots & a_{mn} \end{bmatrix} = \begin{bmatrix} u_{11} & \cdots & u_{1k} \\ u_{21} & \cdots & u_{2k} \\ \vdots & & \vdots \\ u_{(m-1)1} & \cdots & u_{(m-1)k} \\ u_{m1} & \cdots & u_{mk} \end{bmatrix} \times \begin{bmatrix} \sigma_{11} & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & \sigma_{kk} \end{bmatrix} \times \begin{bmatrix} v_{11} & v_{12} & \cdots & v_{1n} \\ \vdots & \vdots & & \vdots \\ v_{k1} & v_{k2} & \cdots & v_{kn} \end{bmatrix}$$

- ScaLAPACK's pdgesdv returns $k = \min(m,n)$, with singular values in descending order
- **Compressing** means dropping the smallest singular values, to obtain ***k < min(m,n)***
- One obtains **computational savings** if $k < \frac{m * n}{m+n}$, otherwise the factorization might bring additional computation costs
- The response matrices do not change during evolution → *COMPRESS one time only*

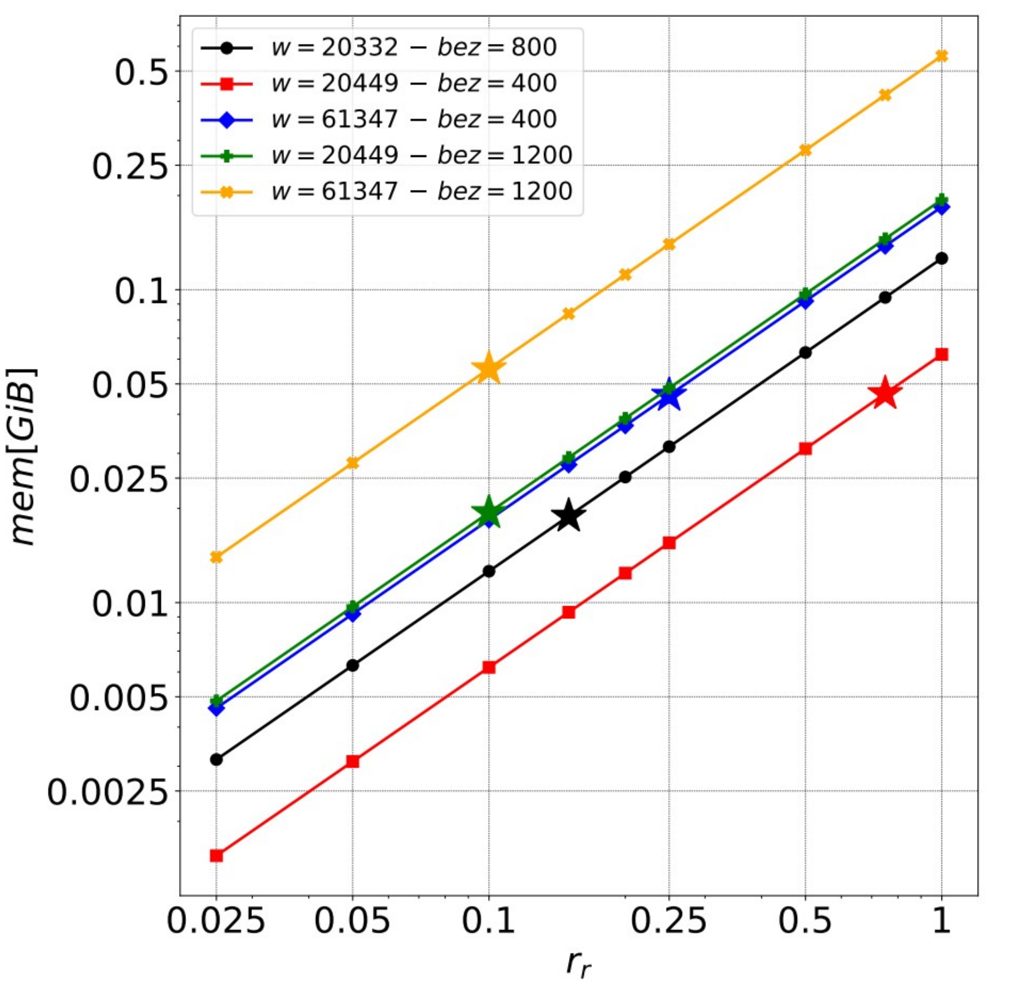


- The **compress_response** program was developed
- JOREK needed adaptation → started from matrices $\mathbf{a}_{ey}$ and $\mathbf{a}_{ye}$ (**plasma-wall interactions**)

Results of tests

3.1 The Tearing Mode (TM) instability

Scan in the resolution, varying the DoF used in **STARWALL**



r_r = rate of retained singular values → smaller = more compression
w = dof walls
bez = dof Bezier elems

