



## Short note

# Comparison of some isoparametric mappings for curved triangular spectral elements

Richard Pasquetti <sup>a,b,\*</sup><sup>a</sup> Lab. J. A. Dieudonné, UMR CNRS 7351, Université de Nice-Sophia Antipolis, F-06108 Nice, France<sup>b</sup> INRIA project CASTOR, France

## ARTICLE INFO

## Article history:

Received 15 December 2015

Received in revised form 12 April 2016

Accepted 18 April 2016

Available online 21 April 2016

## Keywords:

Spectral element method

Isoparametric mappings

Curved triangular elements

Fekete points

Fekete–Gauss approximation

Using the spectral element method (SEM), or more generally  $hp$ -finite elements ( $hp$ -FEM), it is possible to solve with high accuracy various kinds of problems governed by partial differential equations (PDEs), see e.g. [1,2]. However, as soon as the physical domain is not polygonal, the accuracy quickly deteriorates if curved elements are not implemented. This is the reason why various methods have been developed during the last decades, starting from the celebrated transfinite interpolation proposed for quadrangular elements in [3]. In this note we revisit this problem for triangular elements, based on the use of Fekete points for interpolations and of Gauss points for quadratures, i.e. when using the so-called Fekete–Gauss approximation. As detailed in [4], such an approach shows the so-called spectral accuracy. However, differently to the quadrangles based SEM, it does not involve diagonal mass matrices, see e.g. [5–7] and references herein for works trying to preserve this nice property that is especially useful when addressing evolution problems with an explicit time marching. In the frame of the Fekete–Gauss TSEM ( $T$ , for triangle), the present study clearly points out the importance of a good choice of the bending procedure by comparing different isoparametric mappings for the Poisson and Grad–Shafranov PDEs.

Let  $\Omega$  be a two-dimensional domain and consider an affine finite element mesh  $\mathcal{T}_h$  of  $\Omega$  composed of simplices. As is well known, if  $\Omega$  is not polygonal then the mesh  $\mathcal{T}_h$  is no-longer satisfactory if using a high order  $\mathbb{P}_N$  approximation, where  $N$  is the total degree of the polynomial approximation. Usually, one substitutes an isoparametric  $\mathbb{P}_N$  mapping to the affine  $\mathbb{P}_1$  one for the triangles that should approximate the boundary  $\Gamma$  of  $\Omega$ . If  $T \in \mathcal{T}_h$  is such a triangle, we assume it to be in the simple case where one of the vertices, say  $A_1$ , is in  $\Omega$  whereas the two others,  $A_2$  and  $A_3$ , are on  $\Gamma$ , and that the segment  $A_2A_3$  is curved whereas  $A_1A_2$  and  $A_1A_3$  remain straight, see Fig. 1. This may not be sufficient to address some more complex situations where the whole mesh should be deformed and not only the boundary cells, see e.g. [8–10]. Here

\* Correspondence to: Lab. J. A. Dieudonné, UMR CNRS 7351, Université de Nice-Sophia Antipolis, F-06108 Nice, France.

E-mail address: richard.pasquetti@unice.fr.

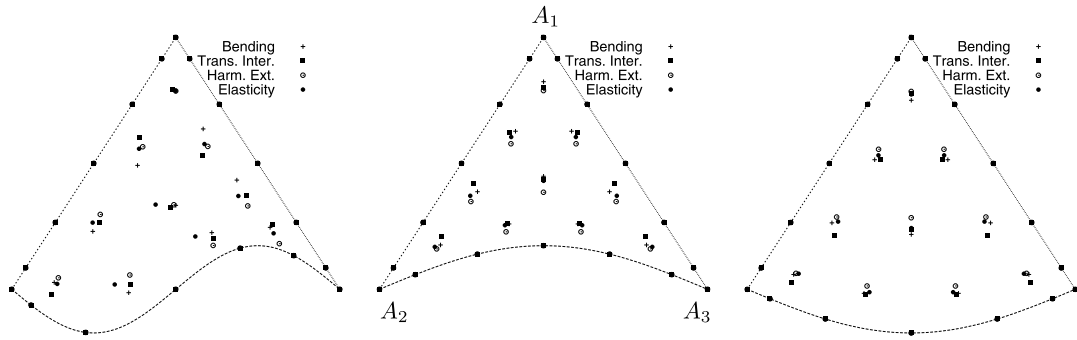


Fig. 1. Curved triangles and location of the interpolation points for the different isoparametric mappings ( $N = 6$ ).

we keep in mind the situation where the required mapping is local and not global, so that its computational cost remains negligible.

