YACS - A NEW 2.5D FEM EIGENMODE SOLVER FOR AXISYMMETRIC RF-STRUCTURES *

B.D. Isbarn†, B. Riemann, M. Sommer, T. Weis
TU Dortmund University (DELTA), Center for Synchrotron Radiation, Dortmund, Germany

Abstract

Most feasibility studies for modern accelerator concepts, including superconducting multicell RF-cavity-resonators in circular accelerators, depend on computing a large number of eigenmode frequencies and field patterns to obtain typical figures of merit. This task includes computationally intensive studies. To obtain the full eigenfrequency spectra most of these studies are performed in 3D, require a great amount of computation resources and thus are limited to a few hundred or thousand eigenmodes. To overcome this issue, some codes make use of the axisymmetric geometry of most of the RF-cavity-resonator structures and solve the problem in 2D. Solving in 2D however reduces the eigenmode spectra to eigenmodes with no azimuthal dependencies (so called monopole modes). Due to the lack of freely available and easy to use 2.5D eigenmode solvers which are able to solve for the full 3D field in a reduced 2.5 dimensional problem, we developed yet another cavity solver (Yacs), a simple FEM based solver capable of solving for the full 3D eigenmodes of axisymmetric problems while only requiring a fraction of the computation resources required by most modern 3D codes.

INTRODUCTION

Numerical calculation of resonant eigenmodes with the eigenfrequency $\omega \in \mathbb{C} \setminus \{0\}$ in a rf-structure usualy involves solving the vector Helmholtz equation for the electric field that arises from the Maxwell equations in a bounded domain $\Omega \subset \mathbb{R}^3$

$$\nabla \times \left(\underline{\mu}^{-1} \nabla \times \boldsymbol{E}\right) - \omega^2 \underline{\epsilon} \boldsymbol{E} = \boldsymbol{0} \, . \tag{1}$$

Closed-form solutions of (1) can only be obtained for simple geometries like cylindrical or spherical resonators.

The Sparse Eigenvalue Problem

In the case of an axisymmetric problem domain we may use spherical coordinates and expand the azimuthal component of the electric field by a Fourier series

$$\boldsymbol{E}(\boldsymbol{r}) = \sum_{m=0}^{\infty} \boldsymbol{E}_m^{(c)}(r,z)\cos(m\theta) + \boldsymbol{E}_m^{(s)}(r,z)\sin(m\theta) \, . \tag{2}$$

Due to the pairwise orthogonality of the Fourier basis functions we receive a decoupled problem and can solve for each mulipole mode m individually. Applying the finite element method on the problem stated in (1) together with the azimuthal fourier series expansion (2) we can approximate (1) with a sparse generalized eigenvalue system. This has allready been discussed thoroughly in earlier works [1]. The resulting sparse generalized eigenvalue system is described by

$$\begin{pmatrix} \underline{\boldsymbol{K}}^{pp} & \underline{\boldsymbol{K}}^{p\theta} \\ \underline{\boldsymbol{K}}^{p\theta^T} & \underline{\boldsymbol{K}}^{\theta\theta} \end{pmatrix} \begin{pmatrix} \hat{\boldsymbol{E}}_p \\ \hat{\boldsymbol{E}}_\theta \end{pmatrix} = \omega^2 \begin{pmatrix} \underline{\boldsymbol{M}}^{pp} & 0 \\ 0 & \underline{\boldsymbol{M}}^{\theta\theta} \end{pmatrix} \begin{pmatrix} \hat{\boldsymbol{E}}_p \\ \hat{\boldsymbol{E}}_\theta \end{pmatrix} \tag{3}$$

$$\underline{\boldsymbol{K}} \hat{\boldsymbol{E}} = \omega^2 \underline{\boldsymbol{M}} \hat{\boldsymbol{E}} \tag{4}$$

with

$$\begin{aligned}
\underline{\boldsymbol{K}}_{ij}^{pp} &= \langle \mu_\theta^{-1} r \nabla_p \times \boldsymbol{\phi}_i, \nabla_p \times \boldsymbol{\phi}_j \rangle + m^2 \langle \mu_p^{-1} r^{-1} \boldsymbol{\phi}_i, \boldsymbol{\phi}_j \rangle \\
\underline{\boldsymbol{K}}_{ij}^{p\theta} &= m \langle \mu_p^{-1} r^{-1} \boldsymbol{\phi}_i, \boldsymbol{\nabla}_p \psi_j \rangle \\
\underline{\boldsymbol{K}}_{ij}^{\theta\theta} &= \langle \mu_p^{-1} r^{-1} \boldsymbol{\nabla}_p \psi_i, \boldsymbol{\nabla}_p \psi_j \rangle \\
\underline{\boldsymbol{M}}_{ij}^{pp} &= \langle \epsilon_p r \boldsymbol{\phi}_i, \boldsymbol{\phi}_j \rangle \\
\underline{\boldsymbol{M}}_{ij}^{\theta\theta} &= \langle \epsilon_\theta r^{-1} \psi_i, \psi_j \rangle
\end{aligned}$$

