R. Hiptmair
L. Scarabosio
C. Urzua Torres

Spring Term 2015

Numerical Methods for Partial Differential Equations

ETH Zürich
D-MATH

Homework Problem Sheet 9

Problem 9.1 Midpoint Quadrature on Triangles (Core problem)

Local quadrature is a core operation in any finite element computation, see [NPDE, Section 3.6.4]. This exercise examines a special local quadrature rule, the concept of order and the use of local quadrature for computing element matrices.

The midpoint quadrature rule on the unit triangle is given by

$$\int_{\widehat{K}} f(\widehat{\mathbf{x}})\,\mathrm{d}\widehat{\mathbf{x}} \approx \frac{1}{6}\sum_i f(\mathbf{m}^i),$$

where $\mathbf{m}^i$ are the midpoints of the edges of $\widehat{K}$.

(9.1a) Show that the midpoint quadrature rule is exact for $f \in \mathcal{P}_2(\mathbb{R}^2)$ that is, it is of *order* 3.

HINT: Find a simple basis for $f \in \mathcal{P}_2(\mathbb{R}^2)$ (see [NPDE, Def. 3.4.8]), integrate each basis function and then compare to the value the quadrature rule would give. Since integration is linear, this result will extend to all of $\mathcal{P}_2(\mathbb{R}^2)$. You may use [NPDE, Lemma 3.6.61].

Solution: A basis for $\mathcal{P}_2(\mathbb{R}^2)$ is $\{1, x, x^2, y, y^2, xy\}$. Due to symmetry of the unit triangle, we only have to test the functions $\{1, x, x^2, xy\}$.

$$\int_{\widehat{K}} 1\ \mathrm{d}\widehat{\mathbf{x}} = \frac{1}{2} = \frac{1}{6}(1+1+1),$$
$$\int_{\widehat{K}} x\ \mathrm{d}\widehat{\mathbf{x}} = \frac{1}{6} = \frac{1}{6}\left(\frac{1}{2}+\frac{1}{2}+0\right),$$
$$\int_{\widehat{K}} x^2\ \mathrm{d}\widehat{\mathbf{x}} = \frac{1}{12} = \frac{1}{6}\left(\frac{1}{2^2}+\frac{1}{2^2}+0\right),$$
$$\int_{\widehat{K}} xy\ \mathrm{d}\widehat{\mathbf{x}} = \frac{1}{24} = \frac{1}{6}\left(0+\frac{1}{2^2}+0\right).$$

For x^3, the quadrature formula gives $1/24$, where the correct integral is $1/20$. This shows that the midpoint quadrature rule is of order 3.

(9.1b) Let K be any triangle. Find the element matrix for the bilinear form

$$\mathsf{a}_K(u,v) = \int_K u(\mathbf{x})v(\mathbf{x})\,\mathrm{d}\mathbf{x}$$

using midpoint quadrature, for Lagrangian finite elements of local polynomial degree 1 and 2.

HINT: See [NPDE, Section 3.3.5] and [NPDE, Eq. (3.5.6)] for the basis functions in barycentric coordinates. You can use [NPDE, Lemma 3.6.61]. Assume that the area $|K|$ of the triangle is known.

Solution: For order 1,

$$\mathbf{A} = \frac{|K|}{12}\begin{pmatrix} 2 & 1 & 1 \\ 1 & 2 & 1 \\ 1 & 1 & 2 \end{pmatrix}.$$

For order 2 the quadrature rule will give

$$\mathbf{A} = \frac{|K|}{3}\begin{pmatrix} & & & & & \\ & & & & & \\ & & & & & \\ & & & 1 & & \\ & & & & 1 & \\ & & & & & 1 \end{pmatrix},$$

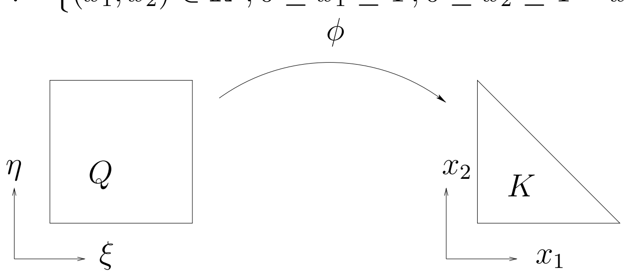
which is of course not correct (the integrands are of order 4). The right answer, according to [NPDE, Lemma 3.6.61] and [NPDE, Eq. (3.5.6)] is

$$\mathbf{A} = \frac{|K|}{180}\begin{pmatrix} 9 & -1 & -1 & -12 & -12 & -8 \\ -1 & 9 & -1 & -12 & -8 & -12 \\ -1 & -1 & 9 & -8 & -12 & -12 \\ -12 & -12 & -8 & 32 & 16 & 16 \\ -12 & -8 & -12 & 16 & 32 & 16 \\ -8 & -12 & -12 & 16 & 16 & 32 \end{pmatrix}.$$

Problem 9.2 The Duffy Trick

In [NPDE, Section 3.6.4.2] we learned that local quadrature rules on the cells of a finite element mesh can be obtained by transformation, see [NPDE, § 3.6.83] for details. In this problem we witness the use of a special transformation to generate quadrature rules on the reference triangle.

