Norwegian University of Science and Technology
Department of Mathematics

MA2501 Numerical Methods
Spring 2017

Exercise set 11

1 Consider the initial value problem

$$x' = \sin(t^2 + x),$$
$$x(0) = 0.$$

a) Use Euler's method with a step size of $h = 1/4$ in order to obtain an approximation of $x(2)$.

b) Use the improved Euler method (Heun's second order method) with a step size of $h = 1/2$ in order to obtain an approximation of $x(2)$.

c) Use the classical Runge–Kutta method with a step size of $h = 1$ in order to obtain an approximation of $x(2)$.

2 Assume that the function $f\colon \mathbb{R} \times \mathbb{R} \to \mathbb{R}$ is continuous and decreasing in its second component. That is,

$$f(t, x) \le f(t, z) \qquad \text{whenever} \qquad x \ge z.$$

Show that the implicit Euler method for the solution of the differential equation

$$x' = f(t, x),$$
$$x(t_0) = x_0,$$

is well-defined. That is, regardless of the step-size $h > 0$, the (non-linear) equation that has to be solved in each iteration has a unique solution.

3 Compute three steps with step size $h = 1$ for the numerical solution of the differential equation

$$x' = -3x - e^x,$$
$$x(0) = 1,$$

using:

a) Euler's method.

b) The improved Euler method.

c) The implicit Euler method. Numerically solve the non-linear equations you obtain by performing two steps of Newton's method in each step of the implicit Euler method.

$\boxed{4}$ The third order Adams–Bashforth method has the form

$$\begin{aligned} x_{k+1} &= x_k + h\Big(\frac{23}{12} f_k - \frac{16}{12} f_{k-1} + \frac{5}{12} f_{k-2}\Big), \\ t_{k+1} &= t_k + h, \\ f_{k+1} &= f(t_{k+1}, x_{k+1}). \end{aligned}$$

Show that this method can be derived by interpolating the function $\tau \mapsto f(\tau, x(\tau))$ in the points t_{k-2}, t_{k-1}, and t_k and then integrating the resulting quadratic polynomial.

$\boxed{5}$ Consider the differential equation

$$\begin{aligned} x' &= x - \frac{x^2}{2}, \\ x(0) &= 1. \end{aligned}$$

Compute three steps with step size $h = 1$ using:

a) The second order Adams–Bashforth method, which for an autonomous ODE is defined by,

$$x_{k+1} = x_k + h\Big(\frac{3}{2} f_k - \frac{1}{2} f_{k-1}\Big),$$

where $f_k \equiv f(x_k)$.

b) The second order Adams–Moulton method, which for an autonomous ODE is (implicitly) defined by

$$x_{k+1} = x_k + h\Big(\frac{1}{2} f_{k+1} + \frac{1}{2} f_k\Big).$$

c) The second order Adams–Bashforth–Moulton method.

In all three cases, use the improved Euler method for computing the iterate x_1.

$\boxed{6}$ Consider the second order initial value problem

$$\begin{aligned} x'' &= -\sin(x) + x', \\ x(0) &= 0, \\ x'(0) &= 1. \end{aligned}$$

a) Rewrite the second order equation as a system of first order equations.

b) Compute two steps of the classical Runge–Kutte method with step size $h = 1/2$ in order to obtain an approximation of $x(1)$.

$\boxed{7}$ **a)** Determine an autonomous system of first order differential equations with accompanying initial conditions written in vector form as

$$\begin{cases} \mathbf{X}' = \mathbf{F}(\mathbf{X}), \\ \mathbf{X}(a) = \mathbf{S}. \end{cases}$$

for the following system

$$\begin{cases} x'' = \sqrt{\frac{6}{1+x^2}} - \frac{1}{2+y^2} + \sin t - 2\cos(x'y'') \\ y''' = -\sqrt{\frac{4}{1+x^2}} - \frac{1}{1+y^4} + \cos t + \sin(x^2 y') \\ x(2) = 2, \quad x'(2) = 1, \quad y(2) = -1, \quad y'(2) = 0, \quad y''(2) = 3. \end{cases}$$

b) Write down one step of the Explicit Euler method and the classical fourth order Runge-Kutta method for this system (C&K p. 314).

c) Implement both methods in the previous part in MATLAB, and use them to approximately solve the system from $t = 2$ to $t = 10$ with $h = 0.1$ and $h = 0.01$. Then, find an "exact" solution by using the built in solver `ode45` in MATLAB with very low error tolerances. Finally, plot the approximate solutions of y from both methods along with the "exact" solution of y and compare the results.

Note: `ode45` requires that the function on the right hand side of the differential equation is of a nonautonomous type, i.e. $f(t, x)$. Use `odeset` to set 'RelTol' to 10^{-12} and 'AbsTol' to 10^{-15} for the 'exact' solution. See the documentation of `ode45` in MATLAB for further details on how to use it.