Numerical Analysis of Ordinary Differential Equations
Programming Exercises

Summer semester 2016

Universität Heidelberg - IWR
Prof. Dr. Guido Kanschat, Dr. Dörte Beigel,
Stefan Meggendorfer, Philipp Siehr

Exercise sheet 3
Until: 23.06.2016

Exercise P3.1 (Heat Equation)
In this exercise we want to solve the heat equation on the time-space-cylinder $I \times \Omega = [0,1] \times [0,1]$

$$\begin{aligned} \partial_t u(t,x) &= \partial_x^2 u(t,x) \qquad t \geq 0 \\ u(0,x) &= u_0(x). \end{aligned} \tag{P3.1}$$

We consider two cases of initial values:

1. $u_0^c(x) = \sin(\pi x)$,
2. $u_0^{dc}(x) = \begin{cases} 1, & \text{for } x \in (0.3, 0.7] \\ 0, & \text{otherwise.} \end{cases}$

This PDE models the distribution of heat in Ω over time $t \in I$. Imagine a burning candle in the area Ω. The initial distribution of the heat is described with the continious initial value U_0^c. Now the candle gets extinguished, and the heat will be distributed equally (over time) in Ω. This is represented by the solution $U(t,x)$.
Discontinous initial values, such as U_0^{dc}, could represent the case of an opening door between two rooms of different heat.
After discretizing (P3.1) in space (= black magic from PDE-Numerics) we recieve the following linear system of ODEs

$$\begin{aligned} \partial_t U(t) &= AU(t) \qquad t \geq 0 \\ U(0) &= U_0 \end{aligned} \tag{P3.2}$$

with a matrix $A \in \mathbb{R}^{(m+1)\times(m+1)}$ and a solution vector $U(t) \in \mathbb{R}^{m+1}$. Each of these $m+1$ equation represents the quantity of heat $(U(t))_i = u(t, x_i)$ in one point x_i of Ω. The stiffness-matrix A is defined by

$$A = m^2 \cdot \begin{pmatrix} 2 & -1 & & & \\ -1 & 2 & -1 & & \\ & \ddots & \ddots & \ddots & \\ & & -1 & 2 & -1 \\ & & & -1 & 2 \end{pmatrix}$$

We need to discretize the initial values as well:

1. $(U_0^c)_i = \sin(\pi x_i)$.
2. $(U_0^{dc})_i = \begin{cases} 1, & \text{for } x_i \in (0.3, 0.7] \\ 0, & \text{otherwise.} \end{cases}$

For equidistant steps in space, use $x_i = \frac{i-1}{m}$, $i = 1, \ldots, m+1$.

a) Implement the stiffness-matrix A.

b) Implement the Theta-method for linear ODEs.

c) Solve the discretized heat equation (P3.2) for both initial values and $\theta = 1$. Use $m = 100$ equidistant steps in space and $n = 100$ equidistant timesteps.

d) Plot your solution $U(t_i, x)$ in each timestep.
Remark: The solution is 2-dimensional. In Matlab you can use

```
[X,Y]=meshgrid(...)
plot3(X,Y,U)
```

Exercise P3.2 (**Nonlinear ODEs**)

a) Solve the equation

$$\partial_t u(t) = u(t)^2, \qquad t \in [2, 2.5]$$
$$u(2) = 1$$

with the Theta-method, for $\theta = 0, 0.5, 1$ and for the step-sizes $h = \frac{1}{10} \cdot 0.5^i$, $i = 1, \ldots, 5$. Plot your results! What happens when you try to calculate the solution for $t \to 3$?

b) Calculate convergence rates α at $t = 2.5$ for $\theta = 0.5, 0.5001, 1$ with the heuristic formula

$$\alpha = \frac{1}{\log(2)} \log\left(\frac{|y_h - y_{h/2}|}{|y_{h/2} - y_{h/4}|}\right).$$

Choose as step-sizes $h = \frac{1}{10} \cdot 0.5^i$, $i = 1, \ldots, 10$. Explain your observations and compare them with the theoretical results shown in the lecture!

Remark: For calculating the approximate solution with the Theta-method with a nonlinear right-hand-side of the given problem you have to solve a nonlinear equation in each time-step. Therefore you have to use Newton's method:

1. Choose initial value (or vector) x_0

2. While $|f(x_l)| > \text{TOL}$, solve the linear equation

$$f'(x_l) \cdot \delta_l = -f(x_l)$$

(the unknown is the Newton-update δ_l) and calculate x_{l+1} by

$$x_{l+1} = \delta_l + x_l$$

Note that $f'(x) = df(x)/dx$ is the Jacobi-matrix if x is a vector!