TECHNISCHE UNIVERSITÄT CHEMNITZ

## Sonderforschungsbereich 393

Parallele Numerische Simulation für Physik und Kontinuumsmechanik

Cornelia Pester

# $\mathcal{C}o\mathcal{C}o\mathcal{S}$

# Computation of Corner Singularities

Preprint SFB393/05-03

**Abstract** This is a documentation of the software package $\mathcal{C}o\mathcal{C}o\mathcal{S}$. The purpose of $\mathcal{C}o\mathcal{C}o\mathcal{S}$ is the computation of corner singularities of elliptic equations in polyhedral corners and crack tips. $\mathcal{C}o\mathcal{C}o\mathcal{S}$ provides a self-contained library for the generation of structured $2D$ finite element meshes, including various routines for mesh manipulation, as well as several algorithms for the solution of quadratic eigenvalue problems with Hamiltonian structure. These and further features will be described in this documentation.

**Key Words** CoCoS, corner singularities, linear elasticity

**AMS subject classification** 65F15, 65N25, 68N99

Preprintreihe des Chemnitzer SFB 393

ISSN 1619-7178 (Print) ISSN 1619-7186 (Internet)

**SFB393/05-03** **April 2005**

## Contents




Author's address:

Cornelia Pester
Institut für Mathematik
Fakultät für Bauingenieur- und Vermessungswesen
Universität der Bundeswehr München
85577 Neubiberg, Germany

cornelia.pester@unibw-muenchen.de
http://www.bauv.unibw-muenchen.de/ pester/

# 1 Introduction

Near polyhedral corners, the solutions to elliptic boundary value problems are usually not smooth in the sense of Sobolev regularity. They are composed of a regular and a singular part. To quantify the singularities, the neighbourhood of the corner is intersected with the unit ball. This ball is centered at the corner and parametrized with spherical coordinates $(r, \varphi, \theta)$, where $r$ describes the distance to the corner and $\varphi$, $\theta$ are the spherical angles.

Due to Kondrat'ev [18], the solution $U$ of the particular boundary value problem can be written as a series with terms of the form

$$r^\alpha u(\varphi, \theta),$$

where $\alpha$ is the so-called *singular exponent*. It turns out [19, 20, 26] that $\alpha$ and $u$ form an eigenpair of an eigenvalue problem which is defined on the unit sphere and associated with the given boundary value problem. The eigenvalues (singular exponents) with small real parts are of particular interest, since they quantify the corner singularities essentially.

For the computation of the corner singularities, the corresponding eigenvalue problem has to be solved. To this end, the software package $\mathcal{COCOS}$ has been developed. The two basic problems which can be solved with $\mathcal{COCOS}$ are the Laplace equation and the linear elasticity problem (Lamé problem), where each is transformed into a quadratic eigenvalue problem of the form

$$\alpha^2 m(u, v) + \alpha \tilde{g}(u, v) = \tilde{k}(u, v)$$

with certain sesquilinear forms $m$, $\tilde{g}$, $\tilde{k}$. The spectrum of this eigenvalue problem is symmetric with respect to the lines $\operatorname{Re}\alpha = -\frac{1}{2}$ and $\operatorname{Im}\alpha = 0$. In order symmetrize the spectrum with respect to the origin, the substitution $\lambda := \alpha + \frac{1}{2}$ is used and one obtains the quadratic eigenvalue problem

$$\lambda^2 m(u, v) + \lambda g(u, v) = k(u, v) \tag{1}$$

with the Hermitian sesquilinear forms $m$, $k$ and the skew-Hermitian sesquilinear form $g$, see Section 5. Indeed, if $\lambda$ is an eigenvalue of (1) then so are $-\lambda$, $\bar{\lambda}$, $-\bar{\lambda}$, see [7, 26, 27]. This property is called *Hamiltonian eigenvalue symmetry* (or quadruplet structure) motivated by the structure of the spectrum of a Hamiltonian matrix, see, for example, [23]. Figure 1 demonstrates that the eigenvalues of (1) are placed symmetric with respect to the real and the imaginary axes. If $\lambda$ is a real eigenvalue, then only the symmetry to the imaginary axis is given.

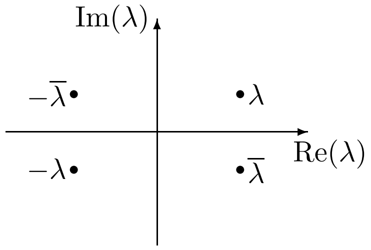



Figure 1: The quadruplet structure of the spectrum of problem (1)

For the Laplace problem, the term $g(u, v)$ vanishes. Hence, by the substitution $\hat{\lambda} = \lambda^2$, problem (1) can be transformed to a linear eigenvalue problem. In the implementation of

$\mathcal{COCOS}$, however, the eigenvalue problem associated with the Laplace problem is treated in the quadratic form (1), in order to apply the same algorithms for its solution as for the Lamé problem. While the eigenvalues and eigenfunctions are real for the Laplace problem, one has to deal with complex-valued vector functions and complex eigenvalues for the Lamé problem.

The spectral shift $\lambda = \alpha + \frac{1}{2}$ and the corresponding symmetrization of the spectrum are performed in order to apply special algorithms which exploit the quadruplet structure of the eigenvalues for an efficient solution of problem (1), see Section 6. Nevertheless, the singular exponents $\alpha$ are the terms of interest; they can be obtained easily from $\alpha = \lambda - \frac{1}{2}$.

Meanwhile, $\mathcal{COCOS}$ has been enriched by many further features which will be described in the following sections.

## 2 General features

$\mathcal{COCOS}$ is a combination of C and Fortran routines. It is written so that it can be translated on different platforms including Linux and HP workstations. The initial and basic part of the implementation including the mesh generation and the computation of the singular exponents was made by Th. Apel and U. Reichel. Additional routines for mesh generation to support the computations for crack problems were added by M. Randianarivony. Further enrichments of the code like the computation of eigenvectors, graphical output, the adding of new eigenvalue algorithms or the integration of an *hp*-version were done by C. Pester and partly by J. Rosam who also provided $\mathcal{COCOS}$ with the computation of stress distributions for the Fichera corner.

After starting $\mathcal{COCOS}$, some control parameters are read from a file named `control.dat`. If this file or some parameters are missing, default values are used instead. The list of the control parameters is summarized in Table 1 and described below. The advantage of this method is that the user does not have to reenter all the parameters on each start of the program or that the program has not to be recompiled on changing a parameter so that serial tests can be performed.

The parameter `ndof` allows to switch between the Laplace and the Lamé problems. In addition to these two model problems, it is possible to solve the mixed boundary value problem for the Laplace-Beltrami operator by choosing `ion` $= 313$.

$\mathcal{COCOS}$ provides both, the *h*- and the *p*-version of the Finite Element Method. The parameter `pgrad` can be used to choose between these two versions. For the *p*-version, the desired polynomial degree of the basis functions has to be assigned to `pgrad`. Then a coarse quadrilateral mesh is read from a file which has to be specified by the user. (The appropriate code was provided by SFB 393 of Chemnitz University of Technology [17] and adapted to our purposes.) All coordinates have to be given relative to $\pi$ (or relative to a parameter $\xi$ for the notch or the circular cone, see Section 4), this means that, for example, the point specified by $(1.5, 1.0)$ in the mesh file is read as the node $(\frac{3}{2}\pi, \pi)$. The required structure of the mesh file is summarized in Table 2 and explained in detail in [17]. The mesh can be modified afterwards; in particular, a geometric refinement to the corner is possible, so that, in fact, an *hp*-version is performed for `pgrad` $> 0$. Depending on `pgrad` and the geometry of the chosen mesh, finite element matrices are generated then.

For the *h*-version (`pgrad` $\leq 0$), linear basis functions and triangular meshes are used. Here, the user has to choose the desired domain (see Section 4), the refinement (number of divisions into the $\varphi$- and $\theta$-directions) and possibly other parameters specifying the domain (boundary conditions, opening angles etc.). Based on the entered data, the corresponding

| Name | Default value | Meaning |
|---|---|---|
| `ion` | 1 | **Output control**: `ion` $< 0$: no output; `ion` $= 0$: minimal output; `ion` $> 0$: output of additional information |
| `iso` | 1 | **Mesh type**: `iso` $= 1$: isotropic triangulation; `iso` $= 0$: anisotropic triangulation of the sphere |
| `iter` | 1000/2000 | Maximum **number of iterations** during the eigenvalue algorithm (the second value is accessed only for `mode` $= 1$) |
| `epsilon` | 5E-6/5E-4 | **Stopping criterion** for the eigenvalue iteration |
| `nint` | 3 | **Number of integration points** per element; `nint` $= 0$: the optimal value is chosen automatically |
| `pgrad` | 2 | **Polynomial degree** of the basis functions: `pgrad` $\leq 0$: $h$-version with linear basis functions on a rectangular mesh; `pgrad` $> 0$: $p$-version or $hp$-version on a quadrilateral mesh |
| `nei` | 6 | **Number of eigenvalues** to be computed |
| `typ` | 2 | **Type of the material parameters**: `typ` $= 1$: $(\lambda, \mu)$; `typ` $= 2$: $(E, \nu)$ |
| `param` | 1.0/0.1 | **Values of the material parameters** (according to `typ`) |
| `write` | 0 | **Additional output** into files |
| `ndof` | 3 | **Number of degrees of freedom**: `ndof` $= 1$: Laplace; `ndof` $= 3$: Lamé |
| `solve` | 1 | **Eigenvalue computation**: `solve`=1: yes; `solve`=0: no; if desired also computation of eigenvectors |
| `mode` | 2 | **Eigenvalue algorithm** (see Section 6) |
| `shift` | 0.5/0.0 | Real and imaginary part of the **shift value** (compute eigenvalues close to `shift`) |

Table 1: Control parameters for $\mathcal{C}o\mathcal{C}o\mathcal{S}$

| line | data | | | | | | |
|---|---|---|---|---|---|---|---|
| 1 | 0 (no material data) or 1 | | | | | | |
| 2 | Nnode Nedge Nelem Ndof NDir NNeum (Nmat Nmatinfo) | | | | | | |
| Nnode | 1 | $x_1$ | $y_1$ | | | | |
| | ⋮ | ⋮ | ⋮ | | | | |
| | Nnode | $x_{Nnode}$ | $y_{Nnode}$ | | | | |
| Nedge | 1 | $V_{1,start}$ | $V_{1,end}$ | $V_{1,middle}$ | $geom_1$ | | |
| | ⋮ | ⋮ | ⋮ | ⋮ | ⋮ | | |
| | Nedge | $V_{Nedge,start}$ | $V_{Nedge,end}$ | $V_{Nedge,middle}$ | $geom_{Nedge}$ | | |
| Nelem | 1 | $E_{1,1}$ | $E_{1,2}$ | $E_{1,3}$ | $(E_{1,4})$ | $(Mat_1)$ | |
| | ⋮ | ⋮ | ⋮ | ⋮ | ⋮ | ⋮ | |
| | Nelem | $E_{Nelem,1}$ | $E_{Nelem,2}$ | $E_{Nelem,3}$ | $(E_{Nelem,4})$ | $(Mat_{Nelem})$ | |
| NDir*(Ndof+1) (i = 1, …, NDir) | $E_{Dir,i}$ | $Bitmask_{Dir,i}$ | | | | | |
| | | $B^{Dir}_{1,i,start}$ | $B^{Dir}_{1,i,end}$ | $B^{Dir}_{1,i,middle}$ | | | Ndof |
| | | ⋮ | ⋮ | ⋮ | | | |
| | | $B^{Dir}_{Ndof,i,start}$ | $B^{Dir}_{Ndof,i,end}$ | $B^{Dir}_{Ndof,i,middle}$ | | | |
| NNeum*(Ndof+1) (i = 1, …, NNeum) | $E_{Neum,i}$ | $Bitmask_{Neum,i}$ | | | | | |
| | | $B^{Neum}_{1,i,start}$ | $B^{Neum}_{1,i,end}$ | $B^{Neum}_{1,i,middle}$ | | | Ndof |
| | | ⋮ | ⋮ | ⋮ | | | |
| | | $B^{Neum}_{Ndof,i,start}$ | $B^{Neum}_{Ndof,i,end}$ | $B^{Neum}_{Ndof,i,middle}$ | | | |
| 2*Nmat (i = 1, …, Nmat) | $Mat_i$ | $Nparam_i$ | | | | | |
| | | $P_{1,i}, \ldots, P_{Nparam_i,i}$ | | | | | |

