Numerical Analysis Part II

Coursework

Thulasi Mylvaganam

Spring Term 2019

Department of Aeronautics, Imperial College London

This coursework consists of one exercise and involves programming in MATLAB. **You must answer each question using the word template provided.** Ensure that all figures are clear, labelled correctly and that you strictly adhere to any word limits stated in the template. Do not alter the template in any way: textboxes should not be changed and the written responses must be in "Calibri", fontsize 12 (handwritten responses are not accepted). Figures should be completely contained inside the textboxes.

Your coursework must be submitted as a PDF (*i.e.* not as a word file) on blackboard.

Notation:

- Given a vector v, its derivative with respect to an independent variable t is denoted by v'. Namely
$$\frac{\mathrm{d}v}{\mathrm{d}t} = v' \, .$$
- Given a vector v, it's transpose is denoted by $v^\top$. Similarly, given a matrix M, its transpose is denoted by $M^\top$.

Exercise 1 (Lorenz Attractor)
The Lorenz attractor is a three dimensional dynamical system, described by the ordinary differential equation (ODE)

$$\begin{aligned} \frac{\mathrm{d}x}{\mathrm{d}t} &= \sigma(y-x), \\ \frac{\mathrm{d}y}{\mathrm{d}t} &= x(\rho - z) - y, \\ \frac{\mathrm{d}z}{\mathrm{d}t} &= xy - \beta z\,, \end{aligned} \tag{1}$$

where σ, ρ and β are constant (scalar) parameters. Define the vector

$$Y = \begin{bmatrix} Y_1 \\ Y_2 \\ Y_3 \end{bmatrix} = \begin{bmatrix} x \\ y \\ z \end{bmatrix},$$

and note that

$$\frac{\mathrm{d}Y}{\mathrm{d}t} = Y' = \begin{bmatrix} \sigma(Y_2 - Y_1) \\ Y_1(\rho - Y_3) - Y_2 \\ Y_1 Y_2 - \beta Y_3 \end{bmatrix} = f(Y)\,. \tag{2}$$

In this exercise you are asked to obtain numerical solutions of this ODE for various parameter values, using two different numerical methods. The initial condition is taken to be the same for all parts of this exercise. In particular, we consider the initial condition

$$Y(0) = \begin{bmatrix} 1 & 1 & 1 \end{bmatrix}^\top .$$

Whenever you are asked to plot a numerical solution, provide **two** figures. In the first figure, plot in three different subplots (use “subplot” in MATLAB) the time histories of x, y and z, *i.e.* plot x vs. t (first subplot), y vs. t (second subplot) and z vs. t (third subplot). In the second figure, create a 3D plot of the trajectory $(x(t), y(t), z(t))$ for the entire interval of t for which you are asked to obtain a numerical solution. Make sure that all axes are clearly and correctly labelled.

1. Linearising $f(Y)$ about $Y = \begin{bmatrix} 0 & 0 & 0 \end{bmatrix}^\top$, we can approximate $f(Y)$ by

   $$f(Y) \approx AY\,,$$

   where A is a constant matrix. Compute the matrix A in terms of σ, β and ρ.

   [10%]

2. Consider the case in which $\rho = 0.5$, $\sigma = 10$ and $\beta = \frac{8}{3}$. We wish to obtain a numerical solution for $t \in [0, 8]$.

(a) Following a linear analysis (about $Y = \begin{bmatrix} 0 & 0 & 0 \end{bmatrix}^\top$), what is the maximum stepsize h_{max} we can use to integrate the ODE using explicit Euler?

[5%]

(b) Obtain and plot (in red) a numerical solution for the (nonlinear) ODE (2) using explicit Euler with step size $h = 0.01$.

[10%]

(c) Obtain and plot (in blue, on the same figures used for part 2(b)) a solution of the (nonlinear) ODE (2) using 4th-order Runge-Kutta with the step size $h = 0.01$.

[10%]

(d) Why are the two numerical solutions obtained in part 2(b) and part 2(c) not identical and which numerical solution is "most correct"?

[15%]

3. Consider now the case in which $\rho = 28$, $\sigma = 10$ and $\beta = \frac{8}{3}$. For this selection of parameters, the system of ODEs (2) is a *chaotic* system. This essentially means that that the behaviour of the system is extremely sensitive to initial conditions: a tiny perturbation of the initial condition may lead to a widely different behaviour in the long-term. We wish to obtain a numerical solution for $t \in [0, 50]$.

(a) Obtain and plot (in red) a numerical solution for the (nonlinear) ODE (2) using explicit Euler with step size $h = 0.00001$.

[10%]

(b) Obtain and plot (in blue, on the same figures used for part 3(a)) a solution of the (nonlinear) ODE (2) using 4th-order Runge-Kutta with the step size $h = 0.01$.

[10%]

(c) Are the two numerical solutions obtained in 3(a) and 3(b) similar to one another? Why are the two numerical solutions similar/not similar?

[30%]