Consider the following mapping from the unit square $Q := (0,1)^2$ to the reference element $K := \left\{(x_1, x_2) \in \mathbb{R}^2; 0 \leq x_1 \leq 1, 0 \leq x_2 \leq 1 - x_1\right\}$ defined by



$$x_1 := \xi,$$
$$x_2 := \eta(1-\xi).$$

This degenerate mapping can be used to transform the integration of a function $f : K \mapsto \mathbb{R}$ over the reference element K to an integration over the unit square Q. This approach is known as Duffy trick and can be used to generate quadrature rules of arbitrary order on the reference triangle.

(9.2a) Compute the Jacobian of the mapping $\phi : Q \mapsto K$ and show that the following transformation formula holds true

$$\int_Q f(\xi, \eta(1-\xi))(1-\xi)\, \mathrm{d}\xi\, \mathrm{d}\eta = \int_K f(x_1, x_2)\, \mathrm{d}x_1\, \mathrm{d}x_2 \tag{9.2.1}$$

for an arbitrary function $f : K \mapsto \mathbb{R}$.

Solution: We have

$$\begin{pmatrix} x_1 \\ x_2 \end{pmatrix} = \phi(\xi, \eta) = \begin{pmatrix} \xi \\ \eta(1-\xi) \end{pmatrix}.$$

Therefore, the Jacobian is given by $D_\phi = \begin{pmatrix} 1 & 0 \\ -\eta & 1-\xi \end{pmatrix}$ and the determinant $|D_\phi(\xi, \eta)| = 1 - \xi$. Therefore using substitution we obtain

$$\int_K f(x_1, x_2)\, \mathrm{d}x_1\, \mathrm{d}x_2 = \int_Q f(\xi, \eta(1-\xi))(1-\xi)\, \mathrm{d}\xi\, \mathrm{d}\eta.$$

(9.2b) On the unit square one can use *tensor product Gauss-Legendre* quadrature formulas, which are obtained as follows, see also [NPDE, Ex. 3.6.95]: If $\zeta_\ell \in]0, 1[$, $\omega_\ell \in \mathbb{R}$, $\ell = 1, \ldots, n$, are the nodes and weights, respectively, of the n-point Gauss-Legendre quadrature rule on $[0, 1]$, then we approximate

$$\int_Q g(x_1, x_2)\, \mathrm{d}\boldsymbol{x} \approx \sum_{j=1}^{n} \sum_{\ell=1}^{n} \omega_j \omega_\ell g(\zeta_j, \zeta_\ell)\ . \tag{9.2.2}$$

Show that this quadrature rule is of (maximal) order $2n$.

HINT: The order of a quadrature rule has been introduced in [NPDE, Def. 3.6.87]. It is known that the n-point Gauss-Legendre quadrature rule on an interval is of maximal order $2n$.

Solution: The result is easily obtained from the tensor product structure of the integral:

$$\int_Q x_1^p x_2^q\, \mathrm{d}x_1\, \mathrm{d}x_2 = \int_0^1 x_1^p\, \mathrm{d}x_1 \int_0^1 x_2^q\, \mathrm{d}x_2,$$

for two integers $p, q \geq 0$. Since the one-dimensional quadrature rule is of order $2n$, the integral above is exact for $p, q \leq 2n - 1$ and thus the tensor product quadrature rule is of order $2n$ too.

(9.2c) We consider the quadrature formula on the unit triangle generated by the Duffy trick from the n^2-point tensor product Gauss-Legendre quadrature formula on the unit square presented in subproblem (9.2b). Determine the (maximal) order of this rule.

HINT: According to [NPDE, Def. 3.6.87] you have to check, which integrals $\int_K x_1^p x_2^q\, \mathrm{d}x$ of monomials with $p + q \leq m$ are integrated exactly. [NPDE, Lemma 3.6.61] is useful for that purpose.

Solution: For the one-dimensional integral

$$\int_0^1 x^p\, \mathrm{d}x = \frac{1}{p+1},$$

we know that a n-point Gauss-Legendre rule integrates polynomial up to degree $p \leq 2n-1$ exactly.

From subproblem (9.2b), we know that the tensor product quadrature rule on Q integrates all polynomials up to $p, q \leq 2n-1$ exactly.

