Assignment 11 - TMA4135

Håvard Hjelmeseth 10.11.2020

1 a)

```python
def fourthOrderRK(f, x_n, y_n, h):
    k1 = f(x_n, y_n)
    k2 = f(x_n + h/2, y_n + (h*k1)/2)
    k3 = f(x_n + h/2, y_n + (h*k2)/2)
    k4 = f(x_n + h, y_n + (h*k3))
    y_next = y_n + (h/6)*(k1 + 2*k2 + 2*k3 + k4)
    x_next = x_n + h
    return x_next, y_next
```

This is my python implementation of the classical 4th order Runge-Kutta method.

b)

We verify numerically the order of convergence, using the example problem

$$y' = -2xy \quad , \quad y(0) = 1$$

and the code from the jupyter notebook "oving10_python_files.ipynb"

```python
def ode_solver(f, x0, xend, y0, h, method=euler):
    # Generic solver for ODEs
    #    y' = f(x,y), y(a)=y0
    # Input: f, the integration interval x0 and xend,
    #         the stepsize h and the method of choice.
    #
    # Output: Arrays with the x- and the corresponding y-values.

    # Initializing:
    y_num = array([y0])    # Array for the solution y
    x_num = array([x0])    # Array for the x-values

    xn = x0                # Running values for x and y
    yn = y0

    # Main loop
    while xn < xend - 1.e-10:             # Buffer for truncation errors
        xn, yn = method(f, xn, yn, h)    # Do one step by the method of choice

        # Extend the arrays for x and y
        y_num = concatenate((y_num, array([yn])))
        x_num = append(x_num,xn)

    return x_num, y_num
```

```python
# Test the order of a method, given a test equation with exact solution
def f1(x, y):
    return -2*x*y

def y_eksakt(x):
    return exp(-x**2)

h = 0.1
x0, xend= 0, 1
y0 = 1

print('h           error          order \n--------------------------------')

for n in range(5):
    x_num, y_num = ode_solver(f1, x0, xend, y0, h, method = fourthOrderRK)
    error = norm(y_eksakt(xend)-y_num[-1])       # Error in the end point
    if n == 0:
        order = NaN          # Nothing to compare
    else:
        order = log2(error_old/error)      # Calculate  the order p
    print(format('{:.3e}   {:.3e}   {:7.2f}'.format( h, error, order)))
    h = 0.5*h            # Reduce the stepsize
    error_old = error
```

```
h            error          order
----------------------------------
1.000e-01    1.625e-06        nan
5.000e-02    1.025e-07       3.99
2.500e-02    6.407e-09       4.00
1.250e-02    3.999e-10       4.00
6.250e-03    2.497e-11       4.00
```

We see that the convergence is of order $p=4$ ∎

$\boxed{2}$

We consider the mass-spring system

$$m_1 u'' = -k_1 u + k_2(v-u)$$
$$m_2 v'' = -k_2(v-u)$$

with initial conditions:

$$u(0) = a \quad , \qquad u'(0) = b$$
$$v(0) = c \quad , \qquad v'(0) = d$$

a)

We substitute the following

$$u = x_1$$
$$u' = x_2$$
$$v = x_3$$
$$v' = x_4$$

Giving us:

$$m_1 u'' = -k_1 u + k_2(v-u)$$
$$m_1 u'' = -k_1 x_1 + k_2(y_1 - x_1)$$
$$u'' = \frac{-k_1 x_1 + k_2(x_3 - x_1)}{m_1}$$

and

$$m_2 v'' = -k_2(v-u)$$
$$m_2 V'' = -k_2(X_3 - X_1)$$
$$V'' = \frac{-k_2(x_3 - x_1)}{m_2}$$

Now since $u = x_1$ and $u' = x_2$

$$\Rightarrow x_1' = (x_1)' = (u)' = u' = x^2$$

$$\Rightarrow x_2' = (x_2)' = (u')' = u''$$

And similarly for v we have $x_3 = v$ and $x_4 = v'$

$$\Rightarrow x_3' = (x_3)' = (v)' = v' = v^2$$

$$\Rightarrow x_4' = (x_4)' = (v')' = v''$$

We now have 4 first order ODE's :

$$(1) \quad x_1' = x_2$$

$$(2) \quad x_2' = \frac{-k_1 x_1 + k_2 (x_3 - x_1)}{m_1}$$

$$(3) \quad x_3' = x_4$$

$$(4) \quad x_4' = \frac{-k_2 (x_3 - x_1)}{m_2}$$

With initial conditions:

$x_1(0) = a$, $x_2(0) = b$
$x_3(0) = c$, $x_4(0) = d$

b)

Performing one step of Heun's method with step length $h = 0.1$, using the following parameters:

$k_1 = 100$ $\quad k_2 = 200$ $\quad m_1 = 10$ $\quad m_2 = 5$

$a = 0$ $\quad b = 1$ $\quad c = 0$ $\quad d = 1$

$$y_0 = \begin{pmatrix} a \\ b \\ c \\ d \end{pmatrix} = \begin{pmatrix} 0 \\ 1 \\ 0 \\ 1 \end{pmatrix}$$

$$y' = \begin{pmatrix} x_1' = x_2 \\ x_2' = \dfrac{-k_1 x_1 + k_2 (x_3 - x_1)}{m_1} \\ x_3' = x_4 \\ x_4' = \dfrac{-k_2 (x_3 - x_1)}{m_2} \end{pmatrix} = f(x_1, x_2, x_3, x_4)$$

$$\tilde{y}_{i+1} = y_i + h f(x_{1i}, x_{2i}, x_{3i}, x_{4i})$$

$$f(x_{10}, x_{20}, x_{30}, x_{40}) = \begin{pmatrix} x_2(0) \\ \frac{1}{m_1}\left(-k_1 x_1(0) + k_2 (x_3(0) - x_1(0))\right) \\ x_4(0) \\ \frac{1}{m_2}\left(-k_2 (x_3(0) - x_1(0))\right) \end{pmatrix} = \begin{pmatrix} 1 \\ 0 \\ 1 \\ 0 \end{pmatrix}$$

$$\hat{y}_1 = y_0 + h f(x_{10}, x_{20}, x_{30}, x_{40})$$

$$= \begin{pmatrix} 0 \\ 1 \\ 0 \\ 1 \end{pmatrix} + 0.1 \begin{pmatrix} 1 \\ 0 \\ 1 \\ 0 \end{pmatrix} = \begin{pmatrix} 0.1 \\ 1 \\ 0.1 \\ 1 \end{pmatrix}$$

$$y_1 = y_0 + \frac{h}{2}\left(f(x_{10}, x_{20}, x_{30}, x_{40}) + f(x_{11}, x_{21}, x_{31}, x_{41}) \right)$$

$$= \begin{pmatrix} 0 \\ 1 \\ 0 \\ 1 \end{pmatrix} + \frac{0.1}{2} \left[\begin{pmatrix} 1 \\ 0 \\ 1 \\ 0 \end{pmatrix} + \begin{pmatrix} 0.1 \\ 1.0 \\ 0.1 \\ 1.0 \end{pmatrix} \right]$$

$$= \begin{pmatrix} 0 \\ 1 \\ 0 \\ 1 \end{pmatrix} + 0.05 \begin{pmatrix} 1.1 \\ 1.0 \\ 1.1 \\ 1.0 \end{pmatrix}$$

$$= \underline{\underline{\begin{pmatrix} 0.055 \\ 1.050 \\ 0.055 \\ 1.050 \end{pmatrix}}}$$