where $\underline{\boldsymbol{K}}$ and $\underline{\boldsymbol{M}}$ are typically referred to as stiffness- resp. mass-matrices, p (in-plane) denotes operations and vectors with respect to the $\hat{\boldsymbol{r}}$ and $\hat{\boldsymbol{z}}$ direction, while $\boldsymbol{\phi}_x$ together with ψ_x refer to the global mapping of the basis functions with $\boldsymbol{\phi}_x$ referring to the vector-valued in-plane component and ψ_x referring to the out of plane scalar component.

IMPLEMENTATION

The primary part of the FEM code has been developed in C++ using state-of-the-art numerical libraries to achieve maximum performance and reliability. The meshing of the problem domain is performed with Triangle [2], general dense and sparse algebra is done using Eigen v3 [3] and solving the sparse generalized eigenvalue problem has been realized with ARPACK [4] in conjunction with UMFPACK [5]. The latter is used for solving sparse linear systems. In this first iteration of Yacs we solely use first order basis functions for expanding the trial and test functions.

Boundary Conditions

In the case of solving multipole modes, we artifically introduce PEC boundary conditions on the rotation axis and thus force the electrical field parallel to the rotation axis to vanish. In order to avoid the singular terms for $r = 0$ in (3), we use a Gauss quadrature scheme that only evaluates points inside the domain and avoids those points on the boundary of the domain.

BENCHMARK

All the benchmarks presented in the following were performed on a simple $\nu = 500\,\text{MHz}$ pillbox cavity for which

* Work supported by the BMBF under contract no. 05K13PEB
† benjamin.isbarn@tu-dortmund.de

ISBN 978-3-95450-168-7

closed-form solutions of (1) are well known [6]. For comparison we also include benchmarking data obtained with COMSOL Multiphysics 5.0 [7] using the Matlab [8] LiveLink™ interface. COMSOL uses second order basis functions for expanding the function space of the test and trial functions.

Monopole Performance

Solving for monopole modes ($m = 0$) can be performed on the reduced 2D problem. In this case, all terms involving the evaluation of m in (3) vanish, resulting in block diagonal stiffness- and mass-matrices and thus increasing the computational performance significantly. COMSOL also supports solving 2D axisymmetric problems. The relative frequency deviations with reference to the analytical solution of the first 6 monopole modes as a function of the number of elements and the time required by Yacs and COMSOL, are displayed in the Figs. 1 and 2. COMSOL has a better convergence behavior which is attributable to the second order polynomial function space basis. Nonetheless Yacs is able to outperform COMSOL when considering the solution time required, down to relative frequency deviations of 1×10^{-5} depending on the mode, which can be beneficial for high-dimensioned optimization problems.

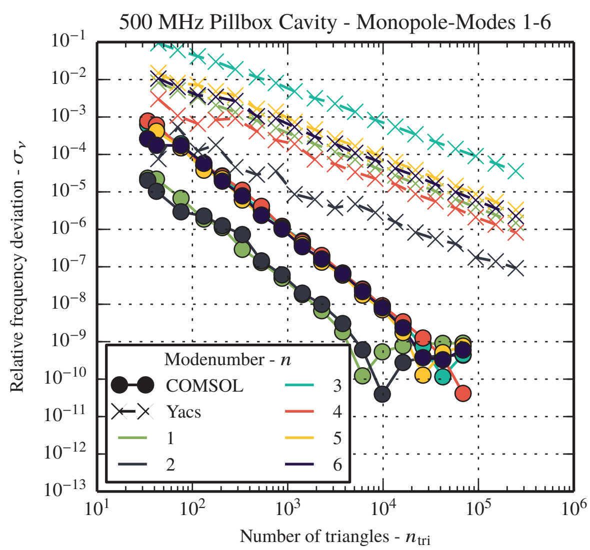

Figure 1: Relative frequency deviation of the first 6 monopole modes obtained from Yacs and COMSOL with respect to the analytical solution of a 500 MHz pillbox-cavity as a function of the number of finite elements in the problem domain.