On the triangle $K := \left\{(x_1, x_2) \in \mathbb{R}^2; 0 \leq x_1 \leq 1\,, 0 \leq x_2 \leq 1 - x_1\right\}$ we have

$$\begin{aligned}
\int_K x_1^p x_2^q \,\mathrm{d}x_1\,\mathrm{d}x_2 &= \int_Q \xi^p \eta^q (1-\xi)^q (1-\xi)\,\mathrm{d}\xi\,\mathrm{d}\eta \\
&= \int_0^1 \xi^p (1-\xi)^{q+1}\,\mathrm{d}\xi \int_0^1 \eta^q\,\mathrm{d}\eta \\
&= \int_0^1 \xi^p \sum_{k=0}^{q+1} \binom{q+1}{k} 1^q (-\xi)^{q+1-k}\,\mathrm{d}\xi \int_0^1 \eta^q\,\mathrm{d}\eta \\
&= \sum_{k=0}^{q+1} \binom{q+1}{k} (-1)^{q+1-k} \int_0^1 \xi^{p+q+1-k}\,\mathrm{d}\xi \int_0^1 \eta^q\,\mathrm{d}\eta \\
&= \frac{p!\,q!}{(p+q+2)!}.
\end{aligned}$$

The first integral is integrated exactly for $p+q+1-k \leq 2n-1$, $k = 0, \ldots, q+1$ and the second integral for $q \leq 2n-1$. Therefore, we only have exact integration on K for $p+q+1 \leq 2n-1$ and the quadrature rule on K is of order $2n-1$. In other words, using the Duffy trick, the presence of the jacobian of the transformation reduces by 1 the order of the quadrature rule when moving from the square to the triangle.

Problem 9.3 Parametric Finite Elements on Curved Triangles (Core problem)

In [NPDE, Section 3.7.4] we have seen how the paradigm of parametric finite elements can be used to deal with *curved elements*, which is important for higher order resolution of boundaries and interfaces. In this problem we study this technique for "triangular" elements whose edges are pieces of parabolas.

Taking the cue of [NPDE, § 3.7.30], we consider the quadratic transformation

$$\begin{aligned}
\mathbf{\Phi}_{\tilde{K}}(\hat{\boldsymbol{x}}) :=& \boldsymbol{a}^1 \hat{\lambda}_1(\hat{\boldsymbol{x}}) + \boldsymbol{a}^2 \hat{\lambda}_2(\hat{\boldsymbol{x}}) + \boldsymbol{a}^3 \hat{\lambda}_3(\hat{\boldsymbol{x}}) + \\
& \boldsymbol{d}^3 4\hat{\lambda}_1(\hat{\boldsymbol{x}})\hat{\lambda}_2(\hat{\boldsymbol{x}}) + \boldsymbol{d}^1 4\hat{\lambda}_2(\hat{\boldsymbol{x}})\hat{\lambda}_3(\hat{\boldsymbol{x}}) + \boldsymbol{d}^2 4\hat{\lambda}_1(\hat{\boldsymbol{x}})\hat{\lambda}_3(\hat{\boldsymbol{x}})
\end{aligned} \tag{9.3.1}$$

mapping the reference triangle (unit triangle) $\widehat{K}$ to the triangle $\tilde{K}$ with curved edges, as shown in Figure 9.1. Here $\hat{\lambda}_i$, $i = 1, 2, 3$, are the barycentric coordinate functions on $\widehat{K}$, and $\boldsymbol{a}^i$, $\boldsymbol{d}^i$ are given vectors $\in \mathbb{R}^2$.

(9.3a) Determine the location of the vertices of $\tilde{K}$ in terms of $\boldsymbol{a}^i$, $\boldsymbol{d}^i$.

Solution: Due to the properties of the mapping $\mathbf{\Phi}_{\tilde{K}}(\hat{\boldsymbol{x}})$, we have that the vertices $\boldsymbol{a}_i$ of $\tilde{K}$ are

$$\begin{aligned}
\boldsymbol{a}_1 &:= \mathbf{\Phi}_{\tilde{K}}(0,0) = \boldsymbol{a}^1, \\
\boldsymbol{a}_2 &:= \mathbf{\Phi}_{\tilde{K}}(1,0) = \boldsymbol{a}^2, \\
\boldsymbol{a}_3 &:= \mathbf{\Phi}_{\tilde{K}}(0,1) = \boldsymbol{a}^3.
\end{aligned} \tag{9.3.2}$$

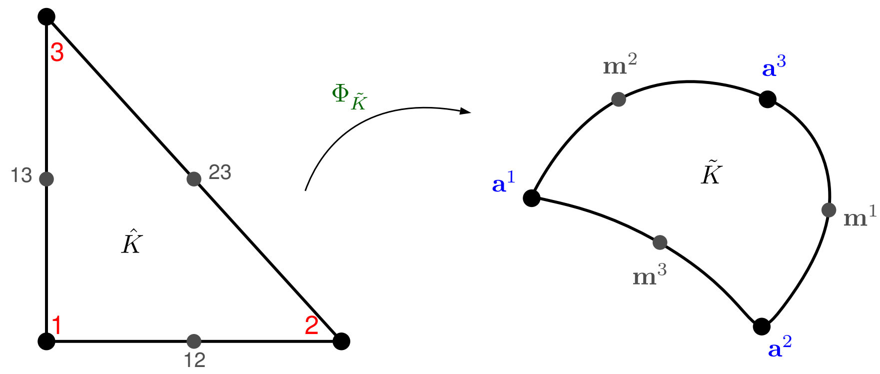


Figure 9.1

(9.3b) Give a geometric interpretation of the vectors $\boldsymbol{d}^i$.

HINT: Rely on the image points of the midpoints of the edges of $\widehat{K}$ and use the notations from Figure 9.1.

