Assignment 5.

Given April 14, due April 28.

Objective: To explore numerical methods of ODE's.

We want to compute the trajectory of a comet. In nondimensionalized variables, the equations of motion are given by the inverse square law:

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

We will always suppose that the comet starts at $t = 0$ with $x = 10$, $y = 0$, $\dot{x} = 0$, and $\dot{y} = v_0$. If v_0 is not too large, the comet will move in an elliptical orbit, returning to $x = 10$, $y = 0$ after a time $T(v_0)$. This is the time for a single orbit. The shape of the ellipse depends on v_0, as you will see in your numerical experiments.

1. Compute the position and velocity at time $t = 30$ with $v_0 = .2$ using a fixed Δt and the three methods: Forward Euler, second order Adams Bashforth, and four stage fourth order Runge Kutta. Do a convergence study to verify the expected order of accuracy of each of these methods.

2. Use a fixed time step to compute $T(v_0)$ as a function of v_0 for v_0 in the range $[.01, .5]$. Plot the result. For this, you will need some way to pick an appropriate time step, possibly by doing a convergence study. You also need to decide on a definition of T that does not require $x(T) = 10$. Because the numerical solution is not exact, the computed trajectory will not exactly come back to its original position even though the solution to the ordinary differential equation does. You should also safeguard your program against the possibility that $T(v_0)$ is infinite[1]. Make a few plots of complete orbits for various values of v_0 to see what the possible shapes are.

3. Write an adaptive time step program that uses the fourth order Runge Kutta together with a local truncation error strategy such as the one discussed in class. Compare the number of time steps needed to achieve a specified level of accuracy for fixed and adaptive time steps for a range of v_0 values. The adaptive program should much more efficient than the fixed step method when the orbit is highly elliptical.

4. The code must be modular and not hard wire the problem or the problem dimension. To test the code quality, solve the system of differential equations

$$\ddot{x}_j = (x_{j+1} - x_j) + (x_{j+1} - x_j)^3 - (x_j - x_{j-1}) - (x_j - x_{j-1})^3 \ ,$$

for $j = 1, 2, \ldots, n-1$ and $x_0 = 0$ and $x_n = n$. The initial conditions are $x_j(0) = j$, $\dot{x}_j = 0$ for $j \neq 1$ and $\dot{x}_1(0) = .1$. Try n values up to at least 100. If you make the right series of plots, or a Matlab movie, you will see a wave propagating through the chain of particles.

[1] This is not a drill.