Technische Universität Berlin
Institut für Mathematik
Dr. R. Kruse, M. Eisenmann, R. Weiske

WiSe 2017/18

Numerical Mathematics for Engineering II
Assignment 6

Theoretical exercises: submit to exercise class on December 6th or 7th, 2017.
Programming exercises: upload to ISIS until Friday 18:00, December 8th, 2017.

1. Excercise: Periodic boundary conditions **4 points**

Let $f\colon \mathbb{R} \to \mathbb{R}$ be sufficiently smooth and 1-periodic. We want to find 1-periodic solutions $u\colon \mathbb{R} \to \mathbb{R}$ of $-u''(x) = f(x)$, i.e., solutions satisfying $u(x) = u(x+1)$ for all $x \in \mathbb{R}$. Set $h = \frac{1}{N+1}$, $N \in \mathbb{N}$. Using the 3-point stencil, the discrete problem on $[0,1)$ reads

$$L_h u_h = -\frac{1}{h^2}\begin{pmatrix} -2 & 1 & 0 & & 0 & 0 & 1 \\ 1 & -2 & 1 & & 0 & 0 & 0 \\ & & \ddots & \ddots & \ddots & & \\ 0 & 0 & 0 & & 1 & -2 & 1 \\ 1 & 0 & 0 & & 0 & 1 & -2 \end{pmatrix} \begin{pmatrix} u_h(x_0) \\ u_h(x_1) \\ \vdots \\ u_h(x_{N-1}) \\ u_h(x_N) \end{pmatrix} = \begin{pmatrix} f(x_0) \\ f(x_1) \\ \vdots \\ f(x_{N-1}) \\ f(x_N) \end{pmatrix}$$

with L_h being an $(N+1) \times (N+1)$-dimensional matrix.

a) Assume that you have a solution u_h for the discrete or a solution u to the continuous problem. Show that $\bar{u}_h := u_h + C$ or $\bar{u} := u + C$ also are solutions for any $C \in \mathbb{R}$.

b) Find a vector $b \in \mathbb{R}^{N+1}$ such that the extended equation

$$\begin{pmatrix} L_h & b \\ b^T & 0 \end{pmatrix} \begin{pmatrix} u_h \\ \varrho \end{pmatrix} = \begin{pmatrix} f_h \\ 0 \end{pmatrix}$$

ensures solvability as with the pure Neumann problem. In particular, discuss solvability of $L_h u_h = f_h$ for the cases $\varrho = 0$ and $\varrho \neq 0$.

2. Excercise: Two-point boundary value problem **4 points**

Consider the boundary value problem

$$\begin{cases} (2-x^2)u''(x) - xu'(x) + 16u(x) = 0, & x \in (-1,1), \\ u(-1) = -1/2, \quad u'(1) = \beta. \end{cases}$$

Please turn the page!

a) Determine $\beta \in \mathbb{R}$ such that the solution is a fourth order polynomial in x and determine the solution for this β.

b) Determine the reduced linear system $L_h u_h = f_h$ with eliminated boundary conditions for this problem approximating u' with D^0 over a uniform grid with N interior nodes.

3. Programming excercise: Two-point boundary value problem **6 points**

We keep the notations of the previous exercise.

a) Write a function `[Lh,fh] = a06e03getPDE(p,beta)`, which computes the system in exercise 2-b) with $N = 2^p - 1$ and $h = \frac{2}{N+1}$.

b) Write a script `a06e03errors.py` that plots the errors for the β found in exercise 2-a) and $p \in \{2, \dots, 9\}$ in a `loglog` plot.

c) Investigate numerically other choices of β.

4. Programming excercise: The Sylvester equation **8 points**

Inform yourself about the Silvester equation and how to solve it in Python.

a) Use the identity $L_h = -\frac{1}{h^2}(I_N \otimes S + S \otimes I_N)$ and assignment 5 exercise 3-c) to obtain the equivalence

$$L_h u_h = f_h \quad \Longleftrightarrow \quad -\frac{1}{h^2}(U_h S + S U_h) = F_h,$$

where U_h, F_h uses the ordering from assignment 5 exercise 3-c) (all matrices are $N \times N$-dimensional). Hint: $S = S^T$.

b) Write a program `a06e04silsolver(p,i)` which solves again the BVPs from assignment 4 exercise 4 but relies on solving the Sylvester equation. Hint: use a built-in function.

c) Compare the errors, EOCs and runtime with your previously written program `a04e04solve(i)`.

total sum: 22 points