Solution: Following the hint we get

$$\begin{aligned}
\boldsymbol{m}_{12} &:= \boldsymbol{\Phi}_{\tilde{K}}(\hat{\boldsymbol{m}_{12}}) = \frac{\boldsymbol{a}^1 + \boldsymbol{a}^2}{2} + \boldsymbol{d}^3, \\
\boldsymbol{m}_{23} &:= \boldsymbol{\Phi}_{\tilde{K}}(\hat{\boldsymbol{m}_{23}}) = \frac{\boldsymbol{a}^2 + \boldsymbol{a}^3}{2} + \boldsymbol{d}^1, \\
\boldsymbol{m}_{13} &:= \boldsymbol{\Phi}_{\tilde{K}}(\hat{\boldsymbol{m}_{13}}) = \frac{\boldsymbol{a}^1 + \boldsymbol{a}^3}{2} + \boldsymbol{d}^2.
\end{aligned} \tag{9.3.3}$$

Where the first terms correspond to the linear part of the transformation and $\boldsymbol{d}^i$ is the offset due to the parabolic contribution.

(9.3c) The mapping $\widetilde{\boldsymbol{\Phi}}_K$ from [NPDE, Eq. (3.7.31)] in [NPDE, § 3.7.30] is a special version of (9.3.1). Determine the corresponding values for $\boldsymbol{a}^i$ and $\boldsymbol{d}^i$ using the local numbering as in [NPDE, Fig. 153].

Solution: From subproblem (9.3a), we already know that $\boldsymbol{a}_1 = \boldsymbol{a}^1$, $\boldsymbol{a}_2 = \boldsymbol{a}^2$ and $\boldsymbol{a}_3 = \boldsymbol{a}^3$ (we remind that the affine transformation from the reference triangle to the triangle in physical space with straight edges maps vertices to vertices and barycentric coordinate functions to barycentric coordinate functions).

Then, in [NPDE, Eq. (3.7.31)], only the edge connecting $\boldsymbol{a}_1$ to $\boldsymbol{a}_2$ is curved, and the midpoint of the edge is displaced by $\delta\boldsymbol{n}$. This means that $\boldsymbol{d}^3 = \delta\boldsymbol{n}$, and $\boldsymbol{d}^1 = \boldsymbol{d}^2 = \boldsymbol{0}$ (because the other two edges are not curved). Alternatively, since $\lambda_1\lambda_2$, $\lambda_2\lambda_3$ and $\lambda_1\lambda_3$ are linearly independent, the same result can be obtained comparing (9.3.1) to [NPDE, Eq. (3.7.31)].

(9.3d) Compute the Jacobian matrix $D\boldsymbol{\Phi}_{\tilde{K}}$.

Solution: First we compute

$$\begin{aligned}
\frac{\partial \boldsymbol{\Phi}_{\tilde{K}}}{\partial \hat{x}_1} &= -\boldsymbol{a}^1 + \boldsymbol{a}^2 + 4\boldsymbol{d}^3(-\hat{\lambda}_2 + \hat{\lambda}_1) + 4\boldsymbol{d}^1\hat{\lambda}_3 + 4\boldsymbol{d}^2(-\hat{\lambda}_3), \\
\frac{\partial \boldsymbol{\Phi}_{\tilde{K}}}{\partial \hat{x}_2} &= -\boldsymbol{a}^1 + \boldsymbol{a}^3 + 4\boldsymbol{d}^3(-\hat{\lambda}_2) + 4\boldsymbol{d}^1\hat{\lambda}_2 + 4\boldsymbol{d}^2(-\hat{\lambda}_3 + \hat{\lambda}_1).
\end{aligned}$$

From where we conclude the Jacobian matrix.

$$D\Phi_{\tilde{K}} = \begin{pmatrix} (\boldsymbol{a}_1^2 - \boldsymbol{a}_1^1) + 4\boldsymbol{m}_1^3(1-2\hat{x}_1) + 4\boldsymbol{c}_1\hat{x}_2 & (\boldsymbol{a}_1^3 - \boldsymbol{a}_1^1) + 4\boldsymbol{m}_1^2(1-2\hat{x}_2) + 4\boldsymbol{c}_1\hat{x}_1 \\ (\boldsymbol{a}_2^2 - \boldsymbol{a}_2^1) + 4\boldsymbol{m}_2^3(1-2\hat{x}_1) + 4\boldsymbol{c}_2\hat{x}_2 & (\boldsymbol{a}_2^3 - \boldsymbol{a}_2^1) + 4\boldsymbol{m}_2^2(1-2\hat{x}_2) + 4\boldsymbol{c}_2\hat{x}_1 \end{pmatrix},$$

where $\boldsymbol{c} := \boldsymbol{m}^1 - \boldsymbol{m}^2 - \boldsymbol{m}^3$.

(9.3e) Compute the determinant of Jacobian matrix $|\det D\Phi_{\tilde{K}}|$.

Solution: As we have a 2×2 matrix

$$\begin{pmatrix} A & B \\ C & D \end{pmatrix},$$

the determinant follows from applying the formula $AD - BC$.

