Semester 2 – Second MATLAB Exercise
Sequences - The Logistic Map

Introduction

This exercise will use Matlab to investigate the behaviour of a fairly innocuous looking equation known as the logistic map but when used draws a kind of fractal. The map is given by

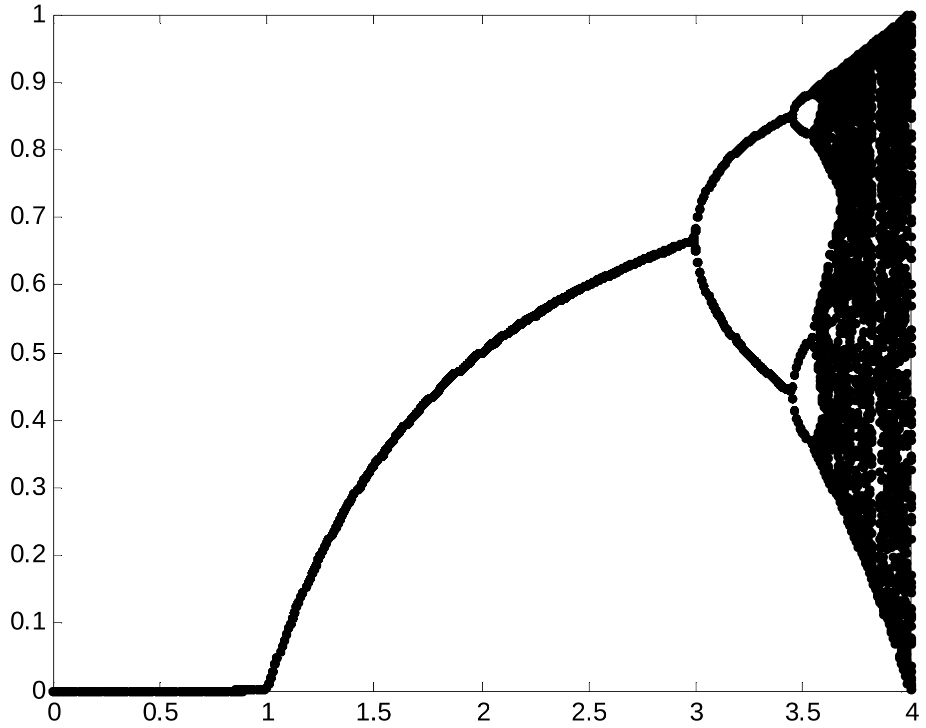
$$x[k] = r\, x[k\text{-}1]\,(1 - x[k\text{-}1]) \qquad \text{subject to } 0 < x[n] < 1$$

where r is a parameter that varies between 0 and 4. Repeatedly applying the above equation with an initial value $x[0]$ in the interval $[0 \;\; 1]$ produces a series of values that either converges to a fixed point, alternates between several fixed points or exhibits chaotic behaviour, all depending on the value of the system parameter, r.

One way of thinking of the above equation is as a harmonic oscillator, describing physical phenomena, such as a weight on a spring or a pendulum. If r is below a certain, critical value, the motion of the oscillator will damp down over time to a steady value, or the pendulum will swing more and more slowly until it finally comes to a stop. At the critical value of r the oscillator (or pendulum) will continue to run indefinitely

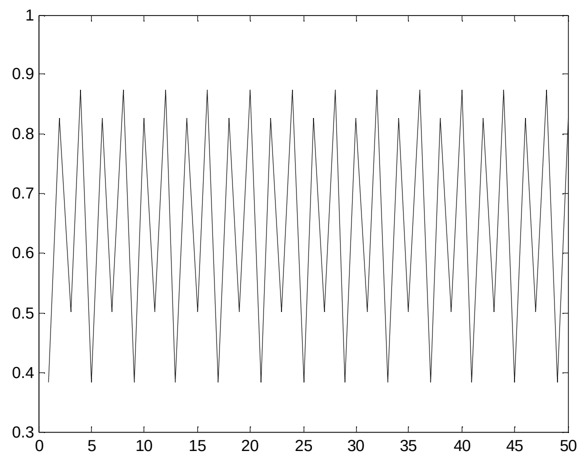
For values of r greater than the critical value the system corresponds to a driven pendulum, a common object that is known to exhibit chaotic behaviour.

One of the most widely known aspects of chaos is the striking patterns that can be created and this example is no exception. The plot below was created by running the above equation for a range of values of r that includes both the chaotic and non-chaotic ranges of the function. A sufficient number of iterations to remove any transient responses were run and then the values of $x[k]$ for N iterates of the map plotted against a range of values for r.



The Logistic Map

The above plot can be interpreted as follows. For values of $r < 1$, *x[k]* will converge to zero. For $1 < r < 3$ approximately *x[k]* converges to the corresponding *y* value. For example, when $r = 2$, *x[k]* converges to *0.5*. Above the bifurcation at the approximate value of *3* *x[k]* oscillates between the two values in the logistic map. For example, when $r = 3.2$, *x[k]* oscillates between to *0.51* and 0.8, so that the sequence will be *0.51*, *0.8*, *0.51*, *0.8*, *0.51*, *0.8* etc. Above the next bifurcation *x[k]* oscillates between 4 values, for example see the figure below.



x[k] sequence for $r = 3.5$.

The value of *r* for which the sequence starts to oscillate corresponds to the position of the first bifurcation. This occurs at towards the middle of the above map. As r increases bifurcations occur where the number of iterates doubles. These are called period doubling bifurcations and continue to a limit point at which the period is 2 to the power of infinity and chaotic behaviour exhibited. The universal number associated with such period doubling sequences is the Feigenbaum number (δ),

$$\delta = \lim_{k\to\infty} \frac{r_k - r_{k-1}}{r_{k+1} - r_k} \approx 4.669\,.$$

Laboratory Work

What must I give in and when?

Do Exercise 1 and Exercise 2 and save the results in the h:/maths1/sequences directory by Friday 19th March 2004.

To assess your work, your files **sequence.m and** and **logistic.m** will be examined on the screen, and then **sequence.m** will be run. The grade you are given will depend on two things:

A judgement of the quality of the MATLAB in your solutions.

A judgement of how well the results are presented when your script file is run.

Exercise 1.

Write a Matlab function file called sequence.m that takes a value of *r* and the number of values to calculate in the output sequence and returns the sequence *x[k]*. The function file will therefore be of the form

xk = sequence(r, no_vals)

The file should first calulate a sufficient number of values of *x[k]* to remove the transient response thus ensuring that the steady state condition is reached. You will have to experiment to find out exactly how many values this is. The next *no_vals* values *x[k]* should then be found and returned as a vector *xk*. Save this file in the directory **h:/maths1/sequences** as **sequence.m**.

Exercise 2

The second exercise is to write a function file to generate the logistic map shown on the previous page.
The function should be called logistic_map.m and will have a similar form to sequence.m,

map = logistic_map(r, no_vals)

In this function the input argument *r* is a vector containing a range of values for *r*, for example 1.0, 1.1, 1,2 1.3 and 1.4. The output *map* should be in the form of a matrix where each column is the sequence of *x* values for the corresponding value in the vector *r*, after eliminating the transient responses and can be displayed by *plot(r , map, 'k.')*. Note that Matlab's element-by-element multiplication is very useful here.

Save this file in the directory **h:/maths1/sequences** as **logistic.m**.

Find the values of the first three bifurcations on your map and add these as a comment to logistic.m

Your files should be saved in the h:/maths1/sequences directory by the end of Friday 19th March 2004.