In the frame of a nodal TSEM, everything is given once knowing a set of interpolation points and a set of quadrature points in the reference element  $\hat{T} = \{(r, s) : r \in (-1, 1), s \in (-1, -r)\}$ , and the images of the interpolation points in each element  $T$  of the mesh. As first described in [4], for the Fekete–Gauss approach matrices are set up to compute at the Gauss points of  $\hat{T}$  the derivatives, with respect to  $r$  and  $s$ , of the polynomial interpolant  $u_N$  of any scalar function  $u$  defined by its values at the Fekete points. Such matrices, say  $D_r$  and  $D_s$ , are rectangular and of size  $m_p \times n_p$ , where  $m_p \geq n_p$  is the number of quadrature points, generally chosen for exact integration of polynomials of degree  $m = 2N$ , and  $n_p = (N + 1)(N + 2)/2$  is the number of interpolation points. Then, since knowing the images of the interpolation points, one can compute the Jacobian matrix and determinant at the quadrature points without expressing explicitly the isoparametric mapping from  $\hat{T}$  to  $T$ . Thus, the problem resumes to localize at best the interpolation points in the element  $T$ .

Four approaches are investigated hereafter:

- The bending procedure that we introduced in [11];
- The transfinite interpolation discussed for the triangles in [12];
- The harmonic extension;
- The linear elasticity approach, see e.g. [10].

Before going into the details of these different approaches, we assume to have at hand a parametric description of the boundary of  $\Omega$ , i.e.,  $\Gamma$  is defined by a vector function, say  $\mathbf{x}_\Gamma(t) = (x_\Gamma(t), y_\Gamma(t))$ , where  $t \in [t_{\min}, t_{\max}]$ . Then, for all the four approaches it is assumed that the boundary nodes, i.e. the interpolation points of  $\Gamma$ , are at the intersection of the lines joining the inner vertex  $A_1$  and the interpolation points of the straight line  $A_2A_3$ , i.e. the edge of the element, say  $\tilde{T}$ , provided by the piecewise linear  $\mathbb{P}_1$  mapping. This requires one to solve, in general using a simple numerical procedure,  $N - 1$  (number of nodes of an edge without the end points) generally non-linear equations. We prefer using such a geometrical approach rather than the parametrization itself, simply because the same curve may be obtained in very different manners. It remains to define  $n'_p$  inner interpolation points, with  $n'_p = n_p - 3N = (N - 1)(N - 2)/2$ .

**The bending procedure:** Assume that  $\tilde{F}$  is the interpolation point obtained with the affine  $\mathbb{P}_1$  mapping. Let  $\tilde{G}$  and  $G$  be the points at the intersections of  $A_1\tilde{F}$  with the line  $A_2A_3$  and the boundary  $\Gamma$ , respectively. Then, the bending procedure [11] consists of stating that  $F$  is homothetic to  $\tilde{F}$  by the homothety of center  $A_1$  and of ratio  $A_1G/A_1\tilde{G}$ , so that:

$$A_1F = \frac{A_1G}{A_1\tilde{G}} A_1\tilde{F} \quad \text{and} \quad \mathbf{d} \equiv \tilde{F}F = \left(\frac{A_1G}{A_1\tilde{G}} - 1\right) A_1\tilde{F}$$

where  $\mathbf{d}$  stands for displacement. One notices that for this bending procedure, only the point  $G$  of  $\Gamma$  influences the location of the interpolation point  $F$ . In order to define the inner interpolation points, one has to solve  $n'_p$  equations.