(9.3f) Write a C++ Method

```
template <class Function>
void assemLocQuadTri(std::vector<Coordinate> const & a,
                     std::vector<Coordinate> const & d,
                     Function const& alpha, ElementMatrix & local);
```

which computes the element matrix for

$$\int_\Omega \boldsymbol{\alpha}(\mathbf{x})\, \mathbf{grad}\, \mathbf{u}(\mathbf{x}) \cdot \mathbf{grad}\, \mathbf{v}(\mathbf{x})\, \mathrm{d}\mathbf{x},$$

where $\boldsymbol{\alpha}(\mathbf{x})$ is a scalar coefficient. Use *parametric* piecewise linear Lagrangian finite elements and the (transformed) edge midpoint quadrature rule from Problem 9.1.

The method takes as inputs an arbitrary function alpha(Coordinate const & x), and the vectors a and d (containing $\boldsymbol{a}^i$ and $\boldsymbol{d}^i$ respectively).

Recall Coordinate corresponds to Dune::FieldVector<calc_t,world_dim> and ElementMatrix corresponds to AnalyticalLocalMass::ElementMatrix in your implementation of subproblem (7.4b). You can assume that ElementMatrix local has been initialized and set to zero before it is passed to the method.

HINT: You may in addition use Eigen::Matrix2d for the Jacobian, as it is equipped with the methods inverse() and determinant().

HINT: A template file is available in the lecture `svn` repository

`assignments_codes/assignment9/Problem3`

There you will also find a testcall written in `main.cc` and its output in `testcall_output.txt`

Solution: Applying the transformation techniques learnt in [NPDE, Section 3.7], we get

$$\begin{aligned} a_{ij} &:= \int_{\tilde{K}} \boldsymbol{\alpha}(\mathbf{x})\, \mathbf{grad}\, \mathrm{b}_j(\mathbf{x}) \cdot \mathbf{grad}\, \mathrm{b}_i(\mathbf{x})\, \mathrm{d}\mathbf{x} \\ &= \int_{\hat{K}} ((D\Phi(\hat{\mathbf{x}}))^{-1} \boldsymbol{\alpha}(\Phi(\hat{\mathbf{x}}))(D\Phi(\hat{\mathbf{x}}))^{-T})\, \mathbf{grad}\, \hat{\mathrm{b}}_j \cdot \mathbf{grad}\, \hat{\mathrm{b}}_i \det D\Phi(\hat{\mathbf{x}})\, \mathrm{d}\hat{\mathbf{x}}. \end{aligned}$$

Consequently, by using the midpoint quadrature rule, we get

$$a_{ij} \approx \mathbf{grad}\,\hat{\mathbf{b}}_j \cdot \mathbf{grad}\,\hat{\mathbf{b}}_i \frac{1}{6} \sum_{k=1}^{3} ((D\boldsymbol{\Phi}(\hat{\boldsymbol{m}}^k))^{-1} \boldsymbol{\alpha}(\boldsymbol{m}^k) (D\boldsymbol{\Phi}(\hat{\boldsymbol{m}}^k))^{-T}) \left|\det D\boldsymbol{\Phi}(\hat{\boldsymbol{m}}^k)\right|.$$

See Listing 9.1 for the code.

Listing 9.1: Implementation of `assemLocQuadTri()`

```
template<class ElementMatrix, class Function>
void assemLocQuadTri(std::vector<Coordinate> const& verts,
                std::vector<Coordinate> const& d,
                Function const & alpha, ElementMatrix & local){
  //create jacobian evaluated at midpoints
  Eigen::Matrix3d auxMat;
  std::vector<Eigen::Matrix2d> jacobians(3);
  Coordinate sumverts; sumverts = 0;
  for(int i=0; i<3; ++i){
    sumverts += verts[i];
    if(i<2){
      jacobians[0].row(i) << verts[1][i]-verts[0][i] + 2*(d[0][i] - d[1][i] - d[2][i]),
                            verts[2][i]-verts[0][i] + 2*(d[0][i] - d[1][i] - d[2][i]) ;
      jacobians[1].row(i) << verts[1][i]-verts[0][i] + 2*(d[0][i] - d[1][i] + d[2][i]),
                         verts[2][i]-verts[0][i];
      jacobians[2].row(i) << verts[1][i]-verts[0][i],
                         verts[2][i]-verts[0][i] + 2*(d[0][i] + d[1][i] - d[2][i]) ;
    }
  }
  ///compute barycentric gradients
  Eigen::Matrix<calc_t,2,3> grads;
  grads << -1, 1, 0,
           -1, 0, 1;
  // add contributions of each midpoint
  for(int mp=0; mp<3; ++mp){
    Eigen::Matrix2d invJ = (jacobians[mp]).inverse();
    double detJac = (jacobians[mp]).determinant();
    Coordinate midpoint = ( sumverts - verts[mp]);
    midpoint /=2;
    midpoint += d[mp];
    double coeff = alpha(midpoint);
    for (unsigned i=0;i<3;++i){
      for (unsigned j=0;j<3;++j){
     double val =
        ((invJ*coeff*invJ.transpose())*grads.col(j)).transpose()*grads.col(i);
     local[i][j] += val*detJac/6;
```

```
        }
    }
```

Problem 9.4 Poisson Equation on a Disk

This problem offers a comprehensive treatment of analytical and algorithmic techniques for parametric linear finite elements on general hybrid meshes (with straight edges), see, in particular [NPDE, Section 3.7.2]. It also revisits assembly of Galerkin matrices from element matrices.