Nnode … number of nodes
Nedge … number of edges
Nelem … number of elements
Ndof … number of degrees of freedom per node
NDir … number of Dirichlet edges
NNeum … number of Neumann edges
Nmat … number of material types
Nmatinfo … maximum number of material information per material type
$Nparam_i$ … number of material info/ parameters for material i
$Mat_i$ … material type number for element i
$P_{j,i}$ … material parameters for material $Mat_i$

$x_i$, $y_i$ … coordinates of node i
$V_{i,start}$, $V_{i,end}$ … numbers of the end nodes of the i-th edge
$V_{i,middle}$ … number of the middle node of the i-th edge if contained in the node list; otherwise 0
$geom_i$ … geometry of the i-th edge (0 for straight edges, 1 for circular arcs, 2 for parabolic edges)
$E_{i,j}$ … number of the j-th edge of element i
$1 \le j \le 3$ for triangles
$1 \le j \le 4$ for quadrilaterals
$E_{Dir,i}$ … number of the i-th Dirchlet edge
$E_{Neum,i}$ … number of the i-th Neumann edge
$Bitmask_i$ … bitmask, see below
$B^{Dir}_{j,i,*}$, $B^{Neum}_{j,i,*}$ $(1 \le j \le Ndof)$ … boundary data for the corresponding node of edge $E_{Dir,i}$ or $E_{Neum,i}$

In $\mathcal{COCOS}$ (eigenvalue problems):
first line: 0 – material data is provided by $\mathcal{COCOS}$
second line: $NNeum = 0$ – no Neumann data is specified; no material information
$Ndof = 1$ for the Laplace problem, $Ndof = 3$ for the Lamé problem
coordinates $x_i$, $y_i$ devided by $\pi$; $V_{i,middle} = 0$ for all edges; $geom_i = 0$ for all edges;
in case of Dirichlet data ($NDir > 0$): $B^{Dir}_{j,i,*} = 0$ (one value per node of the edge $E_{Dir,i}$)
$Bitmask_i$: set one bit for each degree of freedom for which the data is to be read,
in general (and in $\mathcal{COCOS}$): $Bitmask_i = 2^{Ndof} - 1$

Table 2: Overview of the structure of a mesh file which is read by $\mathcal{COCOS}$ to generate the mesh

mesh is generated and the finite element matrices are assembled. Furthermore, one can choose between isotropic and anisotropic meshes using the parameter `iso`. Isotropic means that the elements are shape regular on the sphere (they have approximately the same spatial dimensions), but they are anisotropic in the parameter domain. The anisotropic mesh means that the parameter domain is divided evenly into both directions with the effect that the elements on the sphere become the smaller the closer they are placed to a pole. In order to compute an error estimator for the eigenpairs and to do an adaptive mesh refinement, it is necessary to choose an isotropic mesh [6, 28].

After the assembly of the finite element matrices, the discretized form of problem (1) is given by the matrix eigenvalue problem: Find $\lambda \in \mathbb{C}$, $u \in \mathbb{C}^N$ such that

$$\lambda^2 Mu + \lambda Gu = Ku \tag{2}$$

with $M = M^\top \in \mathbb{R}^{N\times N}$, $G = -G^\top \in \mathbb{R}^{N\times N}$ and $K = K^\top \in \mathbb{R}^{N\times N}$, where $N$ is the problem size (depending on the refinement level of the mesh, the polynomial degree of the basis functions, the number of the degrees of freedom per node (`ndof`) and the boundary condition). Furthermore, $M$ and $K$ are positive definite.

Various algorithms were implemented into $\mathcal{COCOS}$ to compute some of the eigenvalues and corresponding eigenvectors of problem (2). The desired number of the eigenvalues to compute can be chosen by the parameter `nei`. Usually, the `nei` eigenvalues with smallest real part are computed. If one wishes to shift the spectrum, this can be done with the parameter `shift`, see Section 6.

Once the eigenvalues are computed, it is possible to compute and display the corresponding eigenvectors. For the Lamé problem, the eigenvectors correspond to displacements and can be visualized in $3D$, see [31]. Moreover, for isotropic meshes ($h$-version), an error estimator will be computed which can be used for an adaptive refinement of the mesh.

Basic routines, such as matrix-vector operations for sparse matrices or graphical output, are provided by the SFB 393 (Sonderforschungsbereich) of Chemnitz University of Technology [29].

# 3 Generation of finite element meshes

As outlined in Section 1, we consider eigenvalue problems on the unit sphere which is parametrized by the spherical angles $\varphi \in [0, 2\pi)$ and $\theta \in [0, \pi]$ via the transformation

$$x = \cos\varphi \sin\theta, \qquad y = \sin\varphi \sin\theta, \qquad z = \cos\theta.$$

The parameter domain which is spanned by $\varphi$ and $\theta$ is a bounded domain in $\mathbb{R}^2$. The unit sphere $\mathcal{S}^2$ is given by

$$\mathcal{S}^2 = \{(\varphi, \theta) \mid \varphi \in [0, 2\pi),\ \theta \in [0, \pi]\}.$$

Some example domains with their intersections with the unit sphere and the corresponding parameter domains are displayed in Section 4.

For the discretization of problem (1) and for the simplification of the implementation, we use a triangulation with straight-lined elements in the parameter domain.

As described in Section 2, three possible discretization techniques are supported by $\mathcal{COCOS}$; they are demonstrated in Figure 2.

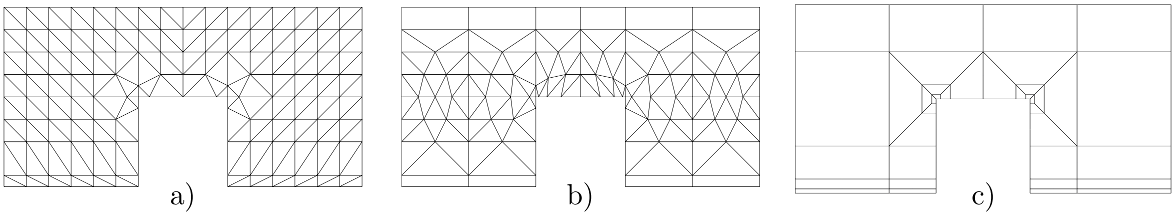



Figure 2: a) uniform triangulation (anisotropic triangulation of the sphere, `iso`=0); b) non-uniform triangulation (isotropic triangulation of the sphere, `iso`=1); c) coarse quadrilateral mesh for the $p$-version (`pgrad` $> 0$)

For a better approximation of the singularities of the solution a graded mesh refinement is used, see [7]. This means that near the corners of the domain some nodes are shifted according to a certain grading parameter $\mu$ (typical for the $h$-version, usually $\mu = 0.5$) or some elements are added scaled by a certain factor $\sigma$ (typical for the $p$-version, usually $\sigma = 0.17$).

For the $h$-version, the shifting of the nodes works as follows: The mesh is generated without grading; then the coordinates of nodes in the neighbourhood of a corner $(\varphi_{ref}, \theta_{ref})$ are redefined so that they are moved to the corner (reference point). If $(\varphi, \theta)$ are the original coordinates and if $|\varphi - \varphi_{ref}| + |\theta - \theta_{ref}| =: r < b$ for a given distance $b$ to the corner, then the new coordinates are given by

$$\begin{bmatrix} \varphi_{new} \\ \theta_{new} \end{bmatrix} = \begin{bmatrix} \varphi_{ref} \\ \theta_{ref} \end{bmatrix} + \begin{bmatrix} \varphi - \varphi_{ref} \\ \theta - \theta_{ref} \end{bmatrix} \cdot \left(\frac{r}{b}\right)^{\frac{1}{\mu}-1}.$$

The mesh grading is demonstrated in Figure 2.

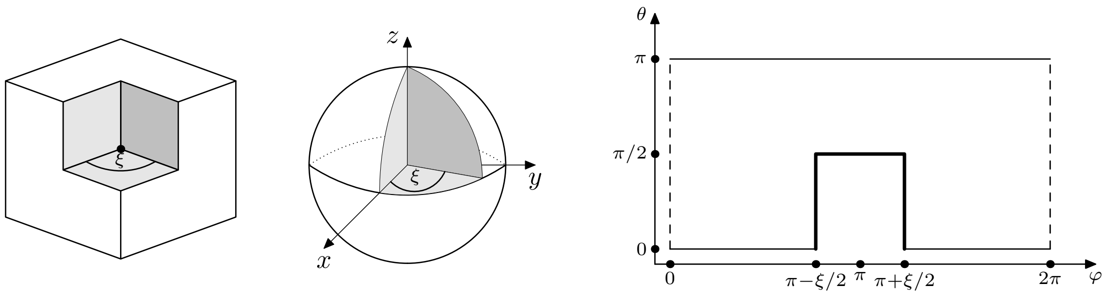
# 4 Example domains for the $h$-version

In this section, an overview of the domains is given that can be chosen as computational domains for the $h$-version in $\mathcal{COCOS}$. For each of them, some short information will be given followed by the graphics of their geometry (the $3D$ domain, its intersection with the unit sphere and the corresponding parameter domain). The parameter domains are relevant for the computations in $\mathcal{COCOS}$; they are triangulated as described in Section 2. Dashed lines in the appropriate figures correspond to periodic boundary conditions (i.e. the values of any function at $\varphi = 2\pi$ have to be the same as at $\varphi = 0$), whereas bold lines mean that either Dirichlet or Neumann boundary conditions can be chosen for this part of the boundary.

For each example domain, some numerical results are added for isotropic material with $\nu = 0.3$. The tables show the real parts of the singular exponents which are closest to zero. For those eigenvalues with $|\mathrm{Im}\,(\alpha)| > 10^{-3}$, the imaginary part is printed as well. Otherwise, complex eigenvalues are merely marked in bold face. Since $\alpha = 0$ is always an eigenvalue of multiplicity `ndof` (1 for the Laplace problem and 3 for the Lamé problem) in case of Neumann boundary conditions, these values are omitted in the tables; the tables start immediately with $\mathrm{Re}\,\alpha_{\mathtt{ndof}+1}$, where we write $\alpha_i$ instead of $\mathrm{Re}\,\alpha_i$.

**Fichera Corner** The Fichera corner is a cube where 1/8 is cut out, that is the set $(-1,1)^3 \setminus [-1,1]^3$. Extended to infinity, the Fichera corner corresponds to a polyhedral cone. Its intersection with the unit sphere is implemented into $\mathcal{COCOS}$ in a slightly generalized way, where the angle $\xi \in (0, 2\pi)$ of the corner can be varied. The parameter $\xi = \frac{\pi}{2}$ corresponds to the original Fichera domain, which is the prototype of a domain where edge and corner singularities interact. It is named after G. Fichera (1922–1996) who was the first to give an approximation of its first corner singular exponent. In 1973 he provided the bounds 0.4335 and 0.4645 (Laplace problem, Dirichlet boundary, $\xi = 90°$). Later, in 1993, the value 0.45418 was obtained by a boundary element method approximation (source: `http://perso.univ-rennes1.fr/monique.dauge/3eJS.html`).