**The transfinite interpolation:** For the triangle, the method makes use of the barycentric coordinates, say  $(\lambda_1, \lambda_2, \lambda_3)$ . Recall that  $\lambda_i = 0$  stands for the edge opposite to the vertex  $i$ ,  $\lambda_i = \text{Constant}$  to a line parallel to this edge and that vertex  $i$  belongs to  $\lambda_i = 1$ . General transfinite interpolation formula are given in [12] (different approaches are however possible, see e.g. [13]). For the triangle, if the displacement  $\mathbf{d}$  vanishes at the two edges  $A_1A_2$  and  $A_1A_3$ , again assuming that  $A_2A_3$  is the curved edge, one has:

$$\mathbf{d}(\lambda_1, \lambda_2, \lambda_3) = \lambda_2 \mathbf{d}(0, 1 - \lambda_3, \lambda_3) + \lambda_3 \mathbf{d}(0, \lambda_2, 1 - \lambda_2).$$

This means that two points of the boundary  $\Gamma$  influence the location of an inner interpolation point  $F$ . They are those at the intersections of  $\Gamma$  with the lines parallel to  $A_1A_2$  and  $A_1A_3$  and passing by the point  $\tilde{F}$ . One notices that the method requires solving  $2n'_p$  equations.

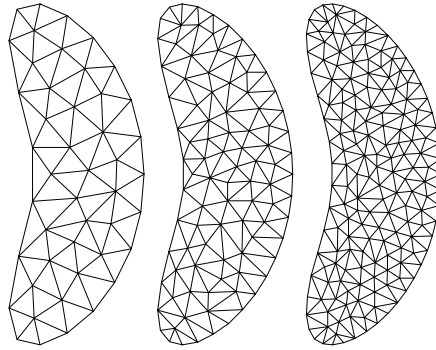


Fig. 2. Computational domain and meshes: coarse (69 elts), medium (151 elts) and fine (293 elts).

**Harmonic extension:** The goal is here to define the inner interpolation points by solving, for each curved element  $T$ , the weak form of the Laplace Dirichlet problem:

$$\Delta \mathbf{d} = 0 \quad \text{in } \tilde{T}, \quad \mathbf{d}|_{\partial \tilde{T}} = \mathbf{g}$$

with  $\mathbf{g}$  given on  $A_2A_3$  and  $\mathbf{g} = 0$  on the two other sides. To resolve this vector Laplace problem, that in fact yields two uncoupled scalar problems, one has to set up the differentiation matrices, say  $D_{\tilde{x}}$  and  $D_{\tilde{y}}$  that allows one to compute derivatives in  $\tilde{T}$ . To this end, one makes use of the differentiation matrices  $D_r$  and  $D_s$  and applies the chain rule. Since the mapping from  $\hat{T}$  to  $\tilde{T}$  is affine, the Jacobian matrix and determinant Jacobian are constant. For the same reason, one may solve directly for the interpolation point coordinates. Finally, one should invert a matrix of size  $n'_p \times n'_p$ , in order to compute the coordinates of the inner interpolation points. The approach is thus rather simple and in practice not costly. In [9], where global mappings are considered, it is however outlined that the harmonic extension may fail to define a mapping.

**Linear elasticity:** Here the curved domain is viewed as the deformation of triangles, that is governed by the equation of linear elasticity. Introducing the Lamé coefficients  $\lambda$  and  $\mu$ , the displacement field is governed by the Navier–Cauchy equation:

$$\mu \Delta \mathbf{d} + (\lambda + \mu) \nabla (\nabla \cdot \mathbf{d}) = 0 \quad \text{in } \tilde{T}, \quad \mathbf{d}|_{\partial \tilde{T}} = \mathbf{g}.$$

There is now a coupling between the components of the displacement field. Using the ingredients previously described for the harmonic extension, the weak form of this elasticity problem yields a matrix system of size  $2n'_p$ . The solution of this system provides the displacements of the inner interpolation points. Again, one can also compute directly their coordinates. Because the forcing term is zero, the mapping is here parametrized by the ratio  $\mu/(\lambda + \mu) = 1 - 2\eta$ , where  $\eta \in (-1, 0.5)$  is the so called Poisson ratio.

Fig. 1 shows the distribution of the interpolation points in a triangle, as obtained with the different strategies for a polynomial approximation degree  $N = 6$ . Of course, only the inner nodes differ. In these examples, the curved boundary is defined parametrically by:  $x = t, t \in (-1, 1)$ , and  $y = 0.3 \sin(\pi t)$  (left),  $y = 0.3 \cos(\pi t/2)$  (center) and  $y = -0.3 \cos(\pi t/2)$  (right).