Multipole Performance

Solving the 2.5D problem ($m \neq 0$) can in principle be performed with COMSOL. Unfortunately, this solver currently only supports coaxial problems so that we were only able to solve the problems in 3D with COMSOL. A comparison of the first 50 dipole respectively quadrupole modes numerically obtained with Yacs, and the analytical solution,

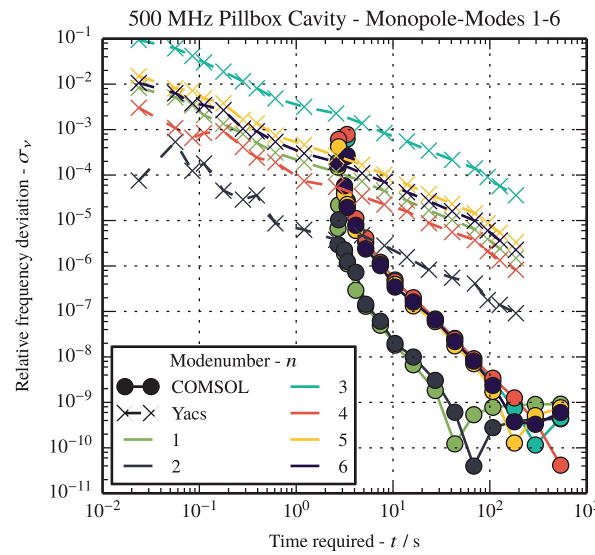

Figure 2: Relative frequency deviation of the first 6 monopole modes obtained from Yacs and COMSOL with respect to the analytical solution of a 500 MHz pillbox-cavity as a function of the solution time required.

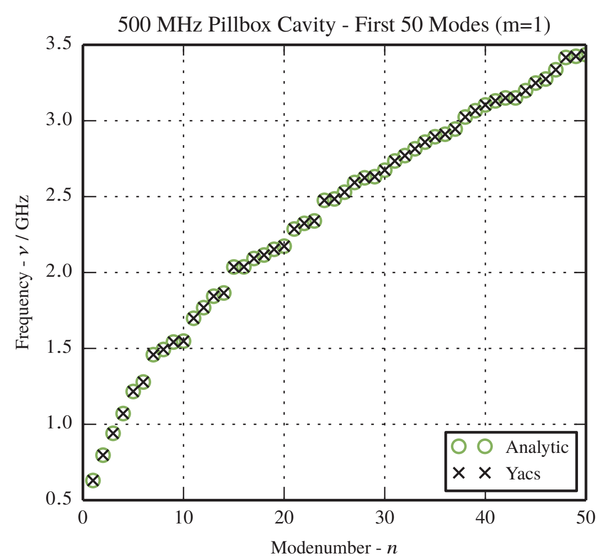

Figure 3: Comparison of analytical and numerical solutions for the first 50 dipole modes ($m = 1$) obtained with Yacs utilizing $n_{\mathrm{elem}} = 246470$ mesh cells.

is displayed in the Figs. 3 and 4. All eigenfrequencies were obtained with a single solver run. The eigenfrequencies obtained with Yacs are in good agreement with the analytical solutions and contain no spurious modes for $\omega > 0\,\mathrm{Hz}$. The relative frequency deviations with reference to the analytical solution of the first 4 multipole modes as a function of the number of elements and the time required by Yacs and COMSOL, are displayed in the Figs. 5 and 6. COMSOL still has a better convergence behavior due to the second order polynomial basis but sacrifices performance for higher

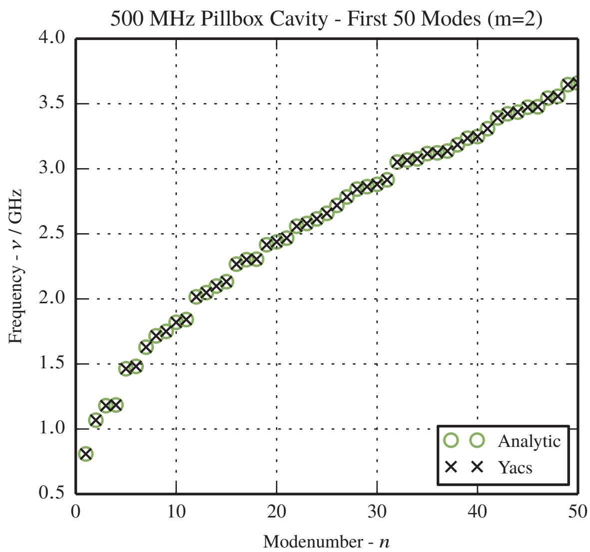

Figure 4: Comparison of analytical and numerical solutions for the first 50 quadrupole modes ($m = 2$) obtained with Yacs utilizing $n_{\mathrm{elem}} = 246470$ mesh cells.

