Exercises Computational Mechanics (4MC10) – Lecture 5

Exercise 1

Heat conduction in a static, homogeneous and isotropic medium is governed by the following equation in terms of the temperature T:

$$\lambda \nabla^2 T - c \frac{\partial T}{\partial t} + \Phi(\vec{x}, t) = 0$$

where the Laplacian $\nabla^2 T$ is defined as $\nabla^2 = \vec{\nabla} \cdot \vec{\nabla}$ and $\vec{\nabla}$ is the gradient with respect to the position vector $\vec{x} = x\vec{e}_x + y\vec{e}_y + z\vec{e}_z$ for fixed time t. $\Phi(\vec{x}, t)$ denotes a distributed heat source, whereas λ and c denote the heat conduction coefficient and the specific heat respectively.

a. Show that in a cartesian basis $\{\vec{e}_x, \vec{e}_y, \vec{e}_z\}$ the above equation can be rewritten as

$$\lambda \frac{\partial^2 T}{\partial x^2} + \lambda \frac{\partial^2 T}{\partial y^2} + \lambda \frac{\partial^2 T}{\partial z^2} - c \frac{\partial T}{\partial t} + \Phi(x, y, z, t) = 0$$

b. Is this an ordinary differential equation or a partial differential equation?

c. Is the equation linear or nonlinear?

d. Which order is it?

If the process we are considering is stationary, *i.e.* constant in time, the temperature is solely a function of $\vec{x}$ and the time derivative in the above equation vanishes.

e. Is the remaining equation an ordinary differential equation or a partial differential equation?

If we furthermore assume that there are no temperature variations in the $\vec{e}_y$ and $\vec{e}_z$-directions, the equation can be further reduced.

f. Is the result an ordinary or a partial differential equation?

Before the one-dimensional differential equation can be solved, boundary conditions must be specified. One possible condition is a fixed heat-flow across the boundary of the domain. In one dimension this condition can be expressed as

$$-\lambda \frac{\mathrm{d}T}{\mathrm{d}x} = Q$$

g. How many boundary conditions are needed in one dimension?

h. Why is it insufficient to specify the above condition at both ends of the domain? Hint: try adding a constant to any solution $T(x)$ of the boundary value problem.

i. How can this problem be avoided?

⋆ **Exercise 2**

Consider the quadratic polynomial

$$u^h(x) = a_0 + a_1 x + a_2 x^2$$

with a_0, a_1 and a_2 constants which are in principle known. This function can also be written in the form

$$u^h(x) = N_1(x)u_1 + N_2(x)u_2 + N_3(x)u_3$$

where u_1, u_2 and u_3 are defined as $u_1 = u^h(0)$, $u_2 = u^h(\frac{1}{2}L)$, $u_3 = u^h(L)$ and $N_1(x)$, $N_2(x)$, $N_3(x)$ are shape functions which are yet to be determined.

a. Write down the three relations between the coefficients a_i and u_i which follow from the definitions of the nodal values u_i.

b. Extract from these equations expressions for a_i in terms of u_i.

c. Verify that substituting the expressions for a_i in the polynomial allows to determine the shape functions $N_i(x)$ as

$$\begin{aligned} N_1(x) &= \left(1 - \frac{2x}{L}\right)\left(1 - \frac{x}{L}\right) \\ N_2(x) &= \frac{4x}{L}\left(1 - \frac{x}{L}\right) \\ N_3(x) &= -\frac{x}{L}\left(1 - \frac{2x}{L}\right) \end{aligned}$$

d. Verify that the shape functions satisfy the property $N_i(x_j) = \delta_{ij}$, where x_j denotes the position of node $j = 1, 2, 3$ and δ_{ij} is the Kronecker delta, i.e. $\delta_{ij} = 1$ if $i = j$ and $\delta_{ij} = 0$ otherwise.

e. Do you see a more straightforward way to determine $N_i(x)$? Hint: what are the roots of the above functions, i.e. for which x is $N_i(x) = 0$?

⋆ Exercise 3

Consider a cubic interpolation (*i.e.* a polynomial interpolation of degree three) between nodes at $x_i = \frac{1}{3}(i-1)L$ with $i = 1, 2, 3, 4$. The function values at these nodes are given by

$$\underset{\sim}{u}^{\mathrm{T}} = \begin{bmatrix} u_1 & u_2 & u_3 & u_4 \end{bmatrix} = \begin{bmatrix} 1 & 3 & 3 & 2 \end{bmatrix}$$

a. Determine the interpolation functions $N_i(x)$. Hint: remember your answer to Exercise 2.e.

b. Make a sketch of these functions and verify the properties they must satisfy.

c. Plot the interpolation functions for a value of $L = 4$ using MATLAB; check your sketch against these plots.

d. Plot the function given by the cubic interpolation between the nodal values given above. Verify that it passes through the nodal values.