In the rest of the paper the different isoparametric mappings are compared for the Poisson equation,  $-\Delta u = f$ , with exact solution  $u_{ex} = \cos(10x) \cos(10y)$  so that  $f = 200 u_{ex}$ , and, from an example given in [14], for the Grad–Shafranov PDE

$$-\nabla \cdot \left( \frac{1}{\mu x} \nabla u \right) = f,$$

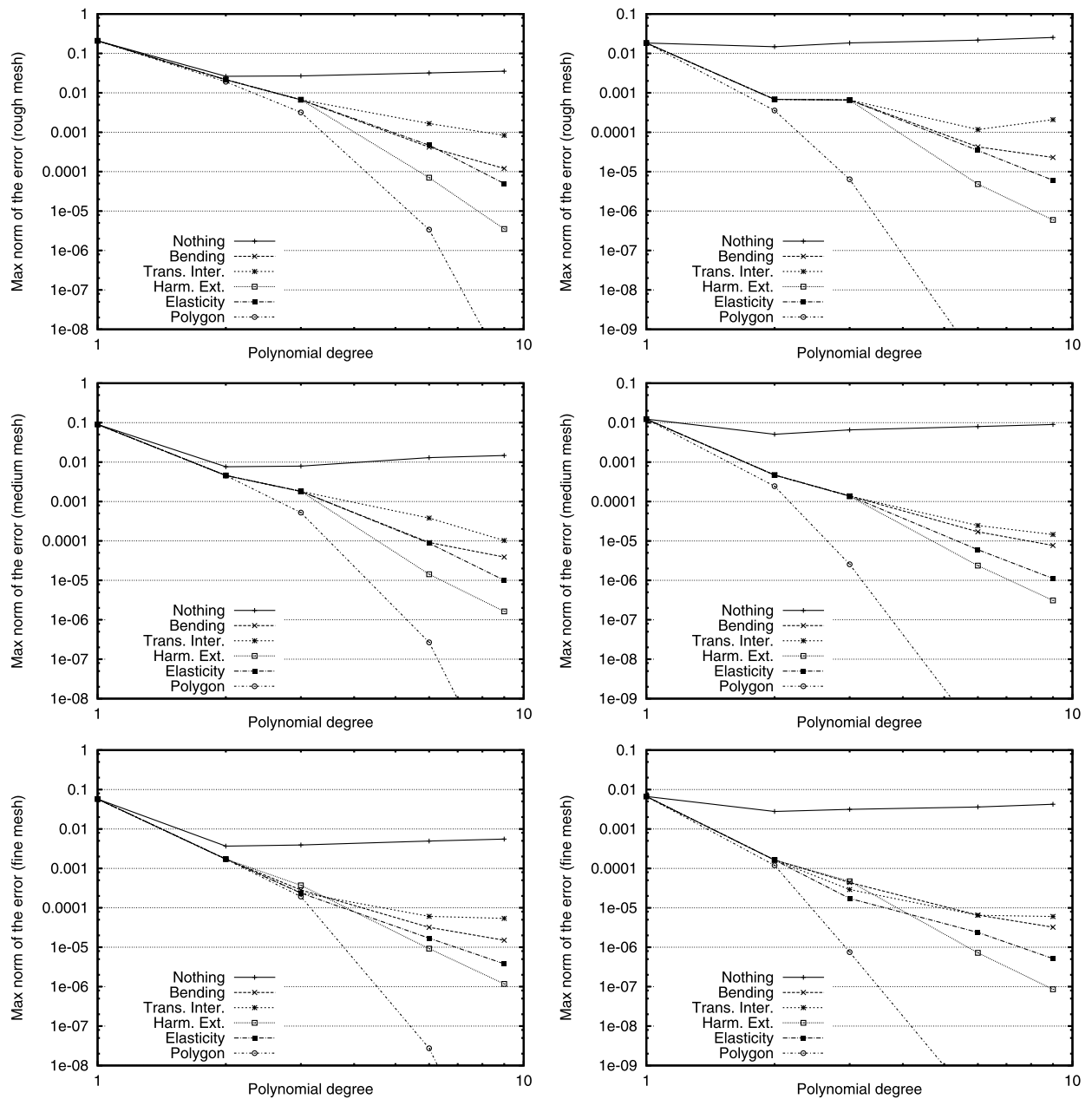
with

$$u_{ex} = -\frac{\kappa}{2\rho^3 q} \left( \frac{1}{4}(x^2 - \rho^2) + \frac{1}{\kappa^2} x^2 y^2 - a^2 \rho^2 \right), \quad f = \frac{1 + \kappa^2}{\mu \kappa \rho^3 q} x.$$