This domain is a benchmark example proposed by Th. Apel and A. Dimitrov.




The spherical domain is described by

$$G = \mathcal{S}^2 \setminus \{(\varphi, \theta) \mid \varphi \in [0, \xi],\, \theta \in [0, \tfrac{1}{2}\pi]\}.$$

The following tables show some numerical results.

| | Dirichlet | | | Neumann | | |
|---|---|---|---|---|---|---|
| $\xi$ | $\alpha_1$ | $\alpha_2$ | $\alpha_3$ | $\alpha_2$ | $\alpha_3$ | $\alpha_4$ |
| 45° | 0.375 | 1.092 | 1.190 | 0.799 | 0.936 | 1.102 |
| 90° | 0.454 | 1.231 | 1.231 | 0.840 | 0.840 | 1.206 |
| 135° | 0.549 | 1.273 | 1.423 | 0.743 | 0.912 | 1.291 |
| 180° | 0.667 | 1.333 | 1.667 | 0.667 | 1.000 | 1.333 |
| 225° | 0.802 | 1.453 | 1.890 | 0.615 | 1.056 | 1.332 |
| 270° | 0.920 | 1.705 | 1.982 | 0.594 | 1.045 | 1.289 |
| 315° | 0.983 | 1.943 | 1.999 | 0.623 | 1.014 | 1.206 |

Eigenvalues: Fichera corner, Laplace problem

| | Dirichlet | | | Neumann | | |
|---|---|---|---|---|---|---|
| $\xi$ | $\alpha_1$ | $\alpha_2$ | $\alpha_3$ | $\alpha_4$ | $\alpha_5$ | $\alpha_6$ |
| 45° | 0.328 | 0.343 | 0.461 | 0.752 | 0.773 | 0.858 |
| 90° | 0.405 | 0.405 | 0.574 | 0.762 | 0.762 | 0.773 |
| 135° | 0.492 | 0.506 | 0.675 | 0.642 | 0.724 | 0.833 |
| 180° | 0.595 | 0.667 | 0.759 | 0.545 | 0.667 | 0.908 |
| 225° | 0.716 | 0.846 | 0.856 | 0.475 | 0.652 | 0.891 |
| 270° | 0.852 | 0.933 | 0.964 | 0.431 | 0.765 | 0.868 |
| 315° | 0.964 | 0.985 | 0.995 | 0.422 | 0.875 | 0.989 |

Eigenvalues: Fichera corner, Lamé problem

In the limit $\xi \to 0$, the Fichera corner (the polyhedral cone) corresponds to the wedge-shaped crack with $\xi = \frac{\pi}{2}$ and to the out-of-plane crack with $\xi \to 0$.

For $\xi = \pi$, the Fichera corner (the polyhedral cone) corresponds to the notch with angle $\xi = \frac{3}{2}\pi$.

In the limit $\xi \to 2\pi$, the Fichera corner (the polyhedral cone) corresponds to the notch with angle $\xi = \pi$ and to the circular cone with opening angle $\xi = \frac{\pi}{2}$.

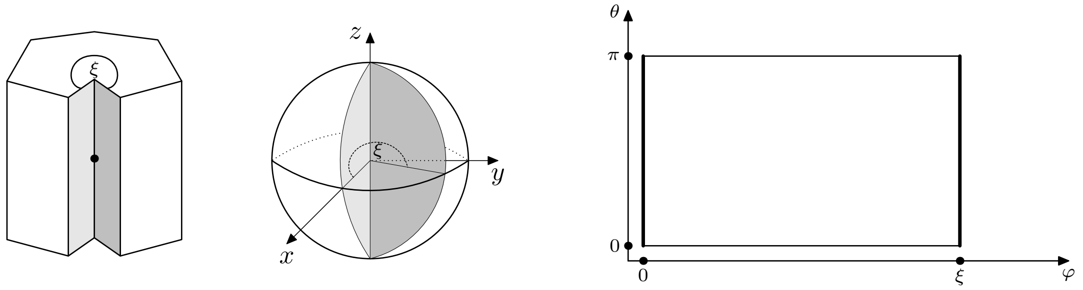
The limit values $\xi = 0$ and $\xi = 2\pi$ are not properly meshed with our algorithm.

**Notch with variable angle** A body with a straight-lined notch is considered. The planes which are adjacent to the notch span an angle $\xi \in (0, 2\pi)$. For the analysis of corner singularities, the domains with $\xi > 180$, i.e. with a concave corner, are of particular interest. The intersection with the sphere corresponds to a zone which is bounded by two geodesic lines that span the angle $\xi$.

This domain is a benchmark example proposed by A. Dimitrov. The singular exponents are those of the two-dimensional case plus all these values increased by an arbitrary natural number: For the Laplace problem with Dirichlet or Neumann boundary, the singular exponents can be expressed by $k\frac{\pi}{\xi} + j$ with $k, j \geq 0$ and for mixed boundary conditions by $(k + \frac{1}{2})\frac{\pi}{\xi} + j$.

For the Lamé system, the exponents can be calculated for isotropic material to almost arbitrary precision by solving a transcendental equation. Some of these results are summarized in the following table for the Lamé problem with mixed boundary conditions. Note that these are not necessarily the smallest eigenvalues, because a Newton iteration was used for their computation.

| $\xi$ | $\alpha_1$ |
|---|---:|
| 45° | 1.224288 |
| 90° | 1.711173 |
| 135° | 0.613445 |
| 180° | **0.500000±0.093549** i |
| 225° | 0.407127 |
| 270° | 0.869819 |
| 315° | **0.288252±0.046444** i |
| 360° | **0.250000±0.046775** i |




The spherical domain is described by

$$G = \{(\varphi, \theta) \mid \varphi \in (0, \xi),\ \theta = (0, \pi)\}.$$

The following tables show some numerical results.

Dirichlet

| $\xi$ | $\alpha_1$ | $\alpha_2$ | $\alpha_3$ |
|---|---|---|---|
| 30° | 6.001 | 7.001 | 8.002 |
| 60° | 3.000 | 4.000 | 5.000 |
| 90° | 2.000 | 3.000 | 4.000 |
| 135° | 1.333 | 2.333 | 2.667 |
| 180° | 1.000 | 2.000 | 2.000 |
| 225° | 0.800 | 1.600 | 1.800 |
| 270° | 0.667 | 1.333 | 1.667 |
| 315° | 0.572 | 1.143 | 1.573 |
| 359° | 0.502 | 1.003 | 1.503 |

Neumann

| $\xi$ | $\alpha_2$ | $\alpha_3$ | $\alpha_4$ |
|---|---|---|---|
| 30° | 1.000 | 2.000 | 3.000 |
| 60° | 1.000 | 2.000 | 3.000 |
| 90° | 1.000 | 2.000 | 2.000 |
| 135° | 1.000 | 1.333 | 2.000 |
| 180° | 1.000 | 1.000 | 2.000 |
| 225° | 0.800 | 1.000 | 1.600 |
| 270° | 0.667 | 1.000 | 1.333 |
| 315° | 0.572 | 1.000 | 1.143 |
| 359° | 0.502 | 1.000 | 1.003 |

Mixed

| $\xi$ | $\alpha_1$ | $\alpha_2$ | $\alpha_3$ |
|---|---|---|---|
| 30° | 3.000 | 4.000 | 5.000 |
| 60° | 1.500 | 2.500 | 3.500 |
| 90° | 1.000 | 2.000 | 3.000 |
| 135° | 0.667 | 1.667 | 2.000 |
| 180° | 0.501 | 1.500 | 1.502 |
| 225° | 0.403 | 1.200 | 1.405 |
| 270° | 0.339 | 1.000 | 1.343 |
| 315° | 0.295 | 0.857 | 1.301 |
| 359° | 0.264 | 0.752 | 1.254 |

Eigenvalues: Notch, Laplace problem

Dirichlet

| $\xi$ | $\alpha_1$ | $\alpha_2$ | $\alpha_3$ |
|---|---|---|---|
| 45° | 2.413 | 3.413 | 4.001 |
| 90° | 1.421 | 2.000 | 2.421 |
| 135° | 1.137 | 1.333 | 1.628 |
| 180° | 1.000 | 1.000 | 1.000 |
| 225° | 0.727 | 0.800 | 0.891 |
| 270° | 0.596 | 0.668 | 0.759 |
| 315° | 0.534 | 0.573 | 0.617 |
| 359° | 0.503 | 0.506 | 0.506 |

Neumann

| $\xi$ | $\alpha_4$ | $\alpha_5$ | $\alpha_6$ |
|---|---|---|---|
| 45° | 0.998 | 1.000 | 1.000 |
| 90° | 1.000 | 1.000 | 1.000 |
| 135° | 1.000 | 1.000 | 1.000 |
| 180° | 1.000 | 1.000 | 1.000 |
| 225° | 0.674 | 0.800 | 1.000 |
| 270° | 0.546 | 0.668 | 0.909 |
| 315° | 0.507 | 0.573 | 0.661 |
| 359° | 0.502 | 0.505 | 0.505 |

Eigenvalues: Notch, Lamé problem

Mixed

| $\xi$ | $\alpha_1$ | $\alpha_2$ | $\alpha_3$ |
|---|---|---|---|
| 45° | 1.224 | 2.000 | 2.225 |
| 90° | 0.711 | 1.000 | 1.712 |
| 135° | 0.615 | 0.668 | 0.793 |
| 180° | **0.502+0.094** i | **0.502−0.094** i | 0.503 |
| 225° | 0.407 | **0.412+0.055** i | **0.412−0.055** i |
| 270° | 0.345 | **0.351+0.039** i | **0.351−0.039** i |
| 315° | **0.303+0.048** i | **0.303−0.048** i | 0.304 |
| 359° | **0.271+0.046** i | **0.271−0.046** i | 0.275 |

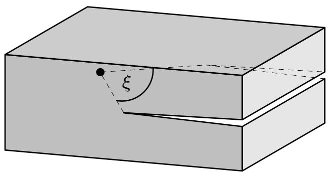
Eigenvalues: Notch, Lamé problem

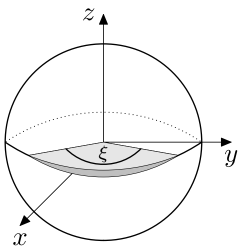
In the limit $\xi = \pi$, the notch corresponds to the Fichera corner with $\xi \to 2\pi$ and to the circular cone with opening angle $\xi = \frac{\pi}{2}$.

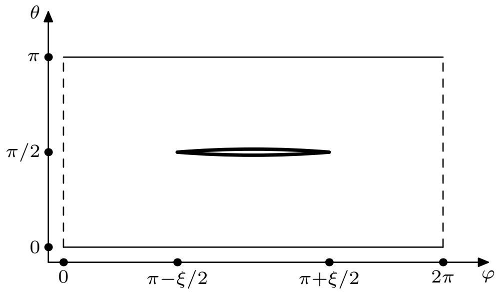
For $\xi = \frac{3}{2}\pi$, the notch corresponds to the Fichera corner with opening angle $\xi = \pi$.

In the limit $\xi \to 2\pi$, the notch corresponds to the wedge-shaped crack with $\xi = \pi$ and to the out-of-plane crack with $\xi = \pi$.

