ME2 Computing- Session 7: Numerical solution of differential equations: boundary value problems

Learning outcomes:

- Being familiar with the finite difference scheme
- Being familiar with direct and iterative methods
- Being familiar with the types of boundary conditions

Before you start:

In your H drive create a folder *H:\ME2MCP\Session7* and work within it.

ODE with boundary values

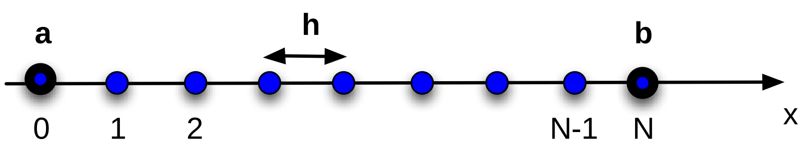
A second order boundary value problem ODE is specified by the ODE itself:

$$\frac{d^2y}{dx^2} + f(x)\frac{dy}{dx} + g(x)y = p(x)$$

by the domain of the solution, i.e. a < x < b, and by the boundary conditions.

The domain of the solution, *[a b]*, is subdivided into N intervals and defined with N+1 points (grid points). The derivatives of the ODEs are approximated at each grid point with the central difference finite difference scheme:

$\frac{dy}{dx} = \frac{y_{i+1} - y_{i-1}}{2h}$	$\frac{d^2y}{dx^2} = \frac{y_{i+1} - 2y_i + y_{i-1}}{h^2}$



The ODE is firstly approximated numerically at the *interior points*, i = *1* to *N-1*, i.e.

$$\frac{y_{i+1} - 2y_i + y_{i-1}}{h^2} + f(x_i)\frac{y_{i+1} - y_{i-1}}{2h} + g(x_i)y_i = p(x_i)$$

Common terms are rearranged together as:

$$\left[\frac{1}{h^2} - \frac{f(x_i)}{2h}\right]y_{i-1} + \left[g(x_i) - \frac{2}{h^2}\right]y_i + \left[\frac{1}{h^2} + \frac{f(x_i)}{2h}\right]y_{i+1} = p(x_i) \quad i = 1 \ldots N-1$$

and the equation rewritten in compact form:

$$a_i y_{i-1} + b_i y_i + c_i y_{i+1} = p_i \quad i = 1 \ldots N-1$$

This procedure generates a set of *N-1* (linear) algebraic equations: every equation, on the left-hand side, is formed by three terms only.

The boundary conditions at the endpoints are then added to assign the constrains on the solution:

$$y_0 = y_a$$
$$y_N = y_b$$

At the end, a set of *N+1* (linear) algebraic equations is formed altogether, with unknown variables $y_0, y_1, y_1, \ldots . y_{N-2}, y_{N-1}, y_N$, i.e.:

$$\begin{bmatrix} 1 & 0 & 0 & 0 & 0 & .. & .. & 0 \\ a_1 & b_1 & c_1 & 0 & 0 & .. & .. & 0 \\ 0 & a_2 & b_2 & c_2 & 0 & .. & .. & 0 \\ 0 & 0 & a_3 & b_3 & c_3 & .. & .. & 0 \\ .. & .. & .. & .. & .. & .. & .. & .. \\ .. & .. & .. & .. & .. & .. & .. & .. \\ 0 & 0 & 0 & 0 & 0 & a_{N-1} & b_{N-1} & c_{N-1} \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} y_0 \\ y_1 \\ y_2 \\ y_3 \\ \\ \\ y_{N-1} \\ y_N \end{bmatrix} = \begin{bmatrix} y_A \\ p_1 \\ p_2 \\ p_3 \\ \\ \\ p_{N-1} \\ y_b \end{bmatrix}$$

These can then be computed either with a direct or an iterative method.
For a *direct* method, to invert the matrix, you can use your algorithm on Gauss elimination from Session2.
For an *iterative* method you can use either Jacobi or Gauss-Siedel method.

Task 1: Direct methods

1. Write a function, *myodebc*, that receives the boudaries of the domain *a* and *b*, the value of the solutions at these points, $y(a) = y_a$ and $y(b) = y_b$, and the number *N* of desired intervals. *myodebc* returns the grid points *x* and the solution *y(x)* at the grid points. The ODE is defined through an external function, *myfunc*, that receives the value of *x* and returns the values of *f(x)*, *g(x)* and *p(x)*.

2. Solve the ODE $\frac{d^2y}{dx^2} + 2x\frac{dy}{dx} + 2y - \cos(3x) = 0$, in the domain [0 π], with boundary conditions $y(0) = 1.5$ and $y(\pi) = 0$. Discretise the domain with 10 intervals. Plot *y(x)*.

3. Repeat the analysis with 100 intervals, and compare with the previous results.

Task 2: Iterative methods

1. Repeat Task 1, and write an equivalent function, *Jacobi*, by using the Jacobi iterative methods. The function Jacobi receives, in addition, as input argument, also the desired accuracy.
   Note that: since the iterative method is explicit, the algorithm might not converge, depending of the ODE.

