# Homework 1 — Implicit RK

Numerical Solution for ODEs

*Due date:* November 23rd, 2024

Support files for this homework can be found as a ZIP file on the webpage.

**Exercise 1.** Write a MATLAB implementation of *one* of the following Implicit Runge-Kutta methods:

**RadauI2**

$$\begin{array}{c|cc} 0 & \frac{1}{4} & -\frac{1}{4} \\ \frac{2}{3} & \frac{1}{4} & \frac{5}{12} \\ \hline & \frac{1}{4} & \frac{3}{4} \end{array}$$

**RadauII2**

$$\begin{array}{c|cc} \frac{1}{3} & \frac{5}{12} & -\frac{1}{12} \\ 1 & \frac{3}{4} & \frac{1}{4} \\ \hline & \frac{3}{4} & \frac{1}{4} \end{array}$$

**Lobatto3**

$$\begin{array}{c|ccc} 0 & 0 & 0 & 0 \\ \frac{1}{2} & \frac{5}{24} & \frac{1}{3} & -\frac{1}{24} \\ 1 & \frac{1}{6} & \frac{2}{3} & \frac{1}{6} \\ \hline & \frac{1}{6} & \frac{2}{3} & \frac{1}{6} \end{array}$$

**Lobatto3B**

$$\begin{array}{c|ccc} 0 & \frac{1}{6} & -\frac{1}{6} & 0 \\ \frac{1}{2} & \frac{1}{6} & \frac{1}{3} & 0 \\ 1 & \frac{1}{6} & \frac{5}{6} & 0 \\ \hline & \frac{1}{6} & \frac{2}{3} & \frac{1}{6} \end{array}$$

**Lobatto3C**

$$\begin{array}{c|ccc} 0 & \frac{1}{6} & -\frac{1}{3} & \frac{1}{6} \\ \frac{1}{2} & \frac{1}{6} & \frac{5}{12} & -\frac{1}{12} \\ 1 & \frac{1}{6} & \frac{2}{3} & \frac{1}{6} \\ \hline & \frac{1}{6} & \frac{2}{3} & \frac{1}{6} \end{array}$$

Initial templates for these methods (`radauI2.m`, `radauII2.m` `lobatto3.m`, `lobatto3b.m` and `lobatto3C.m`) can be found in the support files.

**Exercise 2.** Test your script on the following problems from the support files:

1. `lin1p.m` for $t \in [0, 2]$, $x_0 = 2$, $\tau = 0.04$ and plot $t$ versus the solution $x$:

```
x0=2.0; h=0.04;
figure;
[t,x]=feval(method, @lin1p,0,2, x0, h);
plot(t,x,'bo',t,x,'b');
```

2. `lin2.m` for $t \in [0, 0.1]$, $\boldsymbol{x}_0 = (2, 1)^\top$, $\tau = 0.001$ and plot $t$ versus the solution $x_1$:

```
figure;
x0 = [2;1]; h = 1e-3;
[t,x]=feval(method, @lin2, 0,.1, x0, h);
plot(t,x(:,1),'b');
```

3. `sat_ode.m` for $t \in [0, 6.19216933131963970674]$,

$$\boldsymbol{x}_0 = (1.2, 0, 0, -1.04935750983031990726)^\top,$$

$\tau = 0.001$ and $x_1$ versus $x_2$:

```
figure
x0 = [1.2; 0; 0; -1.04935750983031990726]; h = 1e-3;
[t,x] = feval(method, @sat_ode, 0, 6.19216933131963970674, x0, h);
plot(x(:,1), x(:,2));
```

Save these plots as a PDF using `Save > Save As`. A function called `test_problems.m` is included in the support files, which performs the above operations when passed the name of the implicit Runge-Kutta method to run:

```
test_problems(@lobatto3);
```

**Exercise 3.** Estimate the order of the method by linear regression. See `conv_analysis.m` for a script to perform this, when called with the name of the implicit Runge-Kutta method:

```
conv_analysis(@lobatto3);
```

# Submission

Submit the MATLAB script for the implemented method from *exercise 1*, the PDF files of the plots from *exercise 2*, and enter the order of the method deduced in *exercise 3* via the *Study Group Roster* (*Záznamník učitele*) in SIS before the deadline.