**Wedge-shaped crack** The wedge-shaped crack is a planar crack which ends in an angle $\xi \in (0, 2\pi)$ towards the crack tip.








The spherical domain is described by

$$G = \mathcal{S}^2 \setminus \{(\varphi, \theta) \mid \varphi \in [\pi - \tfrac{1}{2}\xi, \pi + \tfrac{1}{2}\xi],\, \theta = \tfrac{1}{2}\pi\}.$$

The following tables show some numerical results.

Laplace

| $\xi$ | $\alpha_2$ | $\alpha_3$ | $\alpha_4$ |
|---|---|---|---|
| 1° | 1.000 | 1.000 | 1.000 |
| 30° | 0.994 | 1.000 | 1.000 |
| 45° | 0.956 | 1.000 | 1.000 |
| 60° | 0.919 | 1.000 | 1.000 |
| 90° | 0.815 | 1.000 | 1.000 |
| 135° | 0.642 | 1.000 | 1.000 |
| 180° | 0.500 | 1.000 | 1.000 |
| 225° | 0.389 | 1.000 | 1.000 |
| 270° | 0.297 | 1.000 | 1.000 |
| 315° | 0.212 | 1.000 | 1.000 |
| 358° | 0.087 | 1.000 | 1.000 |

Lamé

| $\xi$ | $\alpha_4$ | $\alpha_5$ | $\alpha_6$ |
|---|---|---|---|
| 1° | 1.000 | 1.000 | 1.000 |
| 30° | 0.989 | 0.994 | 0.998 |
| 45° | 0.920 | 0.957 | 0.982 |
| 60° | 0.860 | 0.919 | 0.963 |
| 90° | 0.734 | 0.815 | 0.891 |
| 135° | 0.593 | 0.643 | 0.696 |
| 180° | 0.500 | 0.500 | 0.500 |
| 225° | 0.358 | 0.389 | 0.422 |
| 270° | 0.259 | 0.297 | 0.336 |
| 315° | 0.183 | 0.212 | 0.239 |
| 358° | 0.093 | 0.099 | 0.103 |

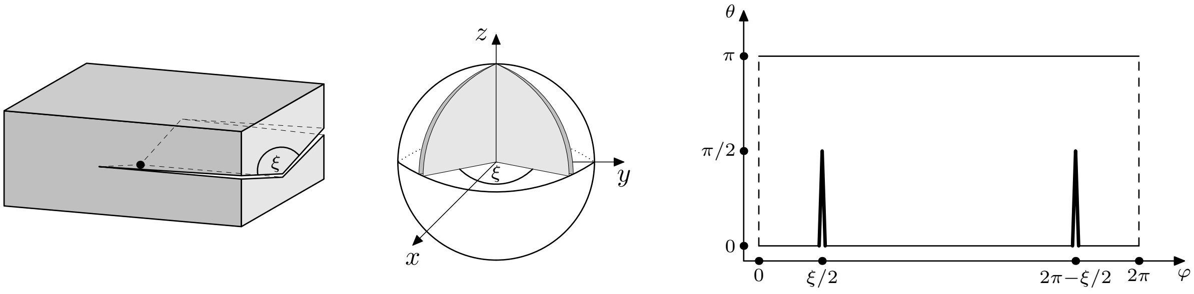
Eigenvalues: Wedge-shaped crack, Neumann boundary conditions

In the limit $\xi \to 0$, the wedge-shaped crack corresponds to the circular cone with opening angle $\xi = \pi$.

For $\xi = \frac{\pi}{2}$, the wedge-shaped crack corresponds to the Fichera corner with $\xi \to 0$ and to the out-of-plane crack with $\xi \to 0$.

For $\xi = \pi$, the wedge-shaped crack corresponds to the notch with angle $\xi \to 2\pi$ and to the out-of-plane crack with $\xi = \pi$.

**Out-of-plane crack** A notch-shaped crack is considered, where the notch has the angle $\xi \in (0, \pi]$.




The spherical domain is described by

$$G = \mathcal{S}^2 \setminus \{(\varphi, \theta) \mid \varphi \in \{\tfrac{1}{2}\xi, 2\pi - \tfrac{1}{2}\xi\}, \, \theta \in [0, \tfrac{1}{2}\pi]\}.$$

The following tables show some numerical results.

Laplace

| $\xi$ | $\alpha_2$ | $\alpha_3$ | $\alpha_4$ |
|---|---|---|---|
| 2° | 0.814 | 0.956 | 1.000 |
| 45° | 0.647 | 0.817 | 1.000 |
| 90° | 0.553 | 0.873 | 1.000 |
| 135° | 0.512 | 0.951 | 1.000 |
| 180° | 0.500 | 1.000 | 1.000 |

Lamé

| $\xi$ | $\alpha_4$ | $\alpha_5$ | $\alpha_6$ |
|---|---|---|---|
| 2° | 0.581 | 0.733 | 0.815 |
| 45° | 0.441 | 0.763 | 0.789 |
| 90° | 0.421 | 0.680 | 0.685 |
| 135° | 0.446 | 0.542 | 0.578 |
| 180° | 0.500 | 0.500 | 0.500 |

Eigenvalues: Out-of-plane crack, Neumann boundary conditions

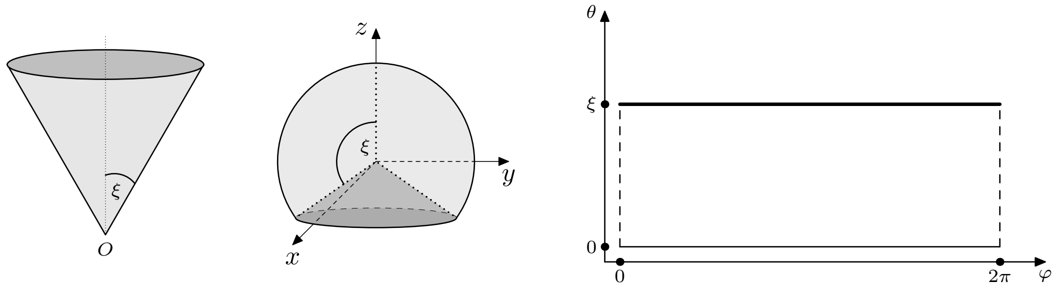
In the limit $\xi \to 0$, the out-of-plane crack corresponds to the Fichera corner with $\xi \to 0$ and to the wedge-shaped crack with $\xi = \frac{\pi}{2}$.

For $\xi = \pi$, the out-of-plane crack corresponds to the notch with angle $\xi = 2\pi$ and to the wedge-shaped crack with $\xi = \pi$.

**Circular cone** The circular cone with apex in the origin is described by its opening angle $\xi$, that is the deflection of the curved surface to the $z$-axis. For the analysis of corner singularities, only angles $\xi > \frac{\pi}{2}$ are of interest.

This domain is a benchmark example proposed by Th. Apel. The singular exponents can be calculated for isotropic materials to almost arbitrary precision by a semi-analytic approach. Those with smallest real part are summarized in the following table for the Laplace problem with Dirichlet boundary conditions, where 7–9 digits each should be correct at least.

| $\xi$ | $\alpha_1$ |
|---|---|
| 30° | 4.083687067219 |
| 45° | 2.547899192793 |
| 60° | 1.777288270251 |
| 90° | 1.000000000058 |
| 135° | 0.463098561812 |




The spherical domain is described by

$$G = \{(\varphi, \theta) \mid \varphi \in [0, 2\pi),\, \theta \in [0, \xi)\}.$$

The following tables show some numerical results.

Dirichlet

| $\xi$ | $\alpha_1$ | $\alpha_2$ | $\alpha_3$ |
|---|---|---|---|
| 30° | 4.084 | 6.840 | 6.840 |
| 45° | 2.548 | 4.407 | 4.407 |
| 60° | 1.777 | 3.196 | 3.196 |
| 90° | 1.000 | 2.000 | 2.000 |
| 135° | 0.463 | 1.245 | 1.245 |
| 179° | 0.105 | 1.000 | 1.000 |

Neumann

| $\xi$ | $\alpha_2$ | $\alpha_3$ | $\alpha_4$ |
|---|---|---|---|
| 30° | 3.121 | 3.121 | 5.494 |
| 45° | 2.000 | 2.000 | 3.634 |
| 60° | 1.468 | 1.468 | 2.753 |
| 90° | 1.000 | 1.000 | 2.000 |
| 135° | 0.857 | 0.857 | 1.245 |
| 179° | 1.000 | 1.000 | 1.000 |

Eigenvalues: Circular cone, Laplace problem

Dirichlet

| $\xi$ | $\alpha_1$ | $\alpha_2$ | $\alpha_3$ |
|---|---|---|---|
| 30° | 2.453 | **4.826+1.357** i | **4.826−1.357** i |
| 45° | 1.678 | **3.123+0.727** i | **3.123−0.727** i |
| 60° | 1.327 | **2.292+0.264** i | **2.292−0.264** i |
| 90° | 1.000 | 1.000 | 1.000 |
| 135° | **0.407** | **0.407** | 0.594 |
| 179° | 0.101 | 0.101 | 0.115 |

Neumann

| $\xi$ | $\alpha_4$ | $\alpha_5$ | $\alpha_6$ |
|---|---|---|---|
| 30° | **1.002** | **1.002** | 1.004 |
| 45° | **1.001** | **1.001** | 1.001 |
| 60° | **1.000** | **1.000** | 1.000 |
| 90° | 1.000 | **1.000** | **1.000** |
| 135° | **0.802** | **0.802** | 0.997 |
| 179° | 1.000 | 1.000 | 1.000 |

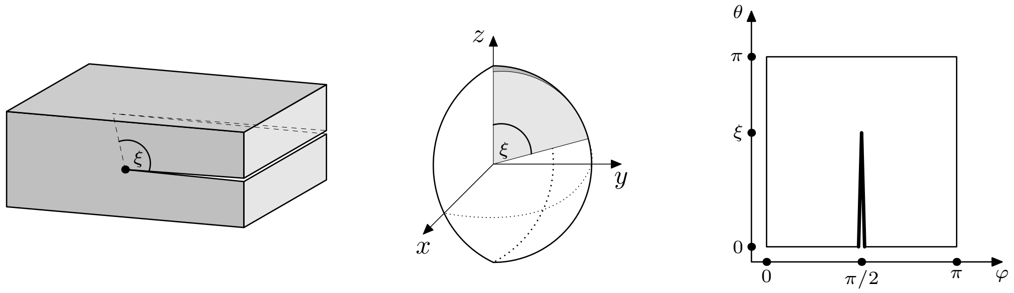
Eigenvalues: Circular cone, Lamé problem

For $\xi = \frac{\pi}{2}$, the circular cone corresponds to the Fichera corner with $\xi \to 2\pi$ and to the notch with angle $\xi = \pi$.

In the limit $\xi \to \pi$, the circular cone corresponds to the wedge-shaped crack with $\xi \to 0$.

**Bazant-Estenssoro crack** A body with a straight crack in the front face is considered. The parameter $\xi$ describes the angle between the continuation of the crack into the material and the front face. In practical applications, this angle is usually not known. It can be considered as a solution of the inverse problem: Find $\xi \in (0, \pi)$ so that $\alpha = 0.5$, where $\alpha$ is an eigenvalue of (3) for the Lamé problem.

The intersection of the described domain with the unit (half) sphere gives a crack in the $y$-$z$-plane ($x = 0$) with deflection $\xi$ to the $z$-axis.




The spherical domain is described by

$$G = \{(\varphi, \theta) \mid \varphi \in (0, \pi),\ \theta \in (0, \pi)\} \setminus \{(\varphi, \theta) \mid \varphi = \tfrac{1}{2}\pi,\ \theta \in [0, \xi]\}.$$

The following tables show some numerical results.