2. Run the function Jacobi with various values of accuracy.

Task 3: Types of boundary conditions

The boundary conditions, specified at the two boundaries *a* and *b*, can be of different types:

- Dirichlet: two values for the solution, *y(x)*, are specified at *a* and *b*:
  $y(a) = BC_a$ and $y(b) = BC_b$ (this is the case for the example in Task 1)

- Neumann: two values for the derivative of the solution, $\frac{dy}{dx}$, are specified at *a* and *b*:
  $\frac{dy}{dx}(a) = BC_a$ and $\frac{dy}{dx}(b) = BC_b$

- Mixed (or Robin): two values for a combination of the solution, *y(x)*, and its derivative, $\frac{dy}{dx}$, are specified at *a* and *b*:
  $c_0 \frac{dy}{dx}(a) + c_1 y(a) = BC_a$ and $c_2 \frac{dy}{dx}(b) + c_3 y(b) = BC_b$

Note that to constrain the derivative at endpoint *a* it is necessary to implement the finite difference forward scheme, whilst at endpoint *b* the finite difference backward scheme is needed, instead.

Note also that by setting $c_0 = c_2 = 0$ the mixed boundary conditions become of Dirichlet type, and that by setting $c_1 = c_3 = 0$ the mixed boundary conditions become of Neumann type.

1. Modify the function, *myodebc*, to accommodate all the various types of boundary conditions. *myodebc* still receives the boundaries of the domain *a* and *b*, the boundary conditions at these points *BC(a)* and *BC(b)*, and the number *N* of desired intervals. In addition, it receives an array *c* of length 4, with the values of c_0, c_1, c_2, c_3, specifying the type of boundary conditions.

2. Solve the ODE $\frac{d^2y}{dx^2} + x\frac{dy}{dx} + y = Qx$, in the domain [0 2], with boundary conditions $\left.\frac{dy}{dx}\right|_{x=0} = 0$ and $\left.\frac{dy}{dx}\right|_{x=2} = -1$. Discretise the domain with 50 intervals. Plot *y(x)* as a set of parametric curves for value of Q = -5,0,5.

Task 4: Heat transfer in a nuclear fuel rod

The fuel rod of a nuclear reactor is a cylindrical structure with the fuel contained within a metal cladding. The heat is generated by the nuclear reaction in the fuel region and conducted, through the thickness of the cladding, to the outer surface of the cladding. Outside the cladding cooling occurs with flowing water at $T_w = 473K$ through convective heat transfer (heat transfer coefficient $h = 6 \cdot 10^4 \frac{W}{m^2K}$).

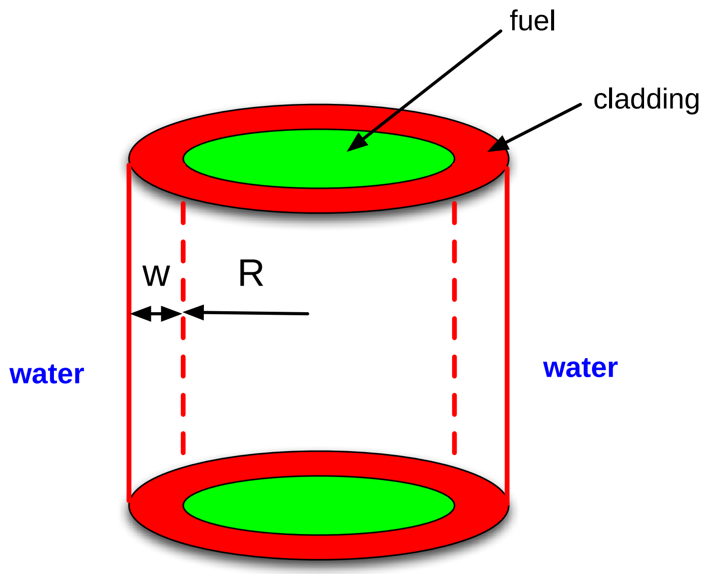
The temperature distribution within the cladding is determined by the ODE:

$$\frac{1}{r}\frac{d}{dr}\left(rk\frac{dT}{dr}\right) = -10^8 \frac{e^{-r/R}}{r}$$

in the region of the cladding $R < r < R + w$, with boundary conditions

$$\left.\frac{dT}{dr}\right|_{r=R} = -\frac{6.32 \cdot 10^5}{k} \text{ and } \left.\frac{dT}{dr}\right|_{r=R+w} = -\frac{h}{k}(T_{r=R+w} - T_w).$$

The thermal conductivity of the metal is $k = 16.75 \frac{W}{mK}$. The dimensions of the rod are: $R = 15mm$ and $w = 3mm$.



1. Compute the temperature distribution within the metal cladding, with N = 50.