Homework 5: Ordinary differential equations

Due March 8.

1. Write a program (or programs) to integrate up to four coupled differential equations (so it can handle the problems below) using the Euler method, fourth order Runge-Kutta, and Verlet implementation of Leapfrog.

   Note: The Verlet implementation of Leapfrog only applies to special cases such as when solving for position and velocity. I recommend looking at problem 3, where there are two positions and two velocities, when designing your Leapfrog program.

   For Runge-Kutta, you may use a packaged routine, although the code here is not so difficult to write yourself.. If you do used a packaged routine, **be sure to use one with a fixed timestep and order so that testing the convergence can be easily performed.** I found one with this option here `https://www.geeksforgeeks.org/runge-kutta-4th-order-method-solve-differential-equation/`, click on 'Python'. (An adaptive routine would use the same methodology discussed in class.)

2. Use your program(s) in Euler method and Runge-Kutta mode(s) to solve the differential equation for $x(t)$:

   $$\frac{d^2x}{dt^2} + x = 0;$$

   with the initial conditions $x(0) = 1, x'(0) = 0$. Note that this has the analytical solution: $x = \cos(t)$. Remember you will need to convert this 2nd order diff. equation to two first order diff. eqns.

   (a) Integrate the equation for $0 \leq t \leq 30$ using each of the methods, and step sizes of 1, .3, .1, .03, and .01. Comment on the behavior of the solutions.

   (b) Plot $\log(|x_{numerical}(30) - x_{exact}(30)|)$ as a function of $\log(stepsize)$ and check for the expected convergence of the error term.

3. Now try the two dimensional orbit described by the potential:

   $$\Phi = -\frac{1}{\sqrt{1 + x^2 + y^2}}.$$

The orbits are given by the coupled differential equations:

$$\frac{d^2x}{dt^2} = -\frac{x}{(1+x^2+y^2)^{3/2}},$$

$$\frac{d^2y}{dt^2} = -\frac{y}{(1+x^2+y^2)^{3/2}}.$$

This time also consider the Leapfrog method.

(a) Integrate this for $0 \le t \le 100$ for the initial conditions $x = 1, y = 0, x' = 0, y' = .3$. Try this with both Leapfrog and Runge-Kutta and step sizes from .01 to 1. Plot x vs. y for these integrations.

(b) Plot the energy $E = (x'^2 + y'^2)/2 + \Phi(x, y)$ as a function of time for your integrations. Comment on which algorithm is best at conserving energy. Such time-reversible algorithms are generally used by physics codes (indeed most N-body gravity codes actually use this algorithm); it is also symplectic, which roughly means it will conserves energy for a static potential.