Laplace

| $\xi$ | $\alpha_2$ | $\alpha_3$ | $\alpha_4$ |
|---|---|---|---|
| 1° | 1.000 | 1.000 | 2.000 |
| 30° | 0.919 | 1.000 | 1.757 |
| 45° | 0.815 | 1.000 | 1.597 |
| 60° | 0.698 | 1.000 | 1.525 |
| 90° | 0.500 | 1.000 | 1.500 |
| 135° | 0.297 | 1.000 | 1.427 |
| 179° | 0.099 | 1.000 | 1.119 |

Lamé

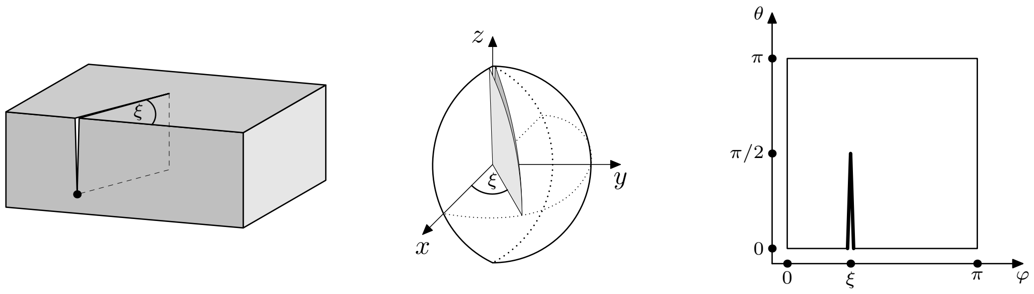
| $\xi$ | $\alpha_4$ | $\alpha_5$ | $\alpha_6$ |
|---|---|---|---|
| 1° | 1.000 | 1.000 | 1.000 |
| 30° | 0.841 | 0.893 | 1.000 |
| 45° | 0.684 | 0.795 | 1.000 |
| 60° | 0.557 | 0.701 | 1.000 |
| 90° | 0.393 | 0.548 | 0.978 |
| 135° | 0.252 | 0.362 | 0.400 |
| 179° | **0.101+0.012** i | **0.101−0.012** i | 0.113 |

Eigenvalues: Bazant-Estenssoro crack, Neumann boundary conditions

For $\xi = \frac{\pi}{2}$, the Bazant-Estenssoro crack equals the crack problem proposed by A.-M. Sändig with $\xi = \frac{\pi}{2}$.

**Crack problem** A brick with a plane crack is considered. The parameter $\xi \in (0, \pi]$ describes the angle between the crack and the front face. The point of interest is the intersection of the crack front with the front face.

This domain is a benchmark example proposed by A.-M. Sändig.




The spherical domain is described by

$$G = \{(\varphi, \theta) \mid \varphi \in (0, \pi),\ \theta \in (0, \pi)\} \setminus \{(\varphi, \theta) \mid \varphi = \xi,\ \theta \in [0, \tfrac{1}{2}\pi]\}.$$

The following tables show some numerical results.

| Laplace | | | |
|---|---|---|---|
| $\xi$ | $\alpha_2$ | $\alpha_3$ | $\alpha_4$ |
| 30° | 0.607 | 1.000 | 1.245 |
| 45° | 0.553 | 1.000 | 1.337 |
| 60° | 0.522 | 1.000 | 1.421 |
| 90° | 0.500 | 1.000 | 1.500 |

| Lamé | | | |
|---|---|---|---|
| $\xi$ | $\alpha_4$ | $\alpha_5$ | $\alpha_6$ |
| 30° | 0.405 | 0.771 | 1.000 |
| 45° | 0.388 | 0.676 | 1.000 |
| 60° | 0.385 | 0.608 | 1.000 |
| 90° | 0.393 | 0.548 | 0.978 |

Eigenvalues: Crack problem, Neumann boundary conditions

For $\xi = \frac{\pi}{2}$, the crack problem equals the Bazant-Estenssoro crack with $\xi = \frac{\pi}{2}$.

Some computational results for the wedge-shaped crack, the out-of-plane crack and the Bazant-Estenssoro crack were already given in [20]. The crack problem was studied, for example, in [14].

For the Lamé problem, $\mathcal{COCOS}$ provides the possibility of a three-dimensional graphical output of the solutions (in combination with the program `fem_ogl` [31]), where the displacements are added to the coordinates of the spherical domain. Figure 3 shows the corresponding solutions for the Bazant-Estenssoro crack and the crack problem $\xi = 45°$ each.

## 5 Assembly of the matrices

After the generation of the mesh, the system matrices $M$, $G$, $K$ are assembled. To this end, the element matrices $M_{el}$, $G_{el}$ and $K_{el}$ are computed for each element.

The original eigenvalue problem reads

$$\alpha(\alpha+1)m(u,v) + (\alpha+1)d(u,v) - \alpha\overline{d(v,u)} = \tilde{k}(u,v) \tag{3}$$

with the singular exponent $\alpha$ and the sesquilinear forms

$$\left.\begin{array}{rcl} m(u,v) &=& \int u\,\bar{v}\,\mathrm{d}\omega \\ d(u,v) &=& 0 \\ \tilde{k}(u,v) &=& \int \nabla_{\mathcal{S}} u \cdot \nabla_{\mathcal{S}} \bar{v}\,\mathrm{d}\omega \end{array}\right\} \quad \begin{array}{l}\text{if } \texttt{ndof} = 1 \\ \text{(Laplace problem)}\end{array} \tag{4}$$

or

$$\left.\begin{array}{rcl} m(u,v) &=& \int a_{ijkh} A^j A^h u^i \bar{v}^k\,\mathrm{d}\omega \\ d(u,v) &=& \int a_{ijkh}(B^j\partial_\theta u_i + C^j\partial_\varphi u^i)(A^h\bar{v}^k)\,\mathrm{d}\omega \\ \tilde{k}(u,v) &=& \int a_{ijkh}(B^j\partial_\theta u_i + C^j\partial_\varphi u^i)(B^h\partial_\theta\bar{v}^k + C^h\partial_\varphi\bar{v}^k)\,\mathrm{d}\omega \end{array}\right\} \quad \begin{array}{l}\text{if } \texttt{ndof} = 3 \\ \text{(Lamé problem)}\end{array} \tag{5}$$

where Einstein's summation convention is used, $i,j,k,h \in \{1,2,3\}$, and where $A^i$, $B^i$, $C^i$, $u^i$, $v^i$ denote the $i$-th component of $A$, $B$, $C$, $u$, $v$, respectively, with

$$A = \begin{bmatrix} \cos\varphi\sin\theta \\ \sin\varphi\sin\theta \\ \cos\theta \end{bmatrix}, \qquad B = \begin{bmatrix} \cos\varphi\cos\theta \\ \sin\varphi\cos\theta \\ -\sin\theta \end{bmatrix}, \qquad C = \frac{1}{\sin\theta}\begin{bmatrix} -\sin\varphi \\ \cos\varphi \\ 0 \end{bmatrix},$$

see [26] for details. The symbols $\nabla_{\mathcal{S}}$ and $\mathrm{d}\omega$ denote the spherical gradient and the spherical surface element, respectively, i.e.

$$\nabla_{\mathcal{S}} u \cdot \nabla_{\mathcal{S}} \bar{v} = \frac{1}{\sin^2\theta}\partial_\varphi u \partial_\varphi \bar{v} + \partial_\theta u \partial_\theta \bar{v} \qquad \text{and} \qquad \mathrm{d}\omega = \sin\theta\,\mathrm{d}\varphi\,\mathrm{d}\theta$$

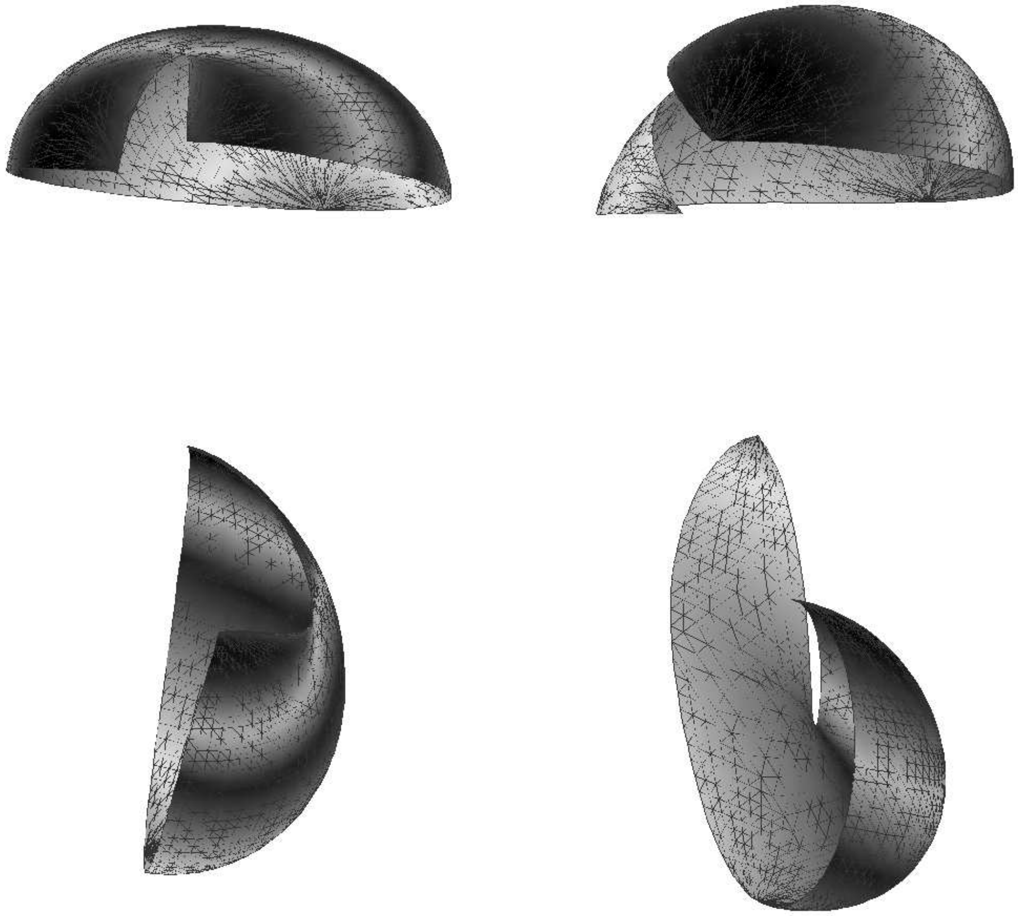


Figure 3: $3D$ illustration of the eigenfunctions (left: Bazant-Estenssoro crack, $\xi = 45°$, $\alpha = 0.684$; right: Crack problem $\xi = 45°$, $\alpha = 0.388$) shown from different viewpoints

for spherical coordinates.

