Norwegian University of Science and Technology
Department of Mathematical Sciences

TMA4130/35 Matematikk 4N/D Fall 2018

Exercise set 11

1 (J) Implement an adaptive ODE solver by the use of Bogacki–Shampine pair of method, see the list of Runge–Kutta methods on Wikipedia.

Use it to solve the Lotka-Volterra equation with tolerance Tol $= 10^{-3}$. Plot the solutions as well as the stepsizes. Note the number of steps used in this case. Repeat the experiment with Heun–Euler and compare the number of steps used in this case.

Repeat the experiment with Tol$=10^{-5}$ and Tol$=10^{-7}$. Increase the parameter `Max_metodekall` if needed.

2 Given the scalar ODE

$$y'(x) = f(y(x)), \qquad y(x_0) = y_0.$$

A general explicit Runge-Kutta method applied to this problem is given by

$$\begin{aligned} k_1 &= f(y_n), \\ k_2 &= f(y_n + h a_{21} k_1), \\ y_{n+1} &= y_n + h(b_1 k_1 + b_2 k_2). \end{aligned}$$

a) Express the local truncation error of the first step

$$d_1 = y(x_0 + h) - y_1$$

as a power series of h. Which conditions on the coefficients a_{21}, b_1 and b_2 have to be satisfied for the method to be of order 1? And of order 2? Is it possible for this method to be of order 3?

b) Find the optimal choice of parameters (you are supposed to give an argument for in which sense your choice is optimal).

Hint: A similar expansion is done for Heun's method in the note, but the one consider here is simpler, since the function f only depend on $y(x)$.

3 Given the 3th order Runge-Kutta method

$$\begin{aligned} k_1 &= f(x_n, y_n) \\ k_2 &= f\left(x_n + \frac{h}{2}, y_n + \frac{h}{2}k_1\right) \\ k_3 &= f\left(x_n + \frac{3h}{4}, y_n + \frac{3h}{4}k_2\right) \\ y_{n+1} &= y_n + \frac{h}{9}\big(2k_1 + 3k_2 + 4k_3\big). \end{aligned} \tag{1}$$

a) (H) Find the stability function $R(z)$ for this method.
Find also the corresponding stability interval $\mathcal{S}$.

Given a system of ODEs

$$\mathbf{y}' = A\mathbf{y}, \qquad A = \begin{bmatrix} -41 & 38 \\ 19 & -22 \end{bmatrix} \tag{2}$$

b) (H) Find the eigenvalues of A.
When solving the ODE (2) by the method (1), what is the maximum stepsize h you can use and still obtain a stable solution.

c) (J) Implement the method and verify your results numerically. Use $\mathbf{y}(0) = [1,1]^T$ and integrate over the interval [0,5] (for example).

4 **a)** (H) Prove that the implicit midpoint rule

$$y_{n+1} = y_n + hf\left(x_n + \frac{h}{2}, \frac{1}{2}(y_n + y_{n+1})\right)$$

is $A(0)$-stable.

b) (J) Implement the method for problems

$$\mathbf{y}' = A\mathbf{y} + \mathbf{g}(x),$$

(see the implementation of implisitt Euler in the jupyter note). Test your implementation on the problem (2), and show that it is possible to use large stepsizes without the numerical solution being unstable.

5 When solving an ordinary differential equation

$$y' = f(x, y)$$

by a method of high order, like the 4th order classical Runge-Kutta methods (see Exercise 10), a good approximation to the solution can be obtained even for quite large stepsizes, see Figure 1. What is lost, however, is the information about the solutions between the points. We will for instance loose information about the the top of y_1 between $x = 6.0$ and$x = 6.5$ in the figure above.

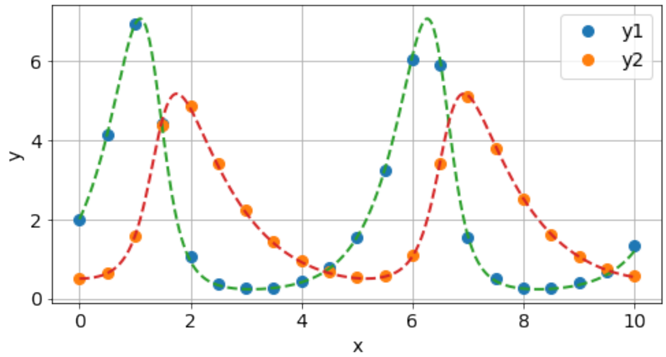

Figure 1: The Lotka–Volterra equation, where • represents the solution from RK4 with $h = 0.5$ and $--$ the exact solution.

How to find approximations to the solutions between the steps? We can of course draw straigh lines between the points (which is usually done), but we can do better!

For simplicity of notation, let us assume that we have a scalar equation. The argument works equally well for each component in a system of equations.

Assume that we have found the solutions at two adjacents points (x_n, y_n) and (x_{n+1}, y_{n+1}). But then we also know the derivatives $y_n' = f(x_n, y_n)$ and $y_{n+1}' = f(x_{n+1}, y_{n+1})$ (by using RK4, this is the same as $k_1$for step n and $n+1$ respectively, so no extra function evaluations are required).

We will then search for a polynomial $p(\theta) \approx y(x_n + \theta h)$, using $x = x_n + \theta h$ for $0 \leq \theta \leq 1$. And since we now have 4 pieces of information, the value of the function and the derivative in two points, we search for a polynomial of degree 3.

a) (H) Find the polynomial

$$p(\theta) = a\theta^3 + b\theta^2 + c\theta + d$$

satisfying

$$\begin{aligned} p(0) &= y_n, & p(1) &= y_{n+1}, \\ \frac{dp}{d\theta}(0) &= hy_n', & \frac{dp}{d\theta}(1) &= hy_{n+1}'. \end{aligned}$$

b) From the result in **a)**, find $p(\theta)$ for

$$y_n = 6.03, \qquad y_{n+1} = 5.91, \qquad y_n' = 5.41, \qquad y_{n+1}' = -8.29, \qquad h = 0.5.$$

(These are taken from the solution y_1 in Figure 1, at $x_n = 6$ and $x_{n=1} = 6.5$.)

c) (J) Plot the solution, and compare roughly with the solution of y_1 for $6 \leq x \leq 6.5$ in Figure 1.