Exercises Computational Mechanics (4MC10) – Lecture 4

Exercise 1

We consider a string of unit length which is subjected to a load $f(x)$ per unit length, and which is free at both ends. The tension in the string is normalized to 1. The vertical displacement $u(x)$ of the string is then described by the differential equation

$$Lu := -D^2u = f \qquad \text{on the interval } (0,1) \tag{1}$$

with the Neumann boundary conditions:

$$Bu = g \tag{2}$$

with

$$Bu := \begin{pmatrix} Du(0) \\ Du(1) \end{pmatrix} \quad \text{and} \quad g = \begin{pmatrix} 0 \\ 0 \end{pmatrix} \tag{3}$$

Given a grid with point x_i $(i = 0, 1, 2, \dots, N)$ and step size $h = 1/N$, in conjunction with (1) and (2), we consider the finite-difference scheme and discrete boundary conditions:

$$L^h u^h = f^h \tag{4a}$$

$$B^h u^h = g^h \tag{4b}$$

with

$$(L^h u^h)_i = -\frac{u^h_{i+1} - 2u^h_i + u^h_{i-1}}{h^2} \qquad \text{for } i = 1, 2, \dots, N \tag{5a}$$

$$B^h u^h = \begin{cases} -\dfrac{3u^h_i - 4u^h_{i+1} + u^h_{i+2}}{2h} & \text{for } i = 0 \\[2ex] \dfrac{3u^h_i - 4u^h_{i-1} + u^h_{i-2}}{2h} & \text{for } i = N \end{cases} \tag{5b}$$

and

$$f^h_i = f(x_i) \qquad \text{for } i = 1, 2, \dots, N \tag{6a}$$

$$g^h = \begin{pmatrix} 0 \\ 0 \end{pmatrix} \tag{6b}$$

a. Use the discretization of the boundary conditions in (4b) to eliminate u^h_0 and u^h_N from the system of equations. Complete the corresponding system below.

$$\underbrace{\begin{pmatrix} \frac{2}{3h^2} & \cdot & \cdot & 0 & 0 & 0 & \cdots & 0 & 0 & 0 \\ \cdot & \frac{2}{h^2} & -\frac{1}{h^2} & 0 & 0 & 0 & \cdots & 0 & 0 & 0 \\ 0 & -\frac{1}{h^2} & \cdot & -\frac{1}{h^2} & 0 & 0 & \cdots & 0 & 0 & 0 \\ \vdots & \vdots & & & & & & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & \cdots & \cdots & \cdots & 0 & \cdot & \cdot & \cdot \\ 0 & 0 & 0 & \cdots & \cdots & \cdots & 0 & 0 & -\frac{2}{3h^2} & \cdot \end{pmatrix}}_{A^h} \begin{pmatrix} u^h_1 \\ u^h_2 \\ u^h_3 \\ \vdots \\ u^h_{N-2} \\ u^h_{N-1} \end{pmatrix} = \begin{pmatrix} f(x_1) \\ \cdot \\ \cdot \\ \vdots \\ \cdot \\ f(x_{N-1}) \end{pmatrix} \tag{7}$$

b. Write a `MATLAB` script that generates the matrix A^h in (7) for variable N and step size $h = 1/N$.

c. Use the method of manufactured solutions to verify the correctness of the `MATLAB` implementation. Note that because the boundary conditions are incorporated in the equations in (7), one must use a sample function that complies with the homogeneous Neumann conditions, $Du(0) = Du(1) = 0$ (Why?). Select your own sample function.

d. Consider the matrix A^h for $h = 1/8$ ($N = 8$). What is the condition number of A^h (type `>> cond(A)` in `MATLAB`)? Is the matrix A^h non-singular?

We replace the Neumann conditions at 0 and 1 by homogeneous Dirichlet conditions. In the discretization, this is imposed by means of the condition $B^h u^h = g^h$ with:

$$B^h = \begin{pmatrix} u_0^h \\ u_N^h \end{pmatrix} \quad g^h = \begin{pmatrix} 0 \\ 0 \end{pmatrix} \tag{8}$$

e. Modify the first row and last row of the matrix A^h in the implementation in `MATLAB` in accordance with (8). What is the condition number of A^h for $h = 1/8$?

f. For $h = 1/8$ and b corresponding to a uniform load $f = 1$, determine the solution of the finite-difference approximation by means of `>> u=A\b`.

g. In the simple case that $N = 3$, the matrix A^h reduces to the 2×2 matrix

$$A^h = \frac{1}{h^2}\begin{pmatrix} 2 & -1 \\ -1 & 2 \end{pmatrix} \tag{9}$$

For simplicity, set $h = 1$. Verify that the matrix in (9) is symmetric positive definite. (Hint: $x^T \cdot A^h \cdot x$ is a quadratic polynomial in x_1, viz.,

$$x^T \cdot A^h \cdot x = 2x_1^2 - 2x_1x_2 + 2x_2^2$$

What are its roots?)

h. For simplicity, set $h = 1$. Determine the eigenvalues of A^h in (9).

i. Set $h = 1$ in (9) and determine the condition number of A^h.

j. Verify the eigenvalues and the value of the condition number numerically in `MATLAB`.

Exercise 2

The problem $A \cdot u = b$ can be solved in `MATLAB` by means of the backslash operator, `>> u=A\b`. Alternatively, we can implement our own direct solver or iterative solver.

a. Reconsider the matrix A^h corresponding to the finite-difference approximation of (1) with Dirichlet boundary conditions:

$$A^h = \frac{1}{h^2}\begin{pmatrix} 2 & -1 & 0 & 0 & \cdots & 0 \\ -1 & 2 & -1 & 0 & \cdots & 0 \\ 0 & -1 & 2 & -1 & \cdots & 0 \\ \vdots & \ddots & \ddots & \ddots & \ddots & \vdots \\ 0 & \cdots & 0 & -1 & 2 & -1 \\ 0 & 0 & \cdots & 0 & -1 & 2 \end{pmatrix} \tag{10}$$

for variable N. Select an N, say $N = 64$. Implement the LU decomposition procedure in `MATLAB`. Verify that $LU = A$. Note that the elimination matrix for the k-th column of the Upper factor U can be computed as:

```
M=speye(N-1,N-1);
for i=(k+1):(N-1)
   M(i,k)=-U(i,k)/U(k,k);
end
```

b. Use the LU decomposition to solve $A \cdot u = b$, with $b = (1, 1, \ldots, 1)$. First solve $L \cdot y = b$ by forward substitution. Then solve u from $U \cdot u = y$ by backward substitution. Verify that the solution obtained by the LU procedure and forward/backward substitution is identical to the solution obtained from `>> u=A\b`.

c. An alternative approach is to solve the system $A \cdot u = b$ iteratively, for instance, by means of Gauss-Seidel relaxation. Implement the Gauss-Seidel relaxation method for A^h according to (10) and $b = (1, 1, \ldots, 1)$, with $N = 64$.