Università della Svizzera italiana | Faculty of Informatics | **Institute of Computational Science ICS**

Introduction to Partial Differential Equations

Fall 2018
M. Multerer and P. Benedusi

Assignment 2

hand in until **Tuesday, 09.10.2018, 16:00**

Problem 1. (Central differences)

Consider the boundary value problem

$$u'(x) = f(x), \quad x \in (0,1), \quad u(0) = \alpha, \quad u(1) = \beta$$

Employ central differences to compute the approximations u_i of $u(x_i)$ for $i = 1, \ldots, n-1$, where $x_i := ih$ and $h := 1/n$. Write down the corresponding linear system of equations $\mathbf{A}\mathbf{u} = \mathbf{f}$. Verify that the matrix $\mathbf{A} \in \mathbb{R}^{(n-1)\times(n-1)}$ is singular for n even.

(5 Points)

Problem 2. (Shortley-Weller approximation)

Let h_W, h_O be positive numbers. Verify the following claim:
For $x \in \mathbb{R}$ und $u \in C^3(\mathbb{R})$ let $u_Z := u(x)$, $u_W := u(x - h_W)$ and $u_O := u(x + h_O)$. Then, it holds

$$u_{xx}(x) = \frac{2}{h_O(h_O + h_W)} u_O - \frac{2}{h_O h_W} u_Z + \frac{2}{h_W(h_O + h_W)} u_W + \mathcal{O}(h),$$

where $h := \max\{h_O, h_W\}$.

(5 Points)

Problem 3. (Mehrstellenverfahren)

Consider the finite difference stencils

$$-\Delta_h u = \frac{1}{6h^2} \begin{bmatrix} -1 & -4 & -1 \\ -4 & 20 & -4 \\ -1 & -4 & -1 \end{bmatrix}_\star u, \qquad R_h f = \frac{1}{6} \begin{bmatrix} & 1/2 & \\ 1/2 & 4 & 1/2 \\ & 1/2 & \end{bmatrix}_\star f.$$

Let $u \in C^6(\overline{\Omega})$ and $f \in C^4(\overline{\Omega})$. Show that there holds

$$\Delta_h u(\mathbf{x}) = \Delta u(\mathbf{x}) + \frac{h^2}{12} \Delta^2 u(\mathbf{x}) + \mathcal{O}(h^4),$$
$$R_h f(\mathbf{x}) = f(\mathbf{x}) + \frac{h^2}{12} \Delta f(\mathbf{x}) + \mathcal{O}(h^4).$$

(5 Points)

Problem 4. (Implementation)

Consider Poisson's problem on the unit square

$$\begin{aligned} -\Delta u(\mathbf{x}) &= f(\mathbf{x}), & \mathbf{x} &\in \Omega := (0,1)^2, \\ u(\mathbf{x}) &= 0, & \mathbf{x} &\in \Gamma. \end{aligned}$$

Employ a finite difference discretisation to solve this equation numerically. To that end, assume an equidistant grid with nodes

$$\mathbf{x}_{(i,j)} = (h \cdot j, h \cdot i) \quad \text{with } (i,j) \in I = \{1, \ldots, n-1\}^2,$$

where $h = 1/n$ and $n \geq 1$.

a) Let

```
h = 1/n; x = (1:n-1)/n; [X,Y] = meshgrid(x);,
```

define a function

```
function F = rhs(@(x,y) f(x,y), X, Y),
```

which evaluates the right hand side f in the nodes $[\mathbf{x}_{i,j}]_{i,j}$ and returns the discretised right hand side.

b) Implement the function

```
function A = fdm2D(n)
```

which assembles the finite difference system matrix in two spatial dimensions.

c) Test your implementation for $f(x, y) = 2\pi^2 \sin(\pi x) \sin(\pi y)$. The analytical solution of this problem is given by $u(x, y) = \sin(\pi x) \sin(\pi y)$. The linear system can be solved by `mldivide` (\ ). Evaluate u on the grid $[\mathbf{x}_{i,j}]_{i,j}$ and plot the error

```
norm(u-u_h,'inf')
```

for decreasing $h = 2^{-i}$ and $i = 1, \dots, 10$. To that end, employ a (`semilogy`) plot with respect to i. How does the error decay?

d) Finally, implement the conjugate gradient method which is given by the following algorithm and solves a linear system up to an error `tol`:

$k = 0$; $\mathbf{d}_0 = \mathbf{r}_0 = \mathbf{f} - \mathbf{A}\mathbf{u}_0$; $\varepsilon = \|\mathbf{r}_0\|_2^2$;
while $\sqrt{\varepsilon} >$ `tol`
  $\mathbf{z} = \mathbf{A}\mathbf{d}_k$;
  $\alpha_k = \varepsilon/(\mathbf{d}_k^\intercal \mathbf{z})$;
  $\mathbf{u}_{k+1} = \mathbf{u}_k + \alpha_k \mathbf{d}_k$;
  $\mathbf{r}_{k+1} = \mathbf{r}_k - \alpha_k \mathbf{z}$;
  $\varepsilon_{\text{old}} = \varepsilon$;
  $\varepsilon = \|\mathbf{r}_{k+1}\|_2^2$;
  $\beta_k = \varepsilon/\varepsilon_{\text{old}}$;
  $\mathbf{d}_{k+1} = \mathbf{r}_{k+1} + \beta_k \mathbf{d}_k$;
  $k = k + 1$;

Write a function

```
function u = conjugateGradient(A, f, u0),
```

which realises the conjugate gradient method numerically. Now, consider the right hand side $f(x, y) = e^{-(x-y)^2}$. Test your implementation of the conjugate gradient method against `mldivide` for $h = 2^{-10}$ and `tol` $= 10^{-8}$. Further, plot the number of iterations in the conjugate gradient method in order to achieve a precision of `tol` $= 10^{-6}$ for decreasing $h = 2^{-i}$ and $i = 1, \dots, 10$.

(10 Points)