We consider the homogeneous Dirichlet problem for the Laplacian Δ:

$$-\Delta u = f \quad \text{in } \Omega, \quad u = 0 \quad \text{on } \partial\Omega, \tag{9.4.1}$$

where Ω is the unit disk

$$\Omega := \left\{ \boldsymbol{x} \in \mathbb{R}^2 \,\middle|\, \|\boldsymbol{x}\| < 1 \right\}.$$

Again, a student does not want to use any finite element library, but prefers to write a code for this boundary value problem from scratch. Warned by his buddy that polar coordinates and a finite difference discretization caused terrible difficulties, he opts for a finite element method on a *hybrid mesh* comprising quadrilaterals and triangles like that depicted in Figure 9.2: the nodes of the mesh are the origin and the intersection points of circles with radii $\frac{j}{N}$, $j = 1, \ldots, N$, with rays at angles $\frac{2\pi j}{N}$, $j = 1, \ldots, N$. He settles for a polygonal approximation of $\partial\Omega$ as in Figure 9.2.

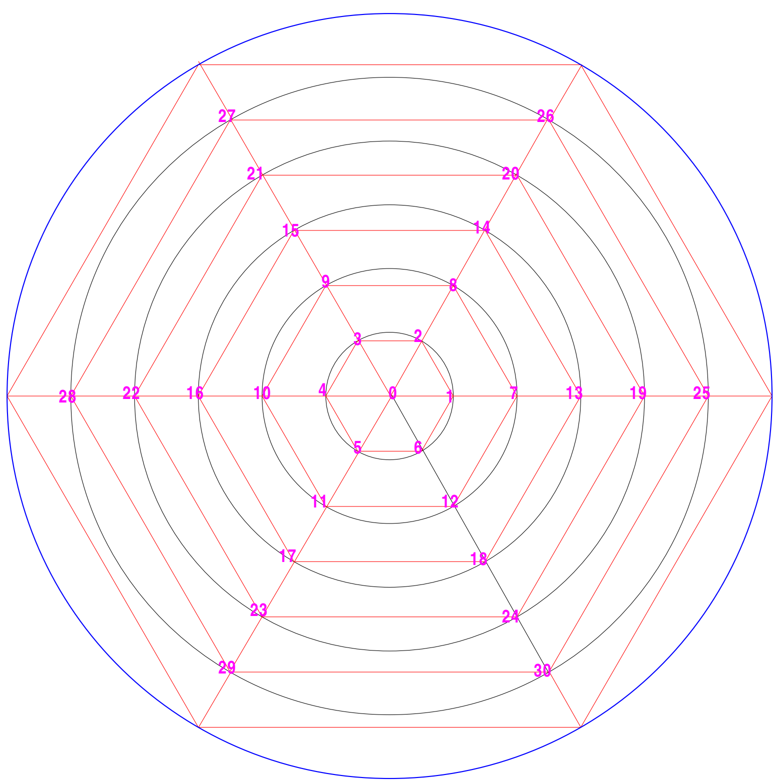


Figure 9.2: Hybrid mesh with piecewise linear approximation of $\partial\Omega$. The red lines are the edges of the mesh.

On the hybrid mesh the student wants to employ a finite element Galerkin discretization of (9.4.1) based on continuous trial/test function that are

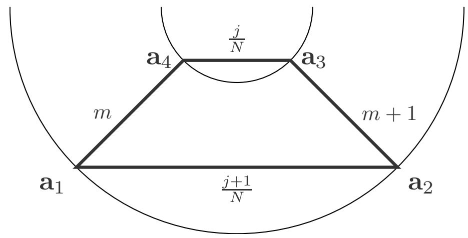


Figure 9.3: quadrilateral element

- piecewise linear on the triangles ($\rightarrow$ [NPDE, Section 3.3])
- parametrically mapped bilinear functions on the quadrilaterals ($\rightarrow$ [NPDE, Section 3.7.2]).

As explained in class, for the quadrilaterals the mapping from the unit square is a bilinear transformation, see [NPDE, Eq. (3.7.16)].

(9.4a) Give the coefficients of the bilinear transformation

$$\mathbf{\Phi}_K(\widehat{\boldsymbol{x}}) = \begin{pmatrix} \alpha_1 + \beta_1 \widehat{x}_1 + \gamma_1 \widehat{x}_2 + \delta_1 \widehat{x}_1 \widehat{x}_2 \\ \alpha_2 + \beta_2 \widehat{x}_1 + \gamma_2 \widehat{x}_2 + \delta_2 \widehat{x}_1 \widehat{x}_2 \end{pmatrix}, \quad \alpha_i, \beta_i, \gamma_i, \delta_i \in \mathbb{R},$$

for the quadrilateral formed by the circles with radii $\frac{j}{N}$ and $\frac{j+1}{N}$, $1 \leq j < N$, and the rays with angles $\frac{2\pi m}{N}$ and $\frac{2\pi(m+1)}{N}$, $0 \leq m < N$, see Figure 9.3.