In both cases the boundary  $\Gamma$  of the computational domain is defined by

$$\Gamma = \{(x, y) : x^2 = \rho^2 + 2a\rho g(t), y = \kappa a \frac{\rho}{x} \sin(t), 0 \leq t < 2\pi\}, \quad g(t) = \cos(t) + 0.75 \exp(-2(t - \pi)^2)$$

and Dirichlet boundary conditions are used. Moreover, as in [15,16] we take  $\rho = 1, a = 0.32, \kappa = 1.7$  and  $q = 1$ , but with  $g(t) \neq \cos(t)$  to obtain a non-convex domain that includes different types of curved elements. For the Grad–Shafranov problem, the  $1/(\mu x)$  value at the Gauss points is computed by interpolation of  $(\mu x)$  from the Fekete to the Gauss points, so that it is exact for non-deformed triangles. For the linear elasticity extension, as in [10] we use  $\eta = 0.4$ , yielding  $\mu/(\lambda + \mu) = 0.2$ . Computations have also been carried out without curving the boundary triangles, then (i) assigning to the boundary points of the straight edge the corresponding values of  $u_{ex}$  on the curved boundary or (ii) simply using  $u_{ex}$ , just like if the computational domain was a polygon.



**Fig. 3.** Max norm of the error vs the polynomial degree  $N$  for the Laplace (at left) and Grad–Shafranov (at right) problems, using the coarse (top), medium (middle) and fine (bottom) meshes.

Computations have been carried out for the three  $\mathbb{P}_1$  meshes shown in Fig. 2 and for the polynomial approximation degrees  $N \in \{1, 2, 3, 6, 9\}$ . With  $u_N$  for the numerical solution and  $\{\mathbf{x}_k\}$  for the interpolation points, log–log plots of the  $l_\infty$  error,  $\epsilon = \max_k |u_N - u_{ex}|(\mathbf{x}_k) / \max_k |u_{ex}|(\mathbf{x}_k)$ , with respect to the polynomial degree  $N$  are shown in Fig. 3. Of course, if the boundary triangles are not curved then the results are bad for the real domain and on the contrary excellent for the polygonal one. If curved elements are involved, the convergence curves look more algebraic than exponential, with however high convergence rates. One notices that for  $N \leq 2$  all methods show the same accuracy, since then there is no inner interpolation point. With respect to the polygonal case, the loss of accuracy may be attributed to a failure of the Gauss quadrature formula: Indeed, since the Jacobian matrix and determinant are no longer constant in each element, polynomials of degree greater than  $2N$  are numerically integrated. We have checked that this was negligible, since using values  $m > 2N$  yields no valuable gain in accuracy, see Table 1. For curved elements the differentiation matrices are however no-longer a simple combination of the matrices  $D_r$  and  $D_s$  defined for the reference element, which implies a sensitive loss of accuracy. With respect to the standard SEM, i.e. based on quadrangular elements, the fact that the interpolation and quadrature points

**Table 1**

Max norm of the error for the Laplace and Grad–Shafranov problems for  $m \geq 2N$ , using the transfinite interpolation, the medium mesh and for  $N = 6$ .

$m$	12	14	16	18
Laplace	$3.8025 \times 10^{-4}$	$4.0246 \times 10^{-4}$	$4.0065 \times 10^{-4}$	$4.0078 \times 10^{-4}$
Grad–Shafranov	$2.4653 \times 10^{-5}$	$2.6421 \times 10^{-5}$	$2.6306 \times 10^{-5}$	$2.6313 \times 10^{-5}$

do not coincide certainly constitutes another source of inaccuracy, see e.g. the SEM results in [15]. From the results of this Note, the bending and the transfinite interpolation methods appear to be less satisfactory for elements of high degree, especially in the Grad–Shafranov case, since a slow-down in the decrease of the error is clearly observed. Our explanation is that isoparametric mappings based on PDEs are more consistent than those based on interpolations, in the sense that the locations of the inner nodes do not only result from some particular points of the actual boundary but from its polynomial interpolant, which indeed defines the boundary of the computational domain. One may conjecture that such a conclusion could extend to the tetrahedron and to  $T$ SEM approximations different from the Fekete–Gauss one.

