MCS 471 Project Five : Numerical Solving of ODEs with Maple

In this project we use Maple to study the numerical solution of a system of ordinary differential equations. We examine the numerical stability of the solvers and the numerical conditioning of the problem.

1. A Chaotic System

The system we study are the famous Lorentz equations:

$$\left\{ \begin{array}{rcl} y_1' & = & u(y_2 - y_1) \\ y_2' & = & vy_1 - y_2 - y_1y_3 \\ y_3' & = & y_1y_2 - wy_3 \end{array} \right.$$

The system depends on three parameters: (u, v, w). The variables in the system are functions of time t. As this is an initial value problem, we need a position to start at, given by the triplet (a, b, c). Thus at $t = 0$, we have $y_1(0) = a$, $y_2(0) = b$, and $y_3(0) = c$.

Throughout the project we take the following values for the parameters: $u = 10$, $v = 28$, $w = \frac{8}{3}$, and initial conditions: $a = 0$, $b = 1$, and $c = 0$.

2. A Numerical Solution with Maple

The default numerical solver in Maple is the Runge-Kutta-Fehlberg method. The solution is returned in the form of a Maple procedure:

```
> sol := dsolve(ivp,numeric);
```

where `ivp` contains the equations and initial conditions for the initial value problem. The solution procedure can be seen as a function of time t. Its evaluation returns four numbers: the value for t and values for all three components. The creation of separate functions for the solution trajectories goes as follows:

```
> y1f := t -> rhs(sol(t)[2]):
> y2f := t -> rhs(sol(t)[3]):
> y3f := t -> rhs(sol(t)[4]):
```

3. Some Plots

A glamorous aspect of the system (and of the project) is the visualization of the solution trajectories. Below are some commands to produce interesting plots:

```
> butterfly := [y1f,0,y3f]:
> bp := plots[spacecurve](butterfly,0..100,numpoints=800):
> plots[display](bp,orientation=[90,90]);
> curve := [y1f,y2f,y3f]:
> plots[spacecurve](curve,0.4..16,numpoints=100);
> plots[spacecurve](curve,5..60,numpoints=350);
```

It takes some patience to generate the plots and they are not strictly necessary for the project.

4. The Assignments

As with any numerical solving procedure, we are interested in the numerical stability and the numerical conditioning.

4.1 Numerical Stability. While we can operate the numerical solver in Maple as a blackbox, we have some control over the accuracy:

```
> sp1 := dsolve(ivp,numeric,abserr=1.0e-7,relerr=1.0e-7);
```

If we play with the tolerances for the absolute and relative error, we start to notice some difference in the answers. The natural question then to ask (and the one we will try to answer) is then: how many decimal places in the answer are correct?

One obvious way is to compute the solution trajectories with different tolerances and compare the anwers at various time steps, e.g., for t = 10, 25, 50, and 100. Another way is to compare the results of the default method with the results of other methods, such as dverk78:

```
> sp2 := dsolve(ivp,numeric,method=dverk78);
```

Also here – similar to above – we can adjust the tolerances. However, be aware that the numerical solvers in Maple use hardware floats and are thus limited to the standard double floating precision.

Make tables comparing the results at several time steps, for various tolerances, using at least two different methods. From those tables, conclude how many decimal places in the answers are correct.

4.2 Numerical Conditioning. The system has three parameters: (u,v,w). In addition to studying the effect of perturbations on these parameters, we will investigate what happens if we modify the initial conditions, given by the triplet (a,b,c).

4.2.1 Perturbations of the parameters of the system. We perturb the parameters as follows:

```
> randomize(): Digits := 16:
> u := evalf(u + 1.0e-4*rand()/10^12):
> v := evalf(v + 1.0e-4*rand()/10^12):
> w := evalf(w + 1.0e-4*rand()/10^12):
```

What is the effect of a perturbation of a magnitude 1.0e-4 on the solution trajectories? Take perturbations of magnitude 1.0e-8 and 1.0e-12 and show the difference in the solution trajectories at times t = 10,25,50, and 100. Display your results in a table.

4.2.2 Perturbations of the initial conditions. The perturbations on the initial conditions are similar, but first we restore the original values for u, v, and w:

```
> u := 10: v := 28: w := 8/3:
> a := evalf(a + 1.0e-4*rand()/10^12):
> b := evalf(b + 1.0e-4*rand()/10^12):
> c := evalf(c + 1.0e-4*rand()/10^12):
```

We are interested in the same questions. What is the effect of a perturbation of a magnitude 1.0e-4 on the solution trajectories? Take perturbations of magnitude 1.0e-8 and 1.0e-12 and show the difference in the solution trajectories at times t = 10,25,50, and 100. Display your results in a table.

5. The deadline is Wednesday 24 April, at 1PM

Bring to class the answers to the project. Besides tables with numerical data, write meaningful sentences to state your conclusions.

Additionally, but not instead of the answers above, you can hand in the print out of the Maple worksheet with your experiments. In this way you can still get some (partial) credit for your (partially) wrong answers. Write complete sentences without spelling and grammatical mistakes.

See http://www.math.uic.edu/~jan/MCS471.html for the hypertext version of this project. In particular, to avoid typing errors and to save time, you may want to download the Maple worksheet.

If you have questions, comments, or difficulties, feel free to come to my office for help.