CSC2322H

Assignment 1

September 27, 2011 University of Toronto **Due:** October 18, 2011

(Reference Kahaner, Moler and Nash, 'Numerical Methods and Software', Prentice Hall: Problem 8-6, pp. 333-334)

In this assignment you are to use ode45 or ode113 of Matlab to investigate some interesting behaviour of a model of a satellite in an earth-moon orbit. In order to access the piecewise polynomial $S(t)$ (and its derivative, $S'(t)$), associated with the approximate solution produced by one of these solvers, you can use the routine deval after applying one of the IVP solvers.

The following differential equations describe the motion of a body in orbit about two much heavier bodies. An example is an Apollo capsule in an earth-moon orbit. The coordinate system is a little tricky. The three bodies determine a plane in space and a two-dimensional Cartesian coordinate system in this plane. The origin is at the center of mass of the two heavier bodies, the x - axis is the line through these two bodies and the distance between them is the unit length. Thus, if μ is the ratio of the mass of the moon to that of the earth, then the moon and earth are located at coordinates $(1-\mu, 0)$ and $(-\mu, 0)$, respectively, and the coordinate system moves as the moon rotates about the earth. The third body, the Apollo capsule, is assumed to have a mass that is negligible compared to the other two, and its position as a function of time is $(x(t), y(t))$. The equations are derived from Newton's law of motion and the inverse square law for gravitation. The first derivatives in the equation come from the rotating coordinate system and from a friction term, which is assumed to be proportional to the velocity with proportionality constant f:

$$\begin{aligned} x'' &= 2y' + x - \frac{\tilde{\mu}(x+\mu)}{r_1^3} - \frac{\mu(x-\tilde{\mu})}{r_2^3} - fx' \\ y'' &= -2x' + y - \frac{\tilde{\mu}y}{r_1^3} - \frac{\mu y}{r_2^3} - fy' \end{aligned}$$

with

$$\mu = \frac{1}{82.45}, \quad \tilde{\mu} = 1-\mu, \quad r_1^2 = (x+\mu)^2 + y^2, \quad r_2^2 = (x-\tilde{\mu})^2 + y^2.$$

Although a great deal is known about these equations, it is not possible to find a closed form solution. One interesting class of problems involves the investigation of periodic solutions in the absence of friction. It is known that the initial conditions,

$$x(0) = 1.2, \quad x'(0) = 0, \quad y(0) = 0, \quad y'(0) = -1.04935751,$$

leads to a solution which is periodic with period $T = 6.19216933$, when $f = 0$. This orbit starts with the Apollo capsule on the far side of the moon with an altitude of 0.2 times the earth-moon distance and a prescribed initial velocity. The resulting orbit brings Apollo in close to the earth, out in a big loop on the opposite side of earth from the moon, back in close to the earth again, and finally back to its original position and velocity on the far side of the moon.

In this formulation of the model, distances are measured from the centers of the earth and moon. Assume that the moon is $238,000$ miles from the earth and that the earth is a sphere with radius $4,000$ miles. (Note that the origin of the coordinate system is within this sphere but not at its center.)

1. Use ode45 or ode113 to compute the solution with the given initial conditions. Verify that the solution is periodic with the given period. Using the interpolant provided with the code, determine how close Apollo comes to the surface of the earth in this orbit
2. When $f = 1$, with the same initial conditions as in 1., determine an approximate solution from $t = 0$ to $t = 5$. Plot the phase plane of the solution. That is, plot $x(t)$ versus $y(t)$. In this case, the Apollo capsule is 'captured' by the earth and eventually crashes.
3. Repeat the computations in 2. with $f = 0.1$. By looking at the phase plane, can you guess what is happening? This will be easier if you repeat the calculation for a longer time, say to $t = 8$.
4. Discuss how you could determine, using the IVP solvers of Matlab, whether a given set of initial conditions leads to a periodic solution. Note you don't have to write code for doing this, only discuss how you could do this task.

In writing up your solution, discuss your choice of tolerance and comment on the accuracy you feel is associated with your answer to part 1.