For the definition of the element matrices, the functions $u$ and $v$ are replaced by nodal basis functions. In the $h$-version, linear basis functions are used. For triangles, they are given by

$$\begin{aligned}
\phi_{x_{1,T}} &= 1-\wp^{-1}[(\varphi-\varphi_{1,T})(\theta_{3,T}-\theta_{2,T})-(\varphi_{3,T}-\varphi_{2,T})(\theta-\theta_{1,T})],\\
\phi_{x_{2,T}} &= \wp^{-1}[(\varphi-\varphi_{1,T})(\theta_{3,T}-\theta_{1,T})-(\varphi_{3,T}-\varphi_{1,T})(\theta-\theta_{1,T})],\\
\phi_{x_{3,T}} &= \wp^{-1}[(\varphi_{2,T}-\varphi_{1,T})(\theta-\theta_{1,T})-(\varphi-\varphi_{1,T})(\theta_{2,T}-\theta_{1,T})],
\end{aligned}$$

where $\wp=(\varphi_{2,T}-\varphi_{1,T})(\theta_{3,T}-\theta_{1,T})-(\varphi_{3,T}-\varphi_{1,T})(\theta_{2,T}-\theta_{1,T})$. In an isotropic triangulation of the sphere, the pole elements appear as rectangles in the parameter domain, see Figure 2. The corresponding (bilinear) basis functions are given by

$$\begin{aligned}
\phi_{x_{1,T}} &= \wp^{-1}(\varphi_{2,T}-\varphi)(\theta_{3,T}-\theta),\\
\phi_{x_{2,T}} &= \wp^{-1}(\varphi-\varphi_{1,T})(\theta_{3,T}-\theta),\\
\phi_{x_{3,T}} &= \wp^{-1}(\varphi_{2,T}-\varphi_{1,T})(\theta-\theta_{1,T})
\end{aligned}$$

with the scaling factor $\wp=(\varphi_{2,T}-\varphi_{1,T})(\theta_{3,T}-\theta_{1,T})$.

For the $p$-version, integrated Legendre polynomials are used. If

$$P_n(x)=\frac{1}{2^n n!}\frac{\mathrm{d}^n}{\mathrm{d}x^n}(x^2-1)^n$$

is the $n$-th Legendre polynomial ($n=0,1,2,\dots$), then the integrated Legendre polynomial is defined by

$$L_n(x)=\gamma_n\int_{-1}^{x}P_{n-1}(\xi)\,\mathrm{d}\xi \quad \text{for } n\geq 2,$$

where

$$\gamma_n=\sqrt{\frac{(2n-3)(2n-1)(2n+1)}{4}}.$$

The first two integrated Legendre polynomials are defined by

$$L_0(x)=\frac{1+x}{2},\quad L_1(x)=\frac{1-x}{2}.$$

For other representations and properties of the Legendre polynomials and their recursive definition, see for example [11, 1, 2, 32] and references therein. The nodal basis functions are defined by

$$\phi_{n\ell}(x,y)=L_n(x)L_\ell(y),\qquad 0\leq n,\ell\leq p,$$

on the reference element $(-1,1)^2$, where $p=$ `pgrad` is the maximum polynomial degree in one variable.

This means that there are $N_{el}:=(p+1)^2$ basis functions for each element in the $p$-version (1 for each node, $p-1$ for each edge and $(p-1)^2$ for the inner element), whereas only $N_{el}:=3$ per element are used in the $h$-version (1 for each node). Hence, the element matrices have `ndof` $\times$ `ndof` blocks each of the dimension $N_{el}\times N_{el}$.

Let the nodal basis functions be numbered as $\phi_1,\dots,\phi_{N_{el}}$, and let for a fixed value $r\in\{1,\dots,\mathtt{ndof}\}$ the vector function $\underline{\phi}_n^r$ be defined by $\underline{\phi}_n^r:=[\phi_n^1,\dots,\phi_n^{\mathtt{ndof}}]^\top$ with $\phi_n^i:=\delta_{ir}\phi_n$ for $i=1,\dots,\mathtt{ndof}$. The integrals

$$(M_{el})_{n\ell}^{rs}:=m(\underline{\phi}_n^r,\underline{\phi}_\ell^s),\quad (D_{el})_{n\ell}^{rs}:=d(\underline{\phi}_n^r,\underline{\phi}_\ell^s)\quad\text{and}\quad (K_{el})_{n\ell}^{rs}:=k(\underline{\phi}_n^r,\underline{\phi}_\ell^s) \tag{6}$$

defined in (4) and (5) are computed numerically by Gauß quadrature for $n, \ell = 1, \ldots, N_{el}$ and for each pair $(r, s)$ of degrees of freedom, $r, s = 1, \ldots, \texttt{ndof}$.

The element matrices $M_{el} = \{M_{el}^{ij}\}_{i,j=1}^{3}$, $D_{el} = \{D_{el}^{ij}\}_{i,j=1}^{3}$ and $K_{el} = \{K_{el}^{ij}\}_{i,j=1}^{3}$ are generated so that

$$\left.\begin{array}{rcl} (M_{el})_{n\ell}^{rs} &=& M_{el}^{ij} \\ (D_{el})_{n\ell}^{rs} &=& D_{el}^{ij} \\ (K_{el})_{n\ell}^{rs} &=& K_{el}^{ij} \end{array}\right\} \iff i = N_{el} \cdot (r-1) + n, \quad j = N_{el} \cdot (s-1) + \ell$$

Note that all element matrices are real and that

$$M_{el} = M_{el}^{\top} \qquad \text{and} \qquad K_{el} = K_{el}^{\top}$$

due to the classical symmetry properties of the elasticity coefficients $a_{ijkh}$. Moreover, when $d(u, v)$ corresponds to the element matrix $D_{el}$, then $\overline{d(v, u)}$ corresponds to $D_{el}^{\top}$.

It was outlined in Section 1 that the spectrum of the quadratic eigenvalue problem (3) can be shifted so that it is symmetric with respect to the origin. To this end, one substitutes $\lambda = \alpha + \frac{1}{2}$; problem (3) is then transformed into

$$\lambda^2 m(u, v) + \lambda(d(u, v) - \overline{d(v, u)}) = \tilde{k}(u, v) + \frac{1}{4} m(u, v) - \frac{1}{2}(d(u, v) + \overline{d(v, u)}).$$

Defining

$$g(u, v) := d(u, v) - \overline{d(v, u)} \quad \text{and} \quad k(u, v) := \tilde{k}(u, v) + \frac{1}{4} m(u, v) - \frac{1}{2}(d(u, v) + \overline{d(v, u)}),$$

one obtains the eigenvalue problem (1),

$$\lambda^2 m(u, v) + \lambda g(u, v) = k(u, v).$$

By analogy, we define the element matrices

$$G_{el} := D_{el} - D_{el}^{\top} \quad \text{and} \quad K_{el} := K_{el} + \frac{1}{4} M_{el} - \frac{1}{2}(D_{el} + D_{el}^{\top}).$$

Corresponding to the given basis (the nodal basis for the $h$-version or the basis spanned by the Legendre polynomials for the $p$-version), the element matrices $M_{el}$, $G_{el}$ and $K_{el}$ are then assembled into the system matrices $M$, $G$ and $K$ which define the quadratic eigenvalue problem (2).

Parts of the code for the $p$-version were taken from the program package SPC-PM-Po2p which was developed in the SFB 393 "Numerische Simulation auf massiv parallelen Rechnern" at Chemnitz University of Technology in 1999–2003 by S. Beuchler, see [12].

# 6 Solving the quadratic eigenvalue problem

## 6.1 Eigenvalue computation

There are numerous methods to solve quadratic matrix eigenvalue problems of type (2),

$$\lambda^2 M u + \lambda G u = K u.$$

An overview of matrix and eigenvalue computations is given in the book by D. Watkins [33].

One method suggested by A. Meyer is the method of simultaneous iteration and computation of gradients [24, 25]. It was the first to be implemented into $\mathcal{C}\mathcal{O}\mathcal{C}\mathcal{O}\mathcal{S}$ and can be started with `mode` $= 1$. Meanwhile, the simultaneous iteration has been defeated by much more powerful eigenvalue algorithms which compute the eigenvalues of a Hamiltonian, a skew-Hamiltonian or a symplectic matrix, see Table 3 for an overview of these matrix types.

| | | |
|---|---|---|
| Let $J = \begin{bmatrix} O & I \\ -I & O \end{bmatrix}$. A real $2 \times 2$-block matrix $A$ is called | | |
| *Hamiltonian* | *skew-Hamiltonian* if | *symplectic* |
| $(JA)^\top = JA$ | $(JA)^\top = -JA$ | $A^\top JA = J$ |
| | Properties: | |
| $A = \begin{bmatrix} F & G \\ H & -F^\top \end{bmatrix}$ with $G = G^\top$, $H = H^\top$ | $A = \begin{bmatrix} F & G \\ H & F^\top \end{bmatrix}$ with $G = -G^\top$, $H = -H^\top$ | $A$ is nonsingular, $\det A = \pm 1$ |
| | Structure of the spectrum of $A$: | |
| Quadruplets: $\left(\lambda,\ -\lambda,\ \bar{\lambda},\ -\bar{\lambda}\right)$ | Pairs: $\left(\lambda,\ \bar{\lambda}\right)$ with even algebraic multiplicity | Quadruplets: $\left(\lambda,\ \frac{1}{\lambda},\ \bar{\lambda},\ -\frac{1}{\bar{\lambda}}\right)$ |

Table 3: Hamiltonian, skew-Hamiltonian and symplectic matrices

The basic idea is to linearize the quadratic eigenvalue problem. For example, one can define $v := \lambda u$ and obtains the equivalent generalized eigenvalue problem

$$\begin{bmatrix} O & I \\ K & -G \end{bmatrix} \begin{bmatrix} u \\ v \end{bmatrix} = \lambda \begin{bmatrix} I & O \\ O & M \end{bmatrix} \begin{bmatrix} u \\ v \end{bmatrix}. \tag{7}$$

Since $M$ is positive definite, we can write (7) as a standard eigenvalue problem,

$$\begin{bmatrix} I & O \\ O & M^{-1} \end{bmatrix} \begin{bmatrix} O & I \\ K & -G \end{bmatrix} \begin{bmatrix} u \\ v \end{bmatrix} = \lambda \begin{bmatrix} u \\ v \end{bmatrix},$$

which can be solved with the Arnoldi algorithm and is implemented under `mode` $= -1$ in $\mathcal{C}\mathcal{O}\mathcal{C}\mathcal{O}\mathcal{S}$. The matrices in the generalized eigenvalue problem (7), however, have no special structure. The Hamiltonian structure of the spectrum of (2) is not exploited and each quadruplet of eigenvalues is computed in four separate steps.

In order to exploit the Hamiltonian structure, another linearization is needed. In [23], it was suggested to define $v := \lambda M u$ and to transform (2) into an eigenvalue problem for a skew-Hamiltonian–Hamiltonian pencil and finally into an eigenvalue problem for a Hamiltonian matrix, see also [3, 22]. Alternatively, the same result is obtained from the definition

$$v := \lambda M u + \frac{1}{2} G u,$$

which yields a standard eigenvalue problem

$$H \begin{bmatrix} v \\ u \end{bmatrix} = \lambda \begin{bmatrix} v \\ u \end{bmatrix}$$

for the Hamiltonian matrix $H \in \mathbb{R}^{2N \times 2N}$,

$$H = \begin{bmatrix} I & -\frac{1}{2}G \\ O & I \end{bmatrix} \begin{bmatrix} O & K \\ M^{-1} & O \end{bmatrix} \begin{bmatrix} I & -\frac{1}{2}G \\ O & I \end{bmatrix},$$

see [5].