Solution: Let a_1, a_2, a_3 and a_4 be the vertices of a quadrilateral bounded by the circles with radius j/N and $(j+1)/N$ and the rays with angles $2\pi m/N$ and $2\pi(m+1)/N$ as shown in the figure below.

First, we obtain the (x,y) coordinates of the four vertices by conversion from polar to carthesian coordinates. The values are shown in the table below.

Point	r	θ	x	y
a_1	$\frac{j+1}{N}$	$\frac{2\pi m}{N}$	$\frac{j+1}{N}\cos\frac{2\pi m}{N}$	$\frac{j+1}{N}\sin\frac{2\pi m}{N}$
a_2	$\frac{j+1}{N}$	$\frac{2\pi(m+1)}{N}$	$\frac{j+1}{N}\cos\frac{2\pi(m+1)}{N}$	$\frac{j+1}{N}\sin\frac{2\pi(m+1)}{N}$
a_3	$\frac{j}{N}$	$\frac{2\pi(m+1)}{N}$	$\frac{j}{N}\cos\frac{2\pi(m+1)}{N}$	$\frac{j}{N}\sin\frac{2\pi(m+1)}{N}$
a_4	$\frac{j}{N}$	$\frac{2\pi m}{N}$	$\frac{j}{N}\cos\frac{2\pi m}{N}$	$\frac{j}{N}\sin\frac{2\pi m}{N}$

Next, we find the coefficients $\alpha, \beta, \gamma, \delta \in \mathbb{R}^2$ as follows:

$$\begin{cases} \alpha = a_1 \\ \beta = a_2 - a_1 \\ \gamma = a_4 - a_1 \\ \delta = a_4 - a_3 - a_2 - a_1 \end{cases}$$

Then the bilinear form we are looking for is $\Phi : [0,1]^2 \rightarrow K$,

$$\Phi(\widehat{x}) = \begin{pmatrix} \alpha_1 + \beta_1 \widehat{x}_1 + \gamma_1 \widehat{x}_2 + \delta_1 \widehat{x}_1 \widehat{x}_2 \\ \alpha_2 + \beta_2 \widehat{x}_1 + \gamma_2 \widehat{x}_2 + \delta_2 \widehat{x}_1 \widehat{x}_2 \end{pmatrix}$$

(9.4b) Compute the Jacobian of the bilinear transformation from the subproblem (9.4a) and its determinant. Both are functions on the unit square, see [NPDE, Ex. 3.7.27].

Solution: The Jacobian is

$$D\Phi(\widehat{x}) = \begin{pmatrix} \beta_1 + \delta_1\widehat{x}_2 & \gamma_1 + \delta_1\widehat{x}_1 \\ \beta_2 + \delta_2\widehat{x}_2 & \gamma_2 + \delta_2\widehat{x}_1 \end{pmatrix}$$

and its determinant is

$$|D\Phi(\widehat{x})| = \beta_1\gamma_2 - \beta_2\gamma_1 + (\beta_1\delta_2 - \beta_2\delta_1)\widehat{x}_1 + (\delta_1\gamma_2 - \delta_2\gamma_1)\widehat{x}_2$$

(9.4c) For the computation of the element (load) vectors the student chooses a simple one-point quadrature formula based on the center of gravity (of a triangle or the unit square, respectively). Based on this choice compute the element (load) vector for the general quadrilateral from subproblem (9.4a).

Solution: The quadrature point $c = (c_x, c_y)$ is the center of gravity of the element, taken as the arithmetic mean of the cartesian coordinates of the element. Next, we compute the area of the element using the determinant formula:

$$|K| = \frac{1}{2}\begin{vmatrix} 1 & 1 & 1 \\ x_1 & x_2 & x_3 \\ y_1 & y_2 & y_3 \end{vmatrix}$$

for a triangle element and

$$|K| = \frac{1}{2}\begin{vmatrix} 1 & 1 & 1 \\ x_1 & x_2 & x_3 \\ y_1 & y_2 & y_3 \end{vmatrix} + \frac{1}{2}\begin{vmatrix} 1 & 1 & 1 \\ x_1 & x_3 & x_4 \\ y_1 & y_3 & y_4 \end{vmatrix}$$

for a quadrilateral element (e.g., the sum of the areas of two triangles). Finally, we use the one-point quadrature formula:

$$\int_K f b_N^j \, \mathrm{d}\boldsymbol{x} \approx \frac{|K|}{k} f(x_c, y_c)$$

with $k = 3$ for a triangle element and $k = 4$ for a quadrilateral element.

(9.4d) The same one-point quadrature as in subproblem (9.4c) is used for the evaluation of the element (stiffness) matrices. Compute the element matrix for the quadrilateral considered in subproblem (9.4a).

