Chaos: Logistic Map

Today we will study one of the simplest possible models which exhibits typical nonlinear dynamics (incl. chaos): the logistic map.

1. Behavior as Function of Time

The logistic map assigns to each x_n at time n the next x_{n+1} for the time $n+1$:

$$x_{n+1} = 4\, r\, x_n(1.0 - x_n) \tag{3}$$

(see in textbook on page 129 Eq.(6.5)). Write a program that starts with $x_0 = 0.5$ and runs for 100 timesteps $(x_1, \ldots, x_{100})$. Use $r = 0.2$ and print out the two columns n and x_n for timesteps $n = 0, \ldots, 100$. What do you observe for $x(n)$? Rerun your program for $r = 0.6, r = 0.85, r = 0.88$, and $r = 0.95$ and describe in each case what you observe. Get me when you are finished.

2. Sensitivity to Initial Condition

Run your program for $r = 0.95$. Compare the results $x(n)$ for the initial conditions $x_0 = 0.5$ and $x_0 = 0.500001$ by plotting their curves in the same graph. To do so redirect your resulting data into files
`executablewith0.5 > file1`
and similarly for `file2` and then graph the two datasets with
`xgraph -m file1 file2`

3. Bifurcation Diagram

Next you will investigate the dependence of your results on r. To do so change your program such that it runs the same simulation for $r = 0.0, 0.0025, 0.005, \ldots, 0.9975, 1.0$. So instead of treating r as a constant add to your program a loop over r. For each r value run the program for 500 time steps but print for only the last 100 timesteps r and x_n. Pipe your result into xgraph. You therefore obtain a graph $x_{\text{large n}}$ as a function of r. Explain what you find.

Molecular Dynamics Simulations: Simple Harmonic Oscillator

4. Euler-Cromer

4a. Copy ~ `kvollmay/classes.dir/capstone_s2005.dir/chaos.dir/chaos4a.cc` into your working directory. Have a look at the program, what it does. Run the program and look at the resulting $x(t)$ data.

4b. What is the solution to $a = -\omega_0^2 x$ with $x(0) = x_0$ and $v(0) = v_0$? Print into a second file this theoretical prediction. Compare your results of the simulation and the theoretical prediction using xgraph.

4c. Now run your simulation for $\Delta t = 0.001, 0.002, 0.005, 0.01, 0.02, 0.05, 0.1$. For each values of Δt save your simulation result. Compare in one graph all simulation results with the theoretical result. Describe what you find and what can you conclude about the best choice of Δt?

5. Velocity Verlet

Replace the Euler-Cromer step with a velocity Verlet step (Eq.(5.45a,b)). Redo 4c and compare your results from the Euler-Cromer and velocity Verlet simulations.

6. Damping

6a. Now add to your simple harmonic oscillator damping. Get me before you work on this.Use the Euler-Cromer alorithm and $\gamma = 0.5$ with $\Delta t = 0.01$ and for 1000 MD steps.

6b. Why can we not use the velocity Verlet algorithm?

7. Damped Driven Harmonic Oscillator

Next add to your system a driving force. Get me before you work on this. Use $A = 0.9, \omega_{\mathrm{D}} = 2.0/3.0$ and do 10000 MD steps each with $\Delta t = 0.01$.

Chaos & MD: Driven Damped Pendulum

8. Driven Damped Pendulum

Copy ~ `kvollmay/classes.dir/capstone_s2005.dir/chaos.dir/chaos8_sample.cc` into your working directory. Familiarize yourself with the program, run it, and look at the resulting $\theta(t)$. Then change the acceleration to an acceleration of a pendulum. Look at the resulting $\theta(t)$. Describe what you observe.

9. Phasespace Plots

9a. Look at the corresponding phasespace plot using `gawk` and `xgraph`.

9b. Change the program such that it runs for $5 \cdot 10^6$ MD steps, and that it prints only after $4 \cdot 10^6$ MDsteps. Look at the phasespace plots (and if you like also $\theta(t)$) for $A = 0.9$, $A = 1.07$ and $A = 1.15$. Describe what you find.

9c Incorporate periodic boundary conditions so that $-\pi < \theta \leq \pi$. Look at the phasespace plots for $A = 1.15$, $A = 1.35$, $A = 1.45$, $A = 1.47$, and $A = 1.50$.

10. Poincaré Plots

Change your program to obtain Poincaré plots. Choose $\Delta t = T_{\rm D}/$`NTCHOP`, where `NTCHOP=1000` and $T_{\rm D}$ is the period of the driving force. Print as before the long time behavior, but now print only every `NTCHOP` times. (Hint: Use the modulo function `%`) Get me if it is not clear how to implement the Poincaré plot. Check your program with $A = 0.9$ and all other A values from 9. Look closely for $A = 1.15$.

11. Bifurcation Diagram

This will guide you to change your program successively to obtain a bifurcation diagram. To be able to get a bifurcation diagram in a reasonable amount of time, choose `NMAXSTEP=500000` and print (later for each A) only `100` lines. Test your program with $A = 0.9$. Next plot $\theta(A)$ instead of $\omega(\theta)$. Now define A no longer as constant but as global variable (before `main` but without `const`). Define `outfile` at the beginning of `main` and add a loop over `A` to your program. Choose $0.9 \leq A \leq 1.15$ in steps of $\Delta A = 0.005$.(While you are still testing use a larger ΔA.)