Algorithms which are based on matrix vector multiplications (like the Arnoldi method, see [33]) yield the largest eigenvalues of $H$. Since the eigenvalues with smallest real part are of interest, the matrix $H$ has to be inverted and the spectrum can simultaneously be shifted to a certain desired point $\tau$ in the complex plane (the control parameter `shift` in $\mathcal{COCOS}$). Since the matrix $(H - \tau I)^{-1}$ loses the Hamiltonian structure, it has to be manipulated further. In Table 4, some modifications are summarized with the corresponding matrix types. For each of these types, algorithms were developed, which exploit the specific structure of the spectrum and produce each quadruplet of eigenvalues in one instead of four steps.

| Matrix modification | Structure |
|---|---|
| $C_2 := (H - \tau I)^{-1}(H + \tau I)^{-1}$ | real and *skew-Hamiltonian* if $\tau$ is real or purely imaginary |
| $C_4 := (H - \tau I)^{-1}(H + \bar{\tau} I)^{-1}(H + \bar{\tau} I)^{-1}(H + \tau I)^{-1}$ | real and *skew-Hamiltonian* if $\tau$ is complex |
| $N_0 := H^{-1}$, $N_2 := H^{-1}C_2$, $N_4 := H^{-1}C_4$ | real and *Hamiltonian* |
| $S_2 := (H - \tau I)^{-1}(H + \tau I)$ | real and *symplectic* if $\tau$ is real |

Table 4: Matrix modifications for $H$

For the skew-Hamiltonian case, the Arnoldi algorithm has been adapted, see [3, 4, 22, 23]. It is called *skew-Hamiltonian implicitly restarted Arnoldi* (*shira*) process. For its implementation, the Fortran software package ARPACK [21] for the Arnoldi algorithm is modified. The *shira* process can be started in $\mathcal{COCOS}$ under `mode` = 2 or `mode` = 4 for real or complex shifts $\tau$, respectively.

The Hamiltonian and symplectic cases are treated with special Lanczos algorithms, see for instance [15, 9, 10]. A summary is given in the paper by D. Watkins [34] on which our implementations are based. The *Hamiltonian implicitly restarted Lanczos* (*hirl*) process and the *Implicitly restarted symplectic Lanczos* (*irsl*) process run in $\mathcal{COCOS}$ under `mode` = 5 and `mode` = 6, respectively. $\mathcal{COCOS}$ provides only the unshifted version of the *hirl* process (i.e. for $N_0 = H^{-1}$, see Table 3), since the shifted versions turned out to be less efficient in a matlab implementation.

In each iteration of the algorithms, a linear system of equations has to be solved. The corresponding system matrices are decomposed, either with a Cholesky factorization using the package TAUCS [16] if the matrix is positive definite, or, otherwise, with an LU factorization using the package SUPERLU [13].

The implemented algorithms are described more detailed in [30]. Numerical results are published in [5].

### 6.2 Eigenvector computation

Once, the eigenvalues are computed, the corresponding eigenvectors can be obtained from a few steps of inverse iteration. If $\lambda_H$ is an approximation of an eigenvalue $\lambda$ and $u^0$ is any non-zero vector in $\mathbb{C}^N$, then the iteration

$$\text{solve} \quad (\lambda_H^2 M + \lambda_H G - K)u^k = u^{k-1}, \qquad k = 1, 2, \ldots$$

converges very fast to a good approximation of the exact eigenvector. In $\mathcal{COCOS}$, this method is implemented for the *shira* process, where already one step yields good results. The drawback of this method is, that another linear systems of equations has to be solved which is rather time-consuming.

In the Lanczos processes, however, the eigenvectors are obtained almost for free. The left and right eigenvectors can be computed for a given eigenvalue $\lambda$ by a few matrix-vector multiplications, see for example [30].

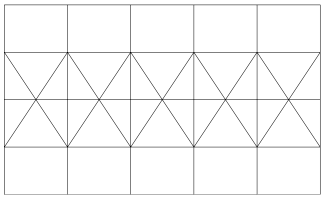
## 7 Adaptive refinement

In [6] and [28], residual a posteriori error estimates were deduced for the mixed boundary value problem and the eigenvalue problem for the Laplace-Beltrami operator for the $h$-version on (isotropic) triangular meshes. Depending on the computed error estimator in each element, the mesh can be refined adaptively in order to compute an improved solution.

For the adaptive refinement of the mesh, the standard red-green refinement procedure is applied, see for example [8]. The algorithm used in $\mathcal{COCOS}$ is composed of the following three steps:

**Initialization**

- Start with a coarse initial mesh, for example 5 divisions into the $\varphi$-direction and 4 divisions into the $\theta$-direction.



- For each element $T$: Compute the error estimator $\eta_T$.
- Compute the maximum element error $\eta_{max} = \max_T \eta_T$.
- Define a scaling factor $\alpha \leq 1$: all elements with $\eta_T > \alpha\eta_{max}$ will be refined.
- Define a percentage $\beta$: at least $\beta \cdot N_{elem}$ elements will be refined, where $N_{elem}$ is the current number of elements.
- Define a scaling factor $\gamma < 1$: decrease $\alpha$ by the factor $\gamma$ if more elements have to be refined.

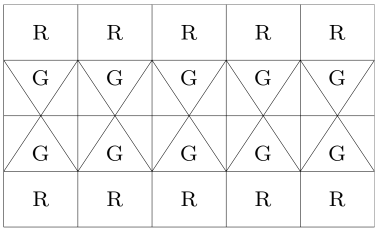
## Marking of the elements

- Repeat
  - $n_{marked} := 0$;
  - For each element $T$:
    - If $\eta_T > \alpha \cdot \eta_{max}$: mark all edges of $T$; $n_{marked} := n_{marked} + 1$;
  - $\alpha := \gamma \cdot \alpha$

  until $n_{marked} \geq \beta \cdot N_{elem}$.
- Repeat
  - $n_{new} := 0$; $n_{red} := 0$.
  - For each element $T$:
    - If $number\_of\_marked\_edges(T) = 3$:
      - mark $T$ **red**; $n_{red} := n_{red} + 1$;
    - Else if $number\_of\_marked\_edges(T) = 2$:
      - mark all edges of $T$; $n_{new} := n_{new} + 1$.
    - Else if $number\_of\_marked\_edges(T) = 1$:
      - If $T$ is a pole element:
        mark all edges of $T$; $n_{new} := n_{new} + 1$;
      - Else: mark $T$ **green**.

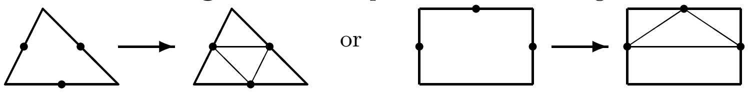
  until $n_{new} = 0$.

**Example:**

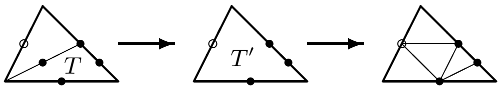



## Refinement of the elements

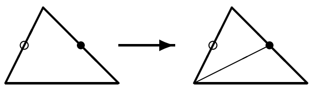
- For each element $T$:
  - If $T$ is marked **red**:
    - If $T$ was not marked **green** before: split $T$ into four triangles.




    - If $T$ is the result of a former green refinement of an element $T'$:
      resolve $T'$ and refine $T'$ like a **red** triangle;
      perform a **green** refinement to the free edge.

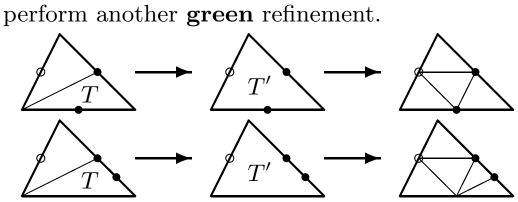



  - If $T$ is marked **green**:
    - If $T$ was not marked **green** before: split $T$ into two triangles.



    - If $T$ is the result of a former green refinement of an element $T'$:
      resolve $T'$ and refine $T'$ like a **red** triangle;
      if necessary perform another **green** refinement.

a)
b)



For technical reasons, pole elements (rectangles) are not marked green. In case they are marked, always the red refinement is performed, where the rectangular structure at the pole is preserved. Figure 4 shows a sequence of meshes that was created in the course of an adaptive refinement for the mixed boundary value problem for the Laplace-Beltrami operator.

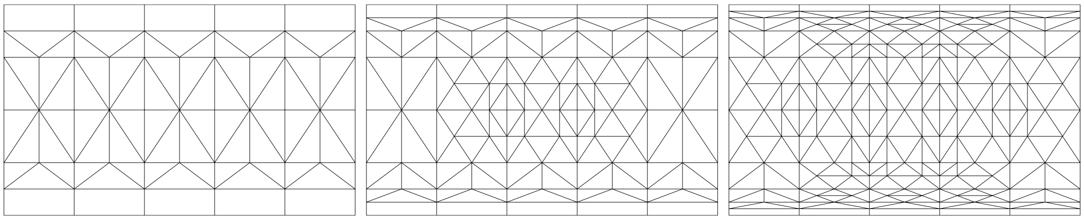


Figure 4: A series of meshes in an adaptive refinement process

## Acknowledgement

Special thanks go to the DFG (Deutsche Forschungsgemeinschaft) for supporting this work and to the SFB 393 (Sonderforschungsbereich) of Chemnitz University of Technology for providing some of the software used by $\mathcal{COCOS}$.

## References

[1] Legendre polynomial. 1999 CRC Press LLC, Wolfram Research, Inc., Internet source. `http://mathworld.wolfram.com/LegendrePolynomial.html`.

[2] Legendre polynomial. efunda, Internet source. `http://www.efunda.com/math/legendre/index.cfm`.

[3] Th. Apel, V. Mehrmann, and D. Watkins. Structured eigenvalue methods for the computation of corner singularities in 3D anisotropic elastic structures. *Comput. Methods Appl. Mech. Engrg.*, 191:4459–4473, 2002.

[4] Th. Apel, V. Mehrmann, and D. Watkins. Numerical solution of large scale structured polynomial or rational eigenvalue problems. In F. Cucker, R. DeVore, P. Olver, and E. Süli, editors, *Foundations of Computational Mathematics, Minneapolis 2002*, volume 312 of *Lecture Note Series*, Cambridge, 2004. London Mathematical Society, Cambridge University Press.

[5] Th. Apel and C. Pester. Quadratic eigenvalue problems in the analysis of cracks in brittle materials. In *Proceedings of the European Congress on Computational Methods in Applied Sciences and Engineering (ECCOMAS)*, Jyväskylä, 2004.

[6] Th. Apel and C. Pester. Clément-type interpolation on spherical domains — interpolation error estimates and application to a posteriori error estimation. *IMA J. Numer. Anal.*, 25(2):310–336, 2005.

[7] Th. Apel, A.-M. Sändig, and S. I. Solov'ev. Computation of 3D vertex singularities for linear elasticity: Error estimates for a finite element method on graded meshes. *Math. Model. Numer. Anal.*, 36:1043–1070, 2002.

[8] R. E. Bank. The efiicient implementation of local mesh refinement algorithms. In *Adaptive computational methods for partial differential equations*, pages 74–81, College Park/Md., 1983. Proc. Workshop College Park/Md. 1983.

[9] P. Benner and H. Faßbender. An implicitly restarted symplectic Lanczos method for the Hamiltonian eigenvalue problem. *Lin. Alg. Appl.*, 263:75–111, 1997.

[10] P. Benner and H. Faßbender. An implicitly restarted symplectic Lanczos method for the symplectic eigenvalue problem. *SIAM J. Matrix Anal. Appl*, 22(3):682–713, 2000.

[11] S. Beuchler. Preconditioning for the $p$-version of the FEM by bilinear elements. Report SFB393/01-17, Preprint-Reihe des SFB393 der Technischen Universität Chemnitz, 2001.

[12] S. Beuchler and A. Meyer nad M. Pester. SPC-PM3AdH v 1.0 - Programmer's Manual. Report SFB393/01-08, Preprint-Reihe des SFB393 der Technischen Universität Chemnitz, 2001. Revised August 2003.

[13] J. W. Demmel, J. R. Gilbert, and X. S. Li. SuperLU Users' Guide. Technical Report LBNL-44289, Lawrence Berkeley National Laboratory, 1999.