### Acknowledgements

The  $\mathbb{P}_1$  FEM meshes have been generated with the free software “Triangle”. Special thanks to my Colleague Holger Heumann and to Lucas Drescher for providing me with the opportunity to carry out the present study and for stimulating discussions.

### References

- [1] D. Komatitsch, R. Martin, J. Tromp, M.A. Taylor, B.A. Wingate, Wave propagation in 2-D elastic media using a spectral element method with triangles and quadrangles, *J. Comput. Acoust.* 9 (2001) 703–718.
- [2] E.D. Mercerat, J.P. Vilotte, F.J. Sanchez-Sesma, Triangular spectral element simulation of two-dimensional elastic wave propagation using unstructured triangular grids, *Geophys. J. Int.* 166 (2006) 679–698.
- [3] W.J. Gordon, C.A. Hall, Construction of curvilinear co-ordinate systems and applications to mesh generation, *Int. J. Numer. Methods Biomed. Eng.* 7 (1973) 461–477.
- [4] R. Pasquetti, F. Rapetti, Spectral element methods on unstructured meshes: comparisons and recent advances, *J. Comput. Sci.* 27 (1–3) (2006) 377–387.
- [5] G. Cohen, P. Joly, J.E. Roberts, N. Tordjman, Higher order triangular finite elements with mass lumping for the wave equation, *SIAM J. Numer. Anal.* 38 (2001) 2047–2078.
- [6] F.X. Giraldo, M.A. Taylor, A diagonal mass matrix triangular spectral element method based on cubature points, *J. Eng. Math.* 56 (2006) 307–322.
- [7] W.A. Mulder, New triangular mass-lumped finite elements of degree six for wave propagation, *Prog. Electromagn. Res.* 141 (2013) 671–692.
- [8] L. Ding, Z. Lu, T. Guo, An efficient dynamic mesh generation method for complex multi-block structured grids, *Adv. Appl. Math. Mech.* 6 (1) (2014) 120–134.
- [9] A.E. Løvgren, Y. Maday, E.M. Ronquist, Global  $C^1$  maps on general domains, *Math. Models Methods Appl. Sci.* 19 (5) (2009) 803–832.
- [10] P.O. Persson, J. Peraire, Curved mesh generation and mesh refinement using Lagrangian solid mechanics, in: *Proc. of the 47th AIAA Aerospace Sciences Meeting and Exhibit, January 2009, AIAA-2009-949*.
- [11] L. Lazar, R. Pasquetti, F. Rapetti, Fekete–Gauss spectral elements for incompressible Navier–Stokes flows: the two-dimensional case, *Commun. Comput. Phys.* 13 (2013) 1309–1329.
- [12] A. Perronnet, Triangle, tetrahedron, pentahedron transfinite interpolations. application to the generation of  $C^0$  or  $G^1$ -continuous algebraic meshes, in: *Proc. Int. Conf. Numerical Grid Generation in Computational Field Simulations, Greenwich, England, 1998*, pp. 467–476.
- [13] L. Demkowicz, J. Kurtz, D. Pardo, M. Paszynski, W. Rachowicz, A. Zdunek, *Computing with hp-Adaptive Finite Elements*, vol. 1, Chapman & Hall/CRC, 2007.
- [14] H. Lijens, A. Bondeson, O. Sauter, The CHEASE code for toroidal MHD equilibria, *Comput. Phys. Commun.* 97 (1996) 219–260.
- [15] H. Heumann, L. Drescher, K. Schmidt, A high order Galerkin method for computing geometric coefficients of axisymmetric magnetohydrodynamic equilibria, INRIA-report, in preparation.
- [16] J. Lee, A. Cerfon, ECOM: a fast and accurate solver for toroidal axisymmetric MHD equilibria, *Comput. Phys. Commun.* 190 (2015) 72–88.