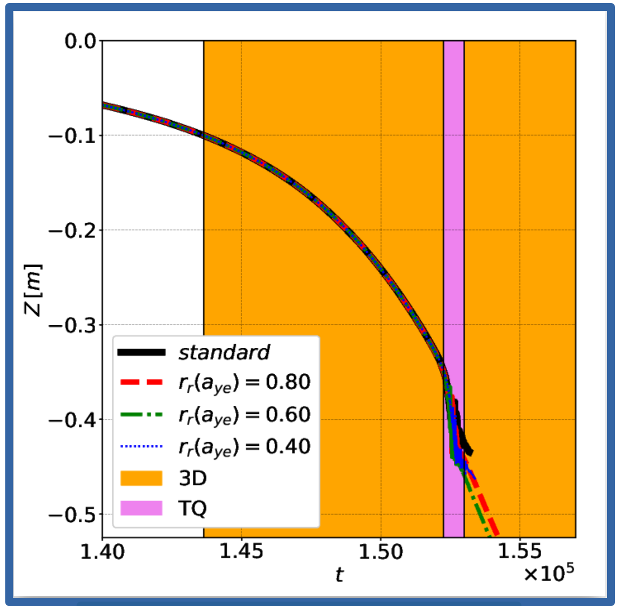
NOTE: ★ indicates the smallest r_r providing accurate results, i.e. the more it is towards the left the more the compression is efficient

- First case if the **black** curve (shows good compression)
- Started a sistematic study on resolution:
  - The **red** curve is taken as basis
  - Tripling w in the **blue** curve
  - Tripling bez in the **green** curve
  - Tripling both w and bez in the **orange** curve

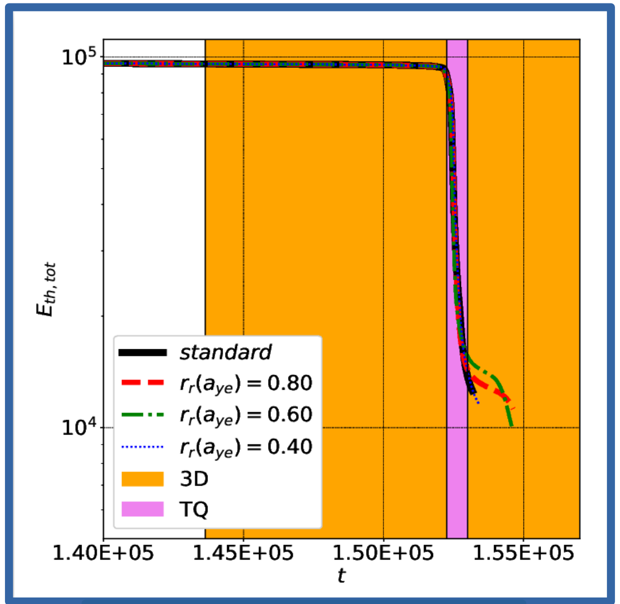
Compression is more efficient on large matrices, in particular for larger resolutions on the plasma side

3.2 Vertical Displacement Event (VDE)

$r_r(a_{ye})$ = **0.8**, **0.60**, **0.40**; $r_r(a_{ey})$ = 1.0; **standard** means non-factorized and uncompressed

