Technical University of Denmark

02685 Scientific Computing for Differential Equations 2017

Assignment 1

Authors:
Miguel Suau de Castro (s161333)
Michal Baumgartner (s161636)

March 16, 2017

Contents

1 The Test Problem and DOPRI54

In this first section we are going to implement a set of numerical methods for solving ordinary differential equations. Since the algorithms are only approximations to the real solution, we shall also test their accuracy and discuss their performance by comparing the results obtained when solving the two following initial value problems:

EQUATIONS

As a first approach, we are going to implement the Explicit Euler's method. The algorithm makes use of finite different methods to replace the derivatives in the differential equation. The independent variable is discretized and the solution is computed based on cosequtive approximations to the real function values.

TALK ABOUT STEP LENGTH

EQUATION FORWARD EULER

Instead of using the previous iterate one could also look at future values to approximate a solution. This method is called backward or implicit Euler:

EQUATION BACKWARD EULER

However, for some problems the solution of the previous equation may require the use of numerical solvers, and thus the algorithm becomes computationally more demanding than the explicit Euler's method. We shall see in the next section the advantange of using this method.

Besides, the trapezodial method can be seen as a combination of both methods:

DESCRIBE TRAPEZOIDAL

Figure **??** shows the solution of the two initial value problems given by explicit, implicit Euler and trapezoidal, along with the true solution. It is easy to see, especially in the graph on the right, that, since we base the solution at one point on previous approximations, the further the points are from the initial value the more inaccurate they become and the greater the distance to the true solution is. This distance is called global error, whilst the error made in every iteration is known as local error.

2 Question 3

Write up the order conditions for an embedded Runge-Kutta method with 3 stages. The solution you advance must have order 3 and the embedded method used for error estimation must have order 2.

$$
\begin{array}{c|ccc}
0 & 0 & 0 & 0 \\
c_2 & a_{21} & 0 & 0 \\
c_3 & a_{31} & a_{32} & 0 \\
\hline
x & b_1 & b_2 & b_3 \\
\widehat{x} & \widehat{b}_1 & \widehat{b}_2 & \widehat{b}_3 \\
\hline
e & d_1 & d_2 & d_3
\end{array}
$$

Table 1: Butcher's tableau schema for Runge-Kutta method with 3 stages.

Order conditions derived from Butcher's table:

$$b^T e = 1 \qquad b_1 + b_2 + b_3 = 1 \tag{1a}$$

$$b^T C e = \frac{1}{2} \qquad \underbrace{b_1 c_1}_{0} + b_2 c_2 + b_3 c_3 = \frac{1}{2} \tag{1b}$$

$$b^T C^2 e = \frac{1}{3} \qquad \underbrace{b_1 c_1^2}_{0} + b_2 c_2^2 + b_3 c_3^2 = \frac{1}{3} \tag{1c}$$

$$b^T A C e = \frac{1}{6} \qquad \underbrace{b_2 a_{21} c_1}_{0} + \underbrace{b_3 a_{31} c_1}_{0} + b_3 a_{32} c_2 = \frac{1}{6} \tag{1d}$$

values of c_2 and c_3 will be set to $\frac{1}{4}$ and 1 respecively. This leaves us with 6 unknown variables (3 as and 3 bs) and only 4 equations so we will add the so called consistency equations.

$$c_2 = a_{21} \tag{1e}$$

$$c_3 = a_{31} + a_{32} \tag{1f}$$

Using Matlab to solve the system we get the following results: $b_1 = -\frac{1}{6}$, $b_2 = \frac{8}{9}$, $b_3 = \frac{5}{18}$, $a_{21} = \frac{1}{4}$, $a_{31} = -\frac{7}{5}$, $a_{32} = \frac{12}{5}$.

Next we will solve the system where c_2 and c_3 are known thus giving 2 equations with 3 unknowns. In order to find a solution $\widehat{b}_2$ is set to be $\frac{1}{2}$.

$$\widehat{b}_1 + \widehat{b}_2 + \widehat{b}_3 = 1 \tag{2a}$$

$$\widehat{b}_2 c_2 + \widehat{b}_3 c_3 = \frac{1}{2} \tag{2b}$$

The above system yields $\widehat{b}_1 = \frac{1}{8}$ and $\widehat{b}_3 = \frac{3}{8}$. Going back to the Butcher's tableau we know that last row (d_1, d_2, d_3) is just the difference of the previous two rows by definition.

0	0	0	0
1/4	1/4	0	0
1	-7/5	12/5	0
x	-1/6	8/9	5/18
$\widehat{x}$	1/8	1/2	3/8

Table 2: Butcher's tableau with error estimators for our method.