Nonlinear Problems in applications (MTH311) — Coursework 2

This is the first of two courseworks which combine for 100% of the credits for this unit. The maximum number of marks for this coursework is 65, worth 65% of the credits. This coursework consists of **two** questions. Parameters q (see Question 1) and a (see Question 2 are **personalised**.

Instructions and rules

- Deadline: 13 May 2011.
- Material to be handed in:
  - (**Hardcopy**) Printed document containing graphs and solutions with necessary headings, annotations and comments (no essay!), and printout of program codes with comments.
  - (**Victory**) Upload two scripts `cw21.m` and `cw22.m` and all functions (also `m`-files) necessary to run the script to the Victory assignment `CW2`. When I run the scripts `cw21.m` it should recover all graphs and numerical outputs for Question 1 of your document (similarly `cw22.m` for Question 2).

  Except for Question 2(a) the scripts and functions account for 80% of the credits. The hardcopy (20%) is only there to show the output, the code, and give additional comments.
- Credit for the coding part of each question:

  100% code performs all computations correctly, is well structured and commented;
  $\geq$**80%** code performs all computations correctly but has problems;
  $\geq$**60%** code performs most of the computations correctly;
  $\geq$**40%** code does not perform computations correctly but could be made to work with minor corrections;
  $\geq$**20%** the intentions behind the code are discernible with some effort.

- This is **individual** coursework. If you are unable to implement some of the functions or scripts you may use a copy of these functions' or scripts' `m`-files from other students **if you declare at the top of your coursework document** which `m`-files are borrowed. You will then get zero credit only for these functions. If you do not declare 'borrowed code' but use it to get your results then you are plagiarising.

  In cases where **two students collaborate** they must declare this at the top of their script and make a separate mark at the top of their printout. These submissions are subject to particularly strict scrutiny for problematic constructions[1].
- For questions, clarifications and further help contact:

  Jan Sieber (`jan.sieber@port.ac.uk`, office LG.146).

[1] examples of problematic constructions (look also for warnings in the Matlab editor):

- hard-coded 'magic' numbers spread throughout the code,
- functions that should be general but only work for this example,
- one part of the code is a repetition of another part,
- stray brackets, misleading variable names or variable usages (say, using `x(:)` if `x` is scalar),
- arrays grow inside a loop,
- a variable is defined but not used.

Question 1: A predator-prey model with harvesting — continued

This question builds on your results from Coursework 1: the system of equations and the parameters (including the personalised q obtained from `GeneratePar`) are as given in Coursework 1.

(a) **(Phase portraits)** Plot all qualitatively different robust phase portraits.

- plot all equilibria, marking them differently, depending on their type (sink, source or saddle);
- plot all periodic orbits (there is at most one);
- for each sink, source and periodic orbit include a trajectory that approaches the sink/source/periodic orbit forward or backward in time;
- for each saddle include all four separatrices.

You should have six parameter regions for p with qualitatively different phase portraits. [10 marks]

(b) **(Periodic orbits)** Compute the family of periodic orbits that branches off from the Hopf bifurcation using your function `MyCurveTrack`. Plot in the (p, x)-plane and in the (p, y)-plane the maxima and minima of the periodic orbits. [10 marks]

(c) **(Saddle connection)** One qualitative change of the phase portrait that you should have observed in part (a) is a re-arrangement of the saddle separatrices. The exact boundary between the two different cases is a parameter value p_c for which the separatrix from on saddle connects to another saddle. Find this parameter p_c to 4 significant digits. [15 marks]

Total for Question 1: 35 marks

Hints:

- Use the same file `GeneratePar.p` as for Coursework 1 to genereate your q.
- On Victory in the folder `Useful Functions` you will find Matlab functions that are potentially useful, and that you can call as part of your own scripts and functions after downloading them. Beware that they are written by me and only provided for your convenience. This means that they may not give meaningful error messages if you call them with inconsistent arguments. Report difficulties to me.