[14] A. Dimitrov, F.-G. Buchholz, and E. Schnack. Free surface effects in crack propagation: A theoretical-numerical-experimental correlation. Technical Report Preprint 2002-1, Universität Karlsruhe, 2002.

[15] R. W. Freund. Lanczos-type algorithms for structured non-Hermitian eigenvalue problems. In J. D. Brown, M. T. Chu, D. C. Ellison, and R. J. Plemmons, editors, *Proceedings of the Cornelius Lanczos International Centenary Conference*, pages 243–245. SIAM, 1994.

[16] D. Irony, G. Shklarski, and S. Toledo. Parallel and fully recursive multifrontal supernodal sparse cholesky. In P. M. A. Sloot, editor, *Computational science - ICCS 2002*, Lect. Notes Comput. Sci. 1988, pages 335–344, Berlin: Springer, 2002. Part 2.

[17] Chr. Israel. NETGEN69 – Ein hierarchischer paralleler Netzgenerator. Preprint, Preprint-Reihe der Chemnitzer DFG-Forschungsgruppe "Scientific Parallel Computing", SPC 95_26, 1995.

[18] V. A. Kondrat'ev. Boundary problems for elliptic equations in domains with conical or angular points. *Trans. Moscow Math. Soc.*, 16:227–313, 1967. English translation.

[19] V. A. Kozlov, V. G. Maz'ya, and J. Roßmann. *Spectral problems associated with corner singularities of solutions to elliptic equations.* Mathematical Surveys and Monographs, Volume 85, American Mathematical Society, 2001.

[20] D. Leguillon. Computation of 3d singularities in elasticity. In M. Costabel, M. Dauge, and S. Nicaise, editors, *Boundary value problems and integral equations in nonsmooth domains*, volume 167 of *Lecture notes in pure and applied mathematics*, pages 161–170,

New York, 1995. Marcel Dekker. Proceedings of a conference at CIRM, Luminy, France, May 3-7, 1993.

[21] R. B. Lehoucq, D. C. Sorensen, and C. Yang. *ARPACK Users' Guide: Solution of Large-Scale Eigenvalue Problems with Implicitly Restarted Arnoldi Methods.* SIAM, Philadelphia, 1998.

[22] V. Mehrmann and D. Watkins. Structure-preserving methods for computing eigenpairs of large sparse skew-Hamiltonian/Hamiltonian pencils. *SIAM J. Sci. Comp.*, 22(6):1905–1925, 2001.

[23] V. Mehrmann and D. Watkins. Polynomial eigenvalue problems with Hamiltonian structure. *Electron. Trans. Numer. Anal.*, 13:106–118, 2002.

[24] A. Meyer, B. Döhler, G. Meinel, and M. Pester. *Großdimensionierte Eigenwertprobleme – Teil III: Algorithmen zur simultanen Berechnung einiger Eigenwerte, Elementarmoduln und Serviceprogramme*, 1983. Dokumentation zum Programmpaket.

[25] A. Meyer, B. Döhler, and L. Skurt. Algorithmen für großdimensionierte Eigenwertprobleme. Wiss. Schriftenreihe, Technische Hochschule Karl-Marx-Stadt, 1983.

[26] A. Meyer and C. Pester. The Laplace and the linear elasticity problems near polyhedral corners and associated eigenvalue problems. Preprint SFB393/04-12, Preprint-Reihe des SFB393 der Technischen Universität Chemnitz, 2004.

[27] C. Pester. Hamiltonian eigenvalue symmetry for quadratic operator eigenvalue problems. 2005. To appear in J. Integral Equations Appl.

[28] C. Pester. A residual a posteriori error estimator for the eigenvalue problem for the Laplace-Beltrami operator. Preprint SFB393/05-01, Preprint-Reihe des SFB393 der Technischen Universität Chemnitz, 2005.

[29] M. Pester. Parallelrechner-Bibliotheken. Internet publication, `http://www-user.tu-chemnitz.de/~pester/par_lib.html`, 1996.

[30] J. Rosam. Berechnung der Rissgeometrie bei spröden elastischen Körpern. Diplomarbeit, TU Chemnitz, 2004.

[31] R. Ruhmer. 3D-Finite-Elemente-Rechnungen und ihre Visualisierung mit Hilfe von OpenGL. Diplomarbeit, TU Chemnitz, 2000.

[32] B. Szabó and I. Babuška. *Finite Element Analysis.* John Wiley & Sons, 1991.

[33] D. Watkins. *Fundamentals of Matrix Computations.* John Wiley & Sons, 2002.

[34] D. Watkins. On Hamiltonian and symplectic Lanczos processes. *Lin. Alg. Appl.*, 385:23–45, 2004.

Other titles in the SFB393 series:

02-01 M. Pester. Bibliotheken zur Entwicklung paralleler Algorithmen - Basisroutinen für Kommunikation und Grafik. Januar 2002.

02-02 M. Pester. Visualization Tools for 2D and 3D Finite Element Programs - User's Manual. January 2002.

02-03 H. Harbrecht, M. Konik, R. Schneider. Fully Discrete Wavelet Galerkin Schemes. January 2002.

02-04 G. Kunert. A posteriori error estimation for convection dominated problems on anisotropic meshes. March 2002.

02-05 H. Harbrecht, R. Schneider. Wavelet Galerkin Schemes for 3D-BEM. February 2002.

02-06 W. Dahmen, H. Harbrecht, R. Schneider. Compression Techniques for Boundary Integral Equations - Optimal Complexity Estimates. April 2002.

02-07 S. Grosman. Robust local problem error estimation for a singularly perturbed reaction-diffusion problem on anisotropic finite element meshes. May 2002.

02-08 M. Springmann, M. Kuna. Identifikation schädigungsmechanischer Materialparameter mit Hilfe nichtlinearer Optimierungsverfahren am Beispiel des Rousselier Modells. Mai 2002.

02-09 S. Beuchler, R. Schneider, C. Schwab. Multiresolution weighted norm equivalences and applications. July 2002.

02-10 Ph. Cain, R. A. Römer, M. E. Raikh. Renormalization group approach to energy level statistics at the integer quantum Hall transition. July 2002.

02-11 A. Eilmes, R. A. Römer, M. Schreiber. Localization properties of two interacting particles in a quasiperiodic potential with a metal-insulator transition. July 2002.

02-12 M. L. Ndawana, R. A. Römer, M. Schreiber. Scaling of the Level Compressibility at the Anderson Metal-Insulator Transition. September 2002.

02-13 Ph. Cain, R. A. Römer, M. E. Raikh. Real-space renormalization group approach to the quantum Hall transition. September 2002.

02-14 A. Jellal, E. H. Saidi, H. B. Geyer, R. A. Römer. A Matrix Model for $\nu_{k_1 k_2} = \frac{k_1+k_2}{k_1 k_2}$ Fractional Quantum Hall States. September 2002.

02-15 M. Randrianarivony, G. Brunnett. Parallel implementation of curve reconstruction from noisy samples. August 2002.

02-16 M. Randrianarivony, G. Brunnett. Parallel implementation of surface reconstruction from noisy samples. September 2002.

02-17 M. Morgenstern, J. Klijn, Chr. Meyer, R. A. Römer, R. Wiesendanger. Comparing measured and calculated local density of states in a disordered two-dimensional electron system. September 2002.

02-18 J. Hippold, G. Rünger. Task Pool Teams for Implementing Irregular Algorithms on Clusters of SMPs. October 2002.

02-19 H. Harbrecht, R. Schneider. Wavelets for the fast solution of boundary integral equations. October 2002.

02-20 H. Harbrecht, R. Schneider. Adaptive Wavelet Galerkin BEM. October 2002.

02-21 H. Harbrecht, R. Schneider. Wavelet Galerkin Schemes for Boundary Integral Equations - Implementation and Quadrature. October 2002.

03-01 E. Creusé, G. Kunert, S. Nicaise. A posteriory error estimation for the Stokes problem: Anisotropic and isotropic discretizations. January 2003.

03-02 S. I. Solov'ëv. Existence of the guided modes of an optical fiber. January 2003.

03-03 S. Beuchler. Wavelet preconditioners for the p-version of the FEM. February 2003.

03-04 S. Beuchler. Fast solvers for degenerated problems. February 2003.

03-05 A. Meyer. Stable calculation of the Jacobians for curved triangles. February 2003.

03-06 S. I. Solov'ëv. Eigenvibrations of a plate with elastically attached load. February 2003.

03-07 H. Harbrecht, R. Schneider. Wavelet based fast solution of boundary integral equations. February 2003.

03-08 S. I. Solov'ëv. Preconditioned iterative methods for monotone nonlinear eigenvalue problems. March 2003.

03-09 Th. Apel, N. Düvelmeyer. Transformation of hexahedral finite element meshes into tetrahedral meshes according to quality criteria. May 2003.

03-10 H. Harbrecht, R. Schneider. Biorthogonal wavelet bases for the boundary element method. April 2003.

03-11 T. Zhanlav. Some choices of moments of refinable function and applications. June 2003.

03-12 S. Beuchler. A Dirichlet-Dirichlet DD-pre-conditioner for p-FEM. June 2003.

03-13 Th. Apel, C. Pester. Clément-type interpolation on spherical domains - interpolation error estimates and application to a posteriori error estimation. July 2003.

03-14 S. Beuchler. Multi-level solver for degenerated problems with applications to p-version of the fem. *(Dissertation)* July 2003.

03-15 Th. Apel, S. Nicaise. The inf-sup condition for the Bernardi-Fortin-Raugel element on anisotropic meshes. September 2003.

03-16 G. Kunert, Z. Mghazli, S. Nicaise. A posteriori error estimation for a finite volume discretization on anisotropic meshes. September 2003.

03-17 B. Heinrich, K. Pönitz. Nitsche type mortaring for singularly perturbed reaction-diffusion problems. October 2003.

03-18 S. I. Solov'ëv. Vibrations of plates with masses. November 2003.

03-19 S. I. Solov'ëv. Preconditioned iterative methods for a class of nonlinear eigenvalue problems. November 2003.

03-20 M. Randrianarivony, G. Brunnett, R. Schneider. Tessellation and parametrization of trimmed surfaces. December 2003.

04-01 A. Meyer, F. Rabold, M. Scherzer. Efficient Finite Element Simulation of Crack Propagation. February 2004.

04-02 S. Grosman. The robustness of the hierarchical a posteriori error estimator for reaction-diffusion equation on anisotropic meshes. March 2004.

04-03 A. Bucher, A. Meyer, U.-J. Görke, R. Kreißig. Entwicklung von adaptiven Algorithmen für nichtlineare FEM. April 2004.

04-04 A. Meyer, R. Unger. Projection methods for contact problems in elasticity. April 2004.

04-05 T. Eibner, J. M. Melenk. A local error analysis of the boundary concentrated FEM. May 2004.

04-06 H. Harbrecht, U. Kähler, R. Schneider. Wavelet Galerkin BEM on unstructured meshes. May 2004.

04-07 M. Randrianarivony, G. Brunnett. Necessary and sufficient conditions for the regularity of a planar Coons map. May 2004.

05-01 C. Pester. A residual a posteriori error estimator for the eigenvalue problem for the Laplace-Beltrami operator. January 2005.

05-02 J. Badía, P. Benner, R. Mayo, E. Quintana-Ortí, G. Quintana-Ortí, J. Saak. Parallel Order Reduction via Balanced Truncation for Optimal Cooling of Steel Profiles. February 2005.

The complete list of current and former preprints is available via
`http://www.tu-chemnitz.de/sfb393/preprints.html`.