Norwegian University of Science and Technology
Institutt for matematiske fag

TMA4130 Mathematics 4N
TMA413 Mathematics 4D
Høst 2021

Exercise set 6

- Date: **Oct 6, 2021**
- Submission deadline: **Oct 20, 2021**

[2]:
```python
import matplotlib
matplotlib.rcParams.update({'font.size': 12})
import matplotlib.pyplot as plt
import numpy as np
```

In this exercise set you will be analyzing and implementing the following explicit Runge-Kutta methods:

Midpoint rule

$$\begin{array}{c|cc} 0 & 0 & 0 \\ \frac{1}{2} & \frac{1}{2} & 0 \\ \hline & 0 & 1 \end{array}$$

Gottlieb & Gottlieb's 3-stage Runge-Kutta (SSPRK3)

$$\begin{array}{c|ccc} 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 \\ \frac{1}{2} & \frac{1}{4} & \frac{1}{4} & 0 \\ \hline & \frac{1}{6} & \frac{1}{6} & \frac{2}{3} \end{array}$$

1 Convergence orders

Calculate analytically the convergence order's of the two methods. Use the order's conditions given in the lectures.

2 Implementing and testing the methods

In this exercise we will numerically solve the ODE

$$y'(t) = f(y), \quad y(0) = y_0$$

in the interval $t \in [0, T]$.

a) Implement two `Python` functions `explicit_mid_point_rule` and `ssprk3` which implement the Runge-Kutta methods from Exercise 1. Each solver function should take as arguments:

- The initial value y_0
- The inital time t_0
- The final time T

- The right-hand side f
- The maximum number of time-steps N_{max}

The function should return two arrays:

- One array ts containing all the time-points
  $0 = t_0, t_1, ..., t_N = T$
- One array ys containing all the function values
  $y_0, y_1, ..., y_N$

Test the methods on the ODE

$$y'(t) = -y(t), \quad y(0) = 1, \quad t \in [0, 10].$$

Hint: Use the code for `explicit_euler` in the lecture notes or use the supporting material e.g. `Heun` in IntroductionNuMeODE, and modify it to each required method.

b) We will now numerically investigate the RK-methods. We can do this since we know what the exact solution to the ODE above is. We assume that the error $e = |y(T) - y_N|$ when using step size τ is approximately

$$e \approx C\tau^p$$

for some $C > 0$ and p. Note that p is what we call the convergence order. We assume that p and C is the same when using different step sizes h. Let e_1 and e_2 be the errors when using step sizes h_1 and h_2. Then we have

$$\frac{e_1}{e_2} \approx \frac{\tau_1^p}{\tau_2^p} = \left(\frac{\tau_1}{\tau_2}\right)^p.$$

Taking logarithms on both sides we get

$$\log(e_1/e_2) \approx p \log(\tau_1/\tau_2)$$

or

$$p \approx \frac{\log(e_1/e_2)}{\log(\tau_1/\tau_2)}. \tag{1}$$

The value on the right-hand side of this equation is what we call the Experimental Order of Convergence, or EOC. We will now try to estimate the order of convergence using EOC-values.

Do the following for each method: 1. For m=0,…,5, set $\tau_m = 2^{-m}$ and find the value of N_{max} for each m. 2. Find the numerical solution $y_{N(m)}$ of the ODE at $T = 10$. 2. Calculate the error $e_m = |y(10) - y_{N_{max,m}}|$. 3. Calculate the EOC for neighbouring step sizes, that is using equation (1) above with e_m and e_{m+1} for $m = 0, ..., 4$. This should give you 5 different approximations.

Draw a conclusion about the order of convergence p for each method. Does it agree with the result in exercise 1?

c) We will finally test both methods on the ODE

$$y'(t) = -2ty(t), \qquad y(0) = 1, \quad t \in [0, 0.5].$$

This has exact solution e^{-t^2}. Find the approximate value of $y(0.5)$ using

- The midpoint method with $N_{max} = 3$
- The SSPRK3 method with $N_{max} = 2$

The number of step sizes are chosen such that each method needs to perform 6 evaluations of the function f. How do the errors $e = |y(T) - y_{N_{max}}|$ compare?

Additional exercise: Does this observation holds for other values of T? For instance, with $T = 0.2$ or $T = 0.8$. Can you tell what happens to y or y' in $T = 0.5$?

3 SIR Model

The SIR model is a system of first order ODE's which model the dynamics of a disease in a society.

There are $S(t)$ susceptible/healthy individuals, $I(t)$ infected individuals and $R(t)$ recovered individuals. Each susceptible person has a risk of becoming infected, a risk which is proportional to the number of infected people $I(t)$, with a proportinality constant $\beta > 0$. Each infected person also has a chance of recovering, with a recovery constant $\gamma > 0$. This leads to the coupled system of first-order ODE's

$$\begin{aligned} S'(t) =& -\beta S(t)I(t), \\ I'(t) =& \beta S(t)I(t) - \gamma I(t), \\ R'(t) =& \gamma I(t). \end{aligned}$$

We can rewrite this in vector form as

$$\mathbf{u}'(t) = \mathbf{f}(\mathbf{u}(t))$$

where we have defined

$$\mathbf{u}(t) = \begin{pmatrix} S(t) \\ I(t) \\ R(t) \end{pmatrix}, \quad \mathbf{f}(\mathbf{u}(t)) = \begin{pmatrix} -\beta S(t)I(t) \\ \beta S(t)I(t) - \gamma I(t) \\ \gamma I(t) \end{pmatrix}.$$

a) Show that the system is conservative, that is that the total number of individuals $S(t) + I(t) + R(t)$ is constant.

Hint: remember which is the derivative of a constant function.

b) Numerically solve the system

$$\mathbf{u}'(t) = \mathbf{f}(\mathbf{u}(t)), \quad \mathbf{u}(0) = \mathbf{u}_0, \quad t \in [0, T]$$

with either of the RK-methods above.

- You may pick end-time T and N_{max},
- Choose an initial number of individuals.
  - A suitable inital conditions could e.g. be $\mathbf{u}_0 = (50, 10, 0)^T$.
- Plot the solution as a function of time.
- Also plot the total number of individuals.
  - Is the total number conserved? To check this, you might calculate the maximum total and the minimum total over the interval.
- The parameters $\beta = 0.2$ and $\gamma = 0.15$ can be helpful for ilustration.

If you need a guide for this problem, you can look at the Lotka-Volterra model. It was presented in the lectures and is also available in the learning material on the wiki-page.

Additional exercise: modify the SIR model such that the population is not longer constant. One idea is to have a proportion of the infected population to die with rate δ. Test the new model as before.