Solution: We compute the stiffness matrix via the pullback function Φ defined at the first sub-point. Using the relation from [NPDE, Eq. (3.7.25)], we have

$$(A_K)_{ij} = \int_{[0,1]^2} \mathrm{grad}(\widehat{b}^j)\big((D\Phi)^\top D\Phi\big)^{-1}\big(\mathrm{grad}(\widehat{b}^i)\big)^\top |\det D\Phi| \, \mathrm{d}\widehat{x}$$

which we evaluate using a one-point quadrature formula, at the center point $\widehat{x}_c = (0.5, 0.5)$. Besides the values of $D\Phi$ and $\det D\Phi$ at $\widehat{x}_c$ we also need the $\mathrm{grad}(\widehat{b}^j)(\widehat{x}_c), j \in \{1,2,3,4\}$ for the local shape functions: $\widehat{b}^1(\widehat{x}) = (1-\widehat{x}_1)(1-\widehat{x}_2), \widehat{b}^2(\widehat{x}) = \widehat{x}_1(1-\widehat{x}_2), \widehat{b}^3(\widehat{x}) = \widehat{x}_1\widehat{x}_2$ and $\widehat{b}^4(\widehat{x}) = (1-\widehat{x}_1)\widehat{x}_2$.

(9.4e) What is the dimension of the finite element space $V_{0,N} \subset H_0^1(\Omega)$?

Solution: The dimension of the finite element space is: $N(N-1)+1$.

(9.4f) Given the mesh in Figure 9.2 with the finite element space already described, sketch the structure of the Galerkin matrix $\mathbb{A}$. For this, number the nodes counterclockwise from the inner part of the mesh to the exterior as indicated in the figure. Do you observe any pattern?

Solution: The innest ring, conformed by the node 0, just interacts with itself and with the next ring (nodes 1 to 6). The remaining rings interact with the previous ring, with themselves and with the next ring. Moreover, each node interacts with 3 consecutive nodes at each level. This fact combined with our numbering gives rise to the following blockmatrices:

$$
B = \begin{pmatrix}
\times & \times & 0 & 0 & 0 & \times \\
\times & \times & \times & 0 & 0 & 0 \\
0 & \times & \times & \times & 0 & 0 \\
0 & 0 & \times & \times & \times & 0 \\
0 & 0 & 0 & \times & \times & \times \\
\times & 0 & 0 & 0 & \times & \times
\end{pmatrix}
$$

Consequently, the structure of our Galerkin matrix will be:

$$
\begin{array}{cc}
 & \begin{array}{c|ccc|ccc|ccc|ccc|ccc|ccc}
0 & 1 & \cdots & 6 & 7 & \cdots & 12 & 13 & \cdots & 18 & 19 & \cdots & 24 & 25 & \cdots & 30 & 31 & \cdots & 36
\end{array} \\
\begin{array}{c}
0 \\ 1 \\ \vdots \\ 6 \\ 7 \\ \vdots \\ 12 \\ 13 \\ \vdots \\ 18 \\ 19 \\ \vdots \\ 24 \\ 25 \\ \vdots \\ 30 \\ 31 \\ \vdots \\ 36
\end{array} &
\left(\begin{array}{c|ccc|ccc|ccc|ccc|ccc|ccc}
\times & \times & \cdots & \times & 0 & \cdots & 0 & 0 & \cdots & 0 & 0 & \cdots & 0 & 0 & \cdots & 0 & 0 & \cdots & 0 \\ \hline
\times & & & & & & & & & & & & & & & & & & \\
\vdots & & B & & & B & & & 0 & & & 0 & & & 0 & & & 0 & \\
\times & & & & & & & & & & & & & & & & & & \\ \hline
0 & & & & & & & & & & & & & & & & & & \\
\vdots & & B & & & B & & & B & & & 0 & & & 0 & & & 0 & \\
0 & & & & & & & & & & & & & & & & & & \\ \hline
0 & & & & & & & & & & & & & & & & & & \\
\vdots & & 0 & & & B & & & B & & & B & & & 0 & & & 0 & \\
0 & & & & & & & & & & & & & & & & & & \\ \hline
0 & & & & & & & & & & & & & & & & & & \\
\vdots & & 0 & & & 0 & & & B & & & B & & & B & & & 0 & \\
0 & & & & & & & & & & & & & & & & & & \\ \hline
0 & & & & & & & & & & & & & & & & & & \\
\vdots & & 0 & & & 0 & & & 0 & & & B & & & B & & & B & \\
0 & & & & & & & & & & & & & & & & & & \\ \hline
0 & & & & & & & & & & & & & & & & & & \\
\vdots & & 0 & & & 0 & & & 0 & & & 0 & & & B & & & B & \\
0 & & & & & & & & & & & & & & & & & &
\end{array}\right)
\end{array}
$$

Published on 22.04.2015.

To be submitted on 29.04.2015.

References

[NPDE] Lecture Slides for the course "Numerical Methods for Partial Differential Equations".SVN revision # 79326.

[NCSE] Lecture Slides for the course "Numerical Methods for CSE".

Last modified on August 3, 2015