⋆ Exercise 4

Consider the function

$$u(x) = \frac{5x^2(1 - x^2)}{2 - x}$$

on the domain (0, 1) and a polynomial approximation $u^h(x)$ of this function which is given by

$$u^h(x) = \sum_{i=0}^{p} a_i x^i$$

The MATLAB function `polyapprox` (provided on CANVAS) determines the coefficients a_i such that $u^h(x)$ best fits $u(x)$ in a least-squares sense, i.e. it minimises the error

$$E = \left[\int_0^1 \left(u^h(x) - u(x) \right)^2 \mathrm{d}x \right]^{\frac{1}{2}}$$

The function may be called from the MATLAB prompt as `[E, c] = polyapprox(p)`, where p is the degree of the polynomial; it returns the error E and, as an optional argument, the condition number c of the least-squares matrix.

a. Inspect the file `polyapprox.m` and make sure you understand how it works. In particular, study the structure of the matrix $\underset{\sim}{N}$ used in it.

b. Use the MATLAB function to generate polynomial approximations of several different degrees. For which p is $E \leq 0.01$?

c. Make a table of the error E versus the number of degrees of freedom in the approximation, $n = p+1$, for n ranging from 1 to 20. Plot the values in a diagram, using a logarithmic scale for the error. Does the approximation converge? What is happening at high polynomial degrees?

d. Make a similar table and diagram for the condition number c of the least-squares matrix. How does the trend in this diagram correspond to that in the error?

e. Plot the basis functions x^p of two subsequent degrees, *e.g.* for $p = 10$ and $p = 11$ in the same diagram. Can you explain the observations made above from this comparison?

⋆ Exercise 5

We consider the same function $u(x)$ as in Exercise 4 but we will now use finite element shape functions rather than polynomials to construct the approximations $u^h(x)$:

$$u^h(x) = \sum_{i=1}^{m+1} a_i N_i(x)$$

where $N_i(x)$ are the linear finite element shape functions as given in the lecture notes and m denotes the number of elements. For simplicity the elements are assumed to have a constant length $h = 1/m$.

a. Copy the file `polyapprox.m` to `feapproxlinear.m`. Replace the polynomial shape functions by those of a linear finite element discretisation using $m = 4$ elements by redefining the matrix $\underset{\sim}{N}$.

b. Verify that the coefficients a_i equal the values of $u^h(x)$ in the corresponding nodes.

c. Determine the error E for the discretisation by four elements. How does it compare to that for the polynomial approximation with the same number of degrees of freedom?

d. Adapt the function `feapproxlinear` such that it can deal with any number of elements $m \geq 1$.

e. Make diagrams of the error E and the condition number c versus the number of degrees of freedom n for n ranging from 2 to 20. Compare the results with those obtained in Exercise 4 for the polynomial approximation. Can you explain the differences?

f. Plot the error versus number of degrees of freedom on a log-log scale. What is the rate of convergence of the approximation?

Exercise 6

Still considering the function $u(x)$ as in Exercises 4 and 5, we will increase the degree of the finite element shape functions to two.

a. Copy the file `feapproxlinear.m` to `feapproxquadratic.m`. In this file, implement the shape functions of a quadratic finite element discretisation using an arbitrary number of elements m.

b. Verify that these shape functions equal one in the nodes they are associated with and vanish in the other nodes.

c. Plot the error E versus the number of degrees of freedom n for approximately the same range of n as before. How does the convergence behaviour compare to that of the linear finite element approximation of Exercise 5?

Answers

Exercise 1

b. Partial differential equation
c. Linear
d. Second order
e. Partial differential equation
f. Ordinary differential equation
g. 2
h. This would fix the temperature field only up to an arbitrary constant.
i. By fixing T itself instead of its gradient at at least one end of the domain.

Exercise 2

a. $u_1 = a_0 \qquad u_2 = a_0 + a_1 \frac{1}{2} L + a_2 \frac{1}{4} L^2 \qquad u_2 = a_0 + a_1 L + a_2 L^2$

b. $a_0 = u_1 \qquad a_1 = \dfrac{-3u_1 + 4u_2 - u_3}{L} \qquad a_2 = \dfrac{2u_1 - 4u_2 + 2u_3}{L^2}$

e. Yes, directly from $N_i(x_j) = \delta_{ij}$.

Exercise 3

a.
$$N_1(x) = \left(1 - \frac{3x}{L}\right)\left(1 - \frac{3x}{2L}\right)\left(1 - \frac{x}{L}\right)$$
$$N_2(x) = 9\,\frac{x}{L}\left(1 - \frac{3x}{2L}\right)\left(1 - \frac{x}{L}\right)$$
$$N_3(x) = -\frac{9}{2}\,\frac{x}{L}\left(1 - \frac{3x}{L}\right)\left(1 - \frac{x}{L}\right)$$
$$N_4(x) = \frac{x}{L}\left(1 - \frac{3x}{L}\right)\left(1 - \frac{3x}{2L}\right)$$

Exercise 4

b. $p \geq 4$
c. Initially yes, but the error no longer decreases for increasing n.
d. The condition number increases. As a consequence the coefficients a_i cannot be determined accurately anymore for high n.
e. The basis functions have a very similar shape and it is therefore difficult to distinguish their contributions. The least-squares problem as a result is ill-conditioned.

Exercise 5

c. $E \approx 0.055$. This is approximately 7 times higher than in the polynomial approximation.

e. The shape functions now differ more substantially from each other. As a result the condition number remains low and the coefficients can be determined without any difficulty. On the other hand the approximation error decreases much slower than for the polynomial approximation.
f. $E = O(n^{-2})$

Exercise 6

c. The error in the quadratic approximation is lower and decreases more rapidly than in the linear case. More precisely, $E = O(n^{-3})$.