

School of Mathematics and Statistics
MAST90026 Computational Differential Equations
2024

Assignment 1: Boundary value problems

Due: 11AM Wednesday, 17th April.

This assignment is worth 20% of the total assessment in this subject. You should submit copies of MATLAB programs (include all files necessary for the programs to run) and sufficient relevant output online through LMS (You may find the Matlab command `publish` useful!). All hand written working should be scanned and converted to a PDF. Your pdf should also include any tables, figures and comments or explanations of your results.

Where the question asks you to write a function with a given declaration you must follow this exactly. A significant portion of the marks are awarded on whether your code produces the correct output for test problems which I will not provide to you until the assignments have been marked. To help you save time I recommend you modify/reuse code from your previous homework assessments.

All files should be compressed into a single zip file *with your student ID number in the file name*.

1 Self-adjoint form of BVPs

The model equation used during lectures to derive 1D Galerkin methods is

$$-(D(x)u')' + q(x)u = f(x)$$

where $x \in (a, b)$ supplemented by Dirichlet, Neumann or Robin boundary conditions at a and b . We also enforce $D(x) > 0$ and $q(x) \geq 0$, $\forall x \in [a, b]$. Consider now the ODE

$$u'' + \tilde{p}(x)u' + \tilde{q}(x)u = \tilde{f}(x).$$

a. Derive expressions for $D(x)$, $q(x)$ and $f(x)$ in terms of $\tilde{p}(x)$, $\tilde{q}(x)$ and $\tilde{f}(x)$.

b. Using your results in (a), write a function with the following declaration

```
function [D, q, f] = toSelfAdjointForm(x, pt, qt, ft)
```

where \mathbf{x} is a column vector of x values of length $N + 1$ and \mathbf{pt} , \mathbf{qt} and \mathbf{ft} are also column vectors of length $N + 1$ corresponding to the functions $\tilde{p}(x)$, $\tilde{q}(x)$ and $\tilde{f}(x)$ evaluated at the values in \mathbf{x} . The outputs \mathbf{D} , \mathbf{q} and \mathbf{f} should be column vectors of length $N + 1$ containing approximations to the functions $D(x)$, $q(x)$ and $f(x)$ evaluated at the values in \mathbf{x} . For part (b) you only need to submit code, however you should test your code on a case with an analytical result.

2 Coding the spectral collocation method

Consider the BVP

$$u'' + \tilde{p}(x)u' + \tilde{q}(x)u = \tilde{f}(x).$$

subject to mixed BCs $u'(-1) = \alpha, u(1) = \beta$.

Write a MATLAB function with the following declaration

```
function [x,u]=BvpSpec(D,pt,qt,ft,alpha,beta,N)
```

that solves the BVP using spectral collocation at $N + 1$ Chebyshev-Gauss-Lobatto points. **D** is the output of `[D, x] = cheb(N)` and **pt**, **qt** and **ft** are the functions $\tilde{p}(x)$, $\tilde{q}(x)$ and $\tilde{f}(x)$ evaluated on the grid defined by **x**. **alpha** and **beta** are the constants α and β respectively.

Hint: remember to handle the inhomogeneous Dirichlet and Neumann boundary conditions correctly; to impose Neumann boundary condition, you should replace the relevant row of the linear system by the discrete counter part of $u'(-1) = \alpha$.

For this question you only need to submit the code for the two functions. No analysis of results is required.

3 Coding the finite element method

Consider the BVP

$$-(D(x)u')' + q(x)u = f(x),$$

subject to mixed BCs $u'(a) = \alpha, u(b) = \beta$.

Write a MATLAB function with the following declaration

```
function u=BvpFE(node,elem,D,q,f,alpha,beta)
```

that solves such a BVP using the finite element method with linear basis functions. **alpha** and **beta** are the constants α and β respectively, and **D**, **q** and **f** are column vectors of length $N + 1$ containing the values of $D(x)$, $q(x)$ and $f(x)$ evaluated at *node*. Here **node** and **elem** are the data structures for 1D meshes as demonstrated in the lecture notes where **node(1) = a** and **node(N+1) = b**. You should use an appropriate numerical integration technique to compute all integrals considering that **D**, **q** and **f** are only defined at the x values in **node**.

4 Linear BVPs

Consider the BVP

$$-(p + x^4)^2 u'' + 32x^6 u = 12x^2,$$

where $p > 0$, subject to $-u'(-1) = \frac{-4}{(1+p)^2}$ and $u(1) = \frac{1}{1+p}$. The exact solution is $u(x) = \frac{1}{p+x^4}$.

Consider the values $p = 1, 0.1, 0.01$ and solve this problem using both the codes you wrote for Question 2 and Question 3. For the finite element method and a given **node** you can generate **elem** as in the example from the Week 3 Powerpoint. Use the following two different choices for **node**:

- node** are evenly spaced values on $[-1, 1]$.
- node** are the Chebyshev-Gauss-Lobatto points.

Compute the error (use the ∞ -norm) and visualise using a log-log plot with horizontal axis of N , the number of mesh intervals. Discuss the main differences you observe between the finite element method and spectral collocation. Also discuss the differences in the finite element method that you observe when changing **node**.

5 Linear BVPs

Use your code from both Questions 1 and 3 to solve the BVP

$$u'' + \tilde{p}(x)u' + \tilde{q}(x)u = \tilde{f}(x)$$

subject to mixed BCs $u'(-1) = 2, u(1) = 0$ using the finite element method with linear basis functions. The functions \tilde{p} , \tilde{q} and \tilde{f} are defined in the function `Assignment_1_Q5_BVP_Functions.m` which takes `node` as an input and returns the vectors `pt`, `qt` and `ft`. Since no analytical solution is available, verify convergence by refining the mesh and then comparing the solutions over the different meshes. The robust way to compare the different solutions is to use norms on the differences however I am happy for you to verify convergence graphically.

6 Nonlinear BVP

Consider the following BVP, arising as a steady-state finite domain version of Fisher's equation, for $L < 50$,

$$-u'' + u(u - 1) = 0, \quad u(0) = u(L) = 0.$$

This equation has a trivial solution, $u = 0$, and another, non trivial, solution that emerges for larger L (note keep $L < 50$).

Explore this behaviour numerically in the following way.

- Since we can't easily solve on an interval of unknown length, rescale the problem so that you solve on $[0,1]$. L will then appear as a parameter in the equations.
- Solve the nonlinear equations, arising from FD discretization with a uniform mesh, by Newton's method

For all of these problems, you need to provide an initial guess for the solution. I found a constant solution

$$u_0(x) = c \quad x \in [0, 1],$$

for some c to be good enough.

Describe the behaviour of this BVP as L increases.

For this question you should use figures of the solutions to describe the behaviour of the system. You should also provide all the MATLAB code.