Dr A. Cookson Department of Mechanical Engineering
2021/22

Formative Assignment 1: Static MATLAB-Based FEM Modelling

ME40064: System Modelling & Simulation

ME50344: Engineering Systems Simulation

You must develop a FEM-based simulation tool to solve the static diffusion-reaction equation in a 1D mesh, which your consultancy firm can re-use for future modelling projects. This tool must be developed following good software development practice and must be validated against standard test cases, accompanied by appropriate documentary proof. Please read this entire document before starting to attempt this assignment.

Q1. Local Element Matrix Functions

Summary

Write MATLAB functions and unit tests to calculate the local element matrics for diffusion and reaction terms.

Task

a. Create a MATLAB function to calculate the local 2-by-2 element matrix for the diffusion operator, for any element, e_N, in the finite element mesh. The inputs to the function are:
   - the diffusion coefficient, D
   - the local element number, e_N
   - the mesh data structure

   You should use the mesh data structure to access each element's Jacobian, J. Generate the mesh data structure using the provided function `OneDimLinearMeshGen.m`.

   Your function should pass the unit test, `FormativeAssignmentUnitTest.m`, which is available on the Moodle page. In this test script you will see a suggested name for your function, as well as the required arguments for your function.

Show

Include the following in your report:
   - a legible screenshot that shows your function passes the test
   - the function's source code

Task

b. Create a MATLAB function to calculate the local 2-by-2 element matrix for the linear reaction operator, for any element, e_N, in the finite element mesh. The inputs to the function are:
   - the reaction coefficient, λ
   - the local element number, e_N
   - the mesh data structure

   Write unit tests for this function. In the code comments briefly explain what the tests are designed to check and why these tests are sufficient to show that your function works correctly.

Show

Include the following in your report:

- a legible screenshot that shows your function passes the test
- the function's source code
- the unit test's source code

Q2. Solving Laplace's Equation using FEM

Summary

Using these functions, now develop a finite element code to solve Laplace's equation, following the code structure given in Lecture 8. Use additional unit tests to verify any new functions that you write, as appropriate, and demonstrate that the complete code is correct using the following analytical test case.

Task

Solve Laplace's equation:

$$\frac{\partial^2 c}{\partial x^2} = 0 \qquad \text{Eq. 1}$$

for the four-element mesh of the domain between x=0 and x=1, shown in Figure 1, for the following boundary conditions:

$$\frac{\partial c}{\partial x}(x = 0) = 2$$

$$c(x = 1) = 0$$

The analytical solution (Eq. 2) for Laplace's equation (Eq. 1) with these boundary conditions is:

$$c = 2(x - 1) \qquad \text{Eq. 2}$$

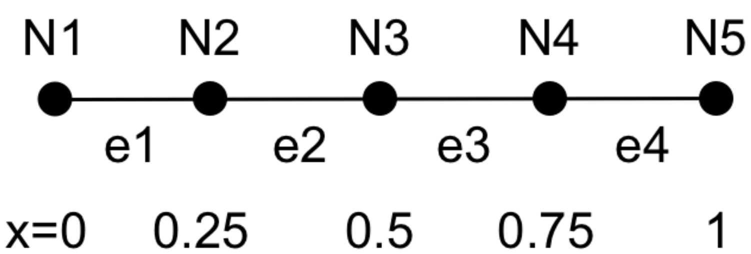

Figure 1: Uniformly sized four-element mesh for the domain x=0 to x=1. Global mesh nodes are numbered N1 to N5.

Show

Include in your report the following information:

- A plot of your code's solution to Eq. 1.
- Comment in the figure caption on the accuracy of your solution and the reason for this
- Your source code used to solve this equation and plot the result

Q3. Verifying your FEM solver for Diffusion-Reaction Equation

Summary

In this task you will extend your code to be able to solve the full steady state diffusion-reaction equation with source terms. You should consider:

- writing unit tests for additional functions
- testing different mesh resolutions & domain sizes, beyond that specified below
- test different boundary conditions and combinations thereof, beyond that specified below
- exploring a range of parameter values to understand the effect of positive and negative source and reaction terms

Task

Demonstrate that your code implements reaction terms correctly, by using it to solve the diffusion-reaction equation:

$$D\frac{\partial^2 c}{\partial x^2} + \lambda c = 0 \qquad \text{Eq. 3}$$

for the following parameter values:

$$D = 1, \lambda = -9$$

and Dirichlet boundary conditions:

$$c(x = 0) = 0$$
$$c(x = 1) = 1.$$

Note, that for these parameters & boundary conditions, Eq. 3 has the following analytical solution:

$$c(x) = \frac{e^3}{e^6 - 1}(e^{3x} - e^{-3x}) \qquad \text{Eq. 4}$$

Show

Include in your report the following information:

- A plot of your code's solution to Eq.3 for meshes with 4, 8, and 16 elements. Include on this plot a line that shows the analytical solution (Eq. 4) to help demonstrate that your solution is accurate.
- Your source code used to solve the equation and generate the plots

Guidance – what to aim for in your submission

Presentation

- Clear graphical presentation of equations, text, diagrams, and plots
    - E.g. axis labels, legible font sizes, no screenshotting of MATLAB plots
- High standard of written English
- Figures should be numbered with a descriptive captions
- Clear layout of report e.g. section headings

Code Quality

- Correctness and elegance
- Good software practice e.g. use of functions, data structures, loops/vectorisation, etc
- Readability – use of meaningful variable names and code commenting

Verification

- Thoroughness of software testing and verification

SUBMISSION GUIDELINES

- This assignment is *formative*, which means it does not count towards your total mark for this unit. HOWEVER, the work you will do on this assignment and the feedback you will be given directly links to what you must complete for the next assignment, which counts for 60% of your unit mark. You are therefore strongly advised to take this opportunity and complete this assignment!

- **You must** include all your MATLAB source code as **text** in the report where indicated so that we can check it runs if necessary:
    - **Do not** paste your code into the document as an OLE item or as an image.
    - **Do not** upload archived/zipped/compressed folders of these source files.

- **Do not** use MATLAB's symbolic algebra toolbox:

  https://uk.mathworks.com/help/symbolic/

  Hint, if you declare a variable in this way: `syms a b c` you are going about things the wrong way.

- You are reminded of the university's policies regarding academic plagiarism: your code, your report, and any figures, must be your own work. You should have passed the Academic Integrity Test before submitting this coursework: https://www.bath.ac.uk/campaigns/academic-integrity-training-and-test/

Submit your work via the online submission link on the unit Moodle page.

Deadline: 4pm, Friday 12th November 2021.