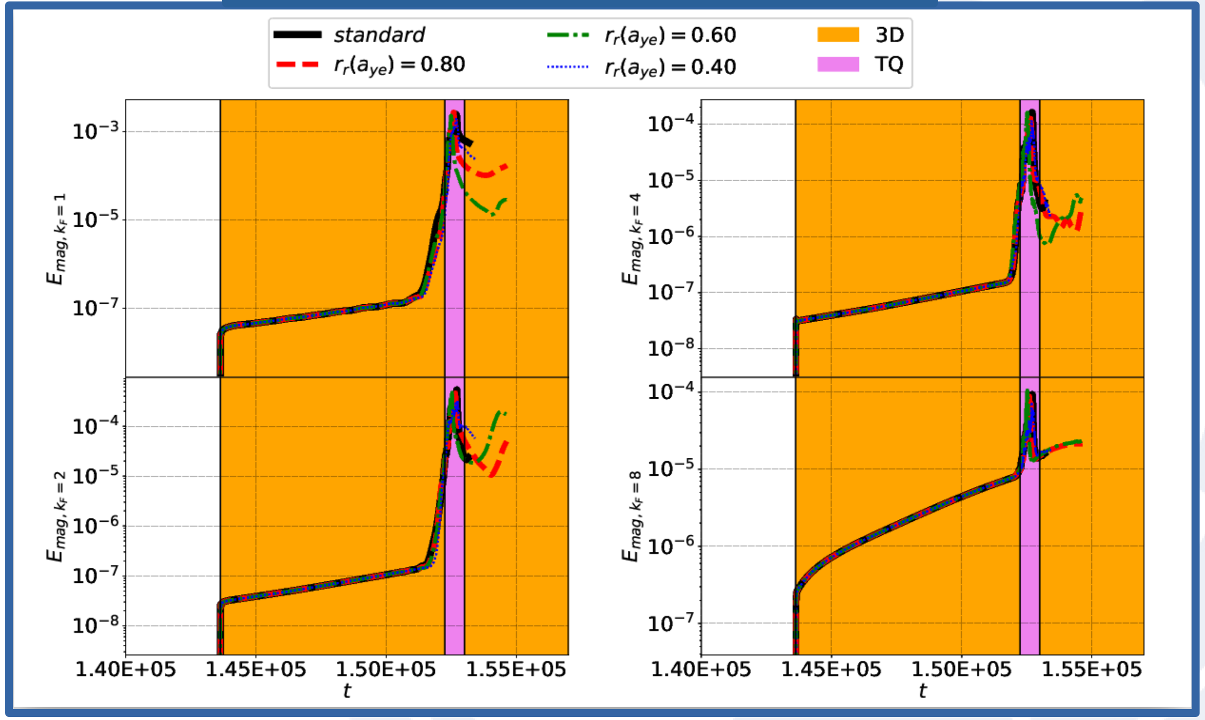


Magnetic Z axis



Thermal Energy

Magnetic Energy harmonics



Conclusion: the results are *accurate* only *up to the Thermal Quench* (**TQ**) phase

3.3 Limitations

1

Inefficiency in **TM** wrt **w dof** is due to definition of SVD and depicted geometry:

$$A(m,n) = U(m,k) \times \Sigma(k,k) \times V^T(k,n)$$

with

$$k \leq min(m,n)$$

where:

min(m,n) “rules” *the amount of compression* and **bez** < **w** in the cases studied

The method requires more development in order to compress the w dof

2

Inaccurate results in **VDE** when compressing both $\mathbf{a}_{ey}$ and $\mathbf{a}_{ye}$

and

Inefficiency in **VDE** to deal with high-non-linearity, even when compressing only $\mathbf{a}_{ye}$

The method requires further development to deal with VDE scenarios

Conclusions and outlook

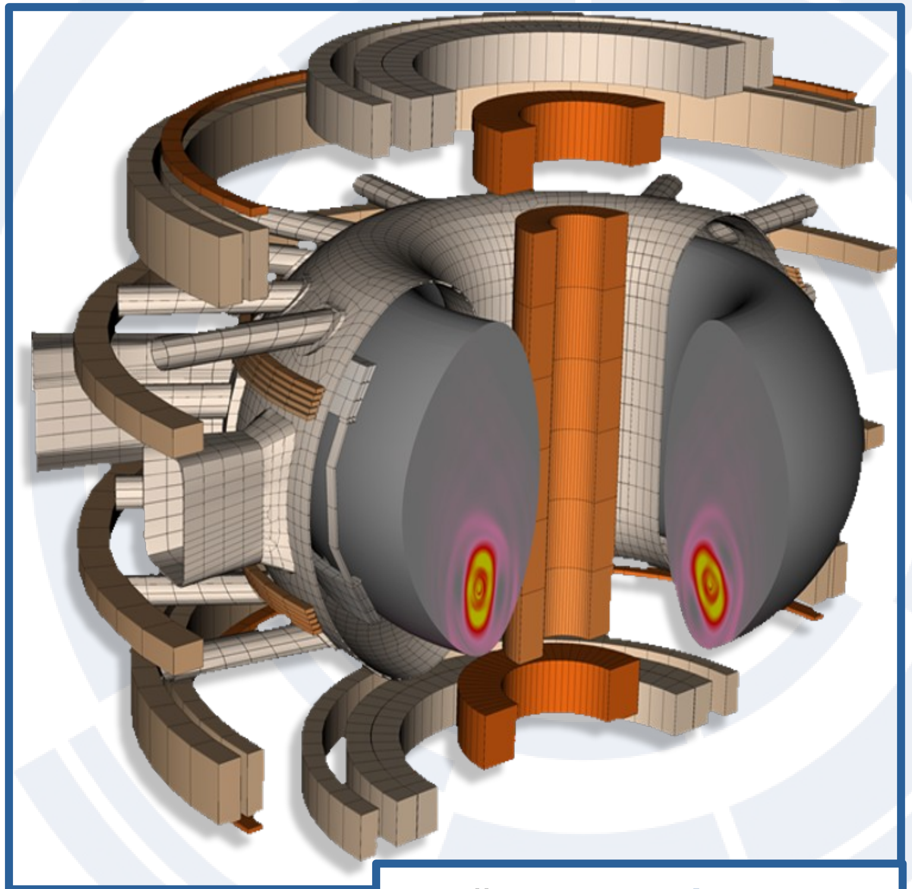
4 Conclusions and outlook

Summary

- First implementation of a *tool for the compression of response matrices* have been provided, along with its validation and assessment
- The *TM test case* could be compressed accurately
- The *VDE test case* requires further study and development
- *Seminar on the new development* held for JOREK users
- The *JOREK documentation* has been updated
- The *pull request* is done and the code is ready to be merged

Future (project for 2025)

- Maintain the pull request alive merging to the master
- "Managing" the size of matrices produced via CARIDDI (*complementary to matrix compression*):
  1. Investigate possible optimization at computation of the response matrices (e. g. using **ELPA** to *solve the eigenvalue problem*)
  2. Assess the applicability of Model Order Reduction (**MOR**) and alternative techniques with respect to the *solution of the eigenvalue problem*



credits to **N. Schwarz**

Thank you for the attention

Questions?