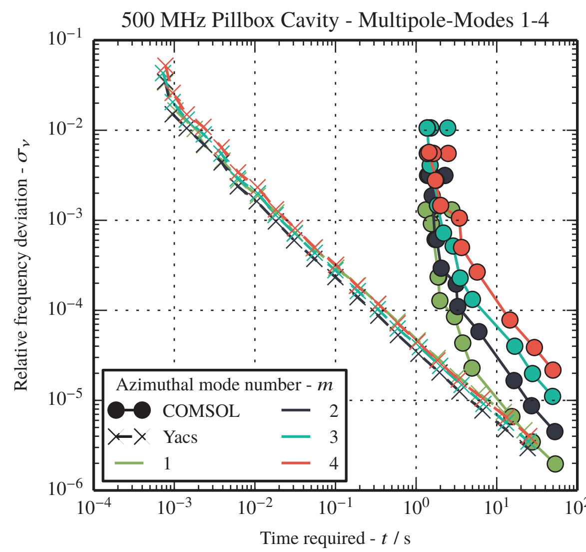

Figure 6: Relative frequency deviation of the first 4 multipole modes obtained from Yacs and COMSOL with respect to the analytical solution of a 500 MHz pillbox-cavity as a function of the solution time required.

multipole orders. This holds especially true when observing the solution times where Yacs manages to outperform COMSOL since the convergence behavior of Yacs is almost constant as a function of the azimuthal mode number m.

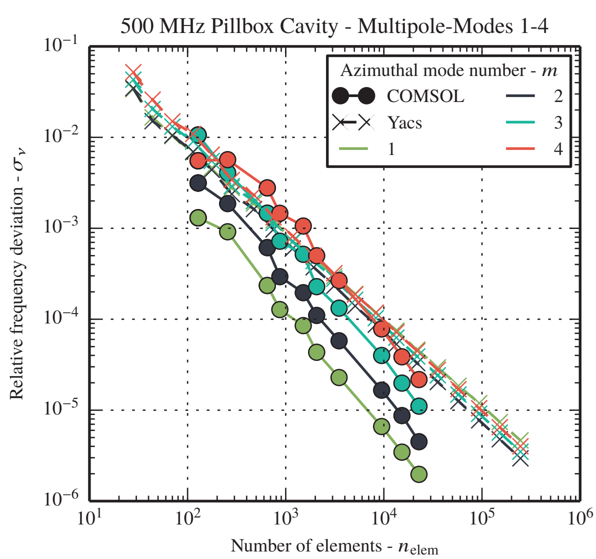

Figure 5: Relative frequency deviation of the first 4 multipole modes obtained from Yacs and COMSOL with respect to the analytical solution of a 500 MHz pillbox-cavity as a function of the number of finite elements in the problem domain.

CONCLUSION

Using Yacs, we are able to solve non-coaxial 2.5D problems in a fraction of the time required by modern 3D codes up to very high accuracies. This especially holds true for higher multipole orders, since 3D codes need to sacrifice mesh cells in order to resolve the change of the field in the azimuthal direction. Besides that, the general convergence behavior of Yacs is still limited due to the usage of first order polynomial basis functions, which is expressed in the fact that even the 3D code of COMSOL exceeds the convergence performance due to the usage of second-order basis functions. In order for Yacs to be competitive in the long term, we need to focus on higher or even arbitrary [9] order basis functions, ideally combined with adaptive mesh refinement.

REFERENCES

[1] O. Chinellato et al., “Computation of optical modes in axisymmetric open cavity resonators”, Future Generation Computer Systems 21 (2005).

[2] J.R. Shewchuk, “Triangle: Engineering a 2D Quality Mesh Generator and Delaunay Triangulator”, Applied Computational Geometry: Towards Geometric Engineering 1148 (1996).

[3] G. Guennebaud et al., “Eigen v3”, `http://eigen.tuxfamily.org`

[4] R.B. Lehoucq et al., “ARPACK Users Guide: Solution of Large Scale Eigenvalue Problems by Implicitly Restarted Arnoldi Methods”, `http://www.caam.rice.edu/software/ARPACK/`

[5] T.A. Davis, “Algorithm 832: UMFPACK V4.3—an unsymmetric-pattern multifrontal method”, ACM Transactions on Mathematical Software 30 (2004).

[6] T. Wangler, *RF Linear Accelerators*, 2nd ed., Wiley-VCH, Weinheim, Germany (2007).

[7] COMSOL Multiphysics, COMSOL, `http://www.comsol.com/`

ISBN 978-3-95450-168-7

[8] MATLAB Release 2013b, The MathWorks, Inc., http://www.mathworks.com/

[9] M. Ainsworth, J. Coyle, “Hierachic hp-edge element families for Maxwell’s equations on hybrid quadrilateral/triangular meshes”, Comput. Methods Appl. Mech. Engrg. 190 (2001).