Question 2: A simple chaotic system by Lorenz

E.N. Lorenz investigated simple system for the onset of convection of a fluid in a tank that is heated at the bottom. The system of differential equations is

$$\begin{aligned}\dot{x}(t) &= -\sigma(x(t)-y(t))\\ \dot{y}(t) &= R\,x(t)-y(t)-x(t)z(t)\\ \dot{z}(t) &= x(t)\,y(t)-\frac{4}{1+a^2}z(t).\end{aligned} \tag{1}$$

The (time-dependent) quantity x measures if convective circulation is clock- or anti-clockwise, y measures the horizontal temperature distribution in the fluid, z describes the vertical temperature distribution in the fluid. The parameter σ is the Prandtl number of the fluid (how well it conducts heat compared to how viscous it is), a is the aspect ratio of the tank, and R describes the amount of heating applied to the bottom of the tank. We fix

$$\sigma = 10.$$

The value of a is personalised: download `GeneratePar2.p` from Victory and call

```
a=GeneratePar2(xxxyyy);
```

where `xxxyyy` is your six-digit student ID, to obtain your personalised value for a.

For $R > 1$ the system has 3 equilibria: the (unstable) trivial equilibrium $E_0 = (0,0,0)$ (corresponding to no convection), and two equilibria of the form

$$\begin{aligned}E_+ &= (x_{\rm eq}, x_{\rm eq}, z_{\rm eq}) \quad \text{and}\\ E_- &= (-x_{\rm eq}, -x_{\rm eq}, z_{\rm eq})\end{aligned}$$

for some $x_{\rm eq}$ and $z_{\rm eq}$ that depend on a and R, and that can be calculated analytically. The two equilibria E_+ and E_- are mirror images of each other in their x- and y-components. Their z-components are identical.

Tasks

(a) **(Hopf bifurcation)** The equilibria E_+ and E_- are stable for all $R \in (1, R_H)$ where R_H is the value of R where the Jacobian of the right-hand side of (1) at $E_\pm$ has two eigenvalues on the imaginary axis (this situation is called a Hopf bifurcation). Find an analytical formula for R_H that works for all a (not only your personalised value). [8 marks]

(b) **(Fractal basins of attraction)** Select $R_1 = R_H(a) - 5$ (where you use your personalised value for a). For R_1 both equilibria, E_+ and E_- are stable. Select an $N \times N$-grid of initial values from the rectangle

$$C = \{x = -20\ldots 20, y = x, z = 0\ldots 40\}.$$

Plot for each of these initial values if it is attracted to E_+ (then give it dark colour) or to E_- (then give it a bright colour). The result of this task should be a plot in the (x,z)-plane where each point is either dark or bright. Pick N as large as your time permits (computation time is proportional to N^2). [15 marks]

(c) **(Sensitivity and attractor)** Pick $R_2 = R_H(a) + 5$ (where you use your personalised value for a). Now the equilibria E_+ and E_- are unstable.

Pick $(x, y, z) = E_+ + (5, 5, 5)$ as initial value and plot the trajectory from time $t = 0$ to $t = 50$ in the (t, x)-plane initially using $N = 1600$ steps. Then quadruple the number of steps 3 times (until you reach $N = 102,400$) and replot the trajectory each time (putting it on top of the original one). If the solver is accurate the trajectories ought to be identical.

Plot the final two trajectories (the ones with the highest accuracy) also in the (x, y)-plane. Observe that they are similar in this view. [7 marks]

Observe that the trajectories are all very different at the end in the (t, x)-plane (so, the solver is not accurate due to *sensitivity*) but that the trajectories look similar in the (x, y)-plane, forming a strange geometric object (the *attractor*).

Total for Question 2: 30 marks

Hints

- For part (a) you may find Maple useful.
- Useful Matlab commands for parts (b) and (c): `repmat`, `subplot`, `contourf`, `reshape`.