CS4210 Assignment 5 Due: 11/8/14 (Sat) at 6pm

You must work either on your own or with one partner. You may discuss background issues and general solution strategies with others, but the solutions you submit must be the work of just you (and your partner). If you work with a partner, you and your partner must first register as a group in CMS and then submit your work as a group. Points may be deducted for poor style and reckless inefficiency.

Topics: Initial Value Problem

1 Cannon Ball

Visit the course home page and under the "Online Readings and Software" section click on the link that takes you to the downloadable chapters of *Numerical Computing with Matlab*. Download the chapter on Ordinary Differential Equations. Submit a single function `P1` to CMS that basically does problem 7.18. (The write-up of this problem begins on page 40.) Use `ode45` with default tolerances. The function `P2()` should produce four figures, one for each of the plots that are described. Each plot displays the 17 required trajectories. Using `title` and/or `xlabel`, display the θ associated with the greatest downwind trajectory, its flight time, the actual downwind distance, the impact velocity, and the required number of f-evaluations. You will need to write an event function that terminates ODE45 as soon as the cannonball lands. (See §7.10.)

2 Epidemic

The Kermack-McKendrick model for the course of an apidemic in a population is given by

$$\begin{aligned}\dot{y}_1 &= -cy_1y_2\\ \dot{y}_2 &= cy_1y_2 - dy_2\\ \dot{y}_3 &= dy_2\end{aligned}$$

where $y_1(t)$ respresents the susceptibles, $y_2(t)$ represents the infected, and $y_3(t)$ represents infectives removed by isolation, death, or recovery and immunity. The parameters c and d represent the infection rate and the removal rate respectively.

Write a function `P2()` that uses `ODE23` with default tolerances to solve this problem numerically at $t =$ `linspace`$(0, 1, 100)$ with $c = 1$, $d = 5$, $y_1(0) = 95$, $y_2(0) = 5$, and $y_3(0) = 0$. `P2()` should plot all three graphs in a single window.

Find interesting modifications of the initial conditions and/or model parameters so that the epidemic does not grow. `P2()` should produce a second figure that confirms this. Use `title` to communicate the initial values/model parameter values that were used.

Find interesting modifications of the initial conditions and/or model parameters so that the population is wiped out. . `P2()` should produce a third figure that confirms this. Use `title` to communicate the initial values/model parameter values that were used.

3 Confirming Kepler

For appropriate values of y_p the following initial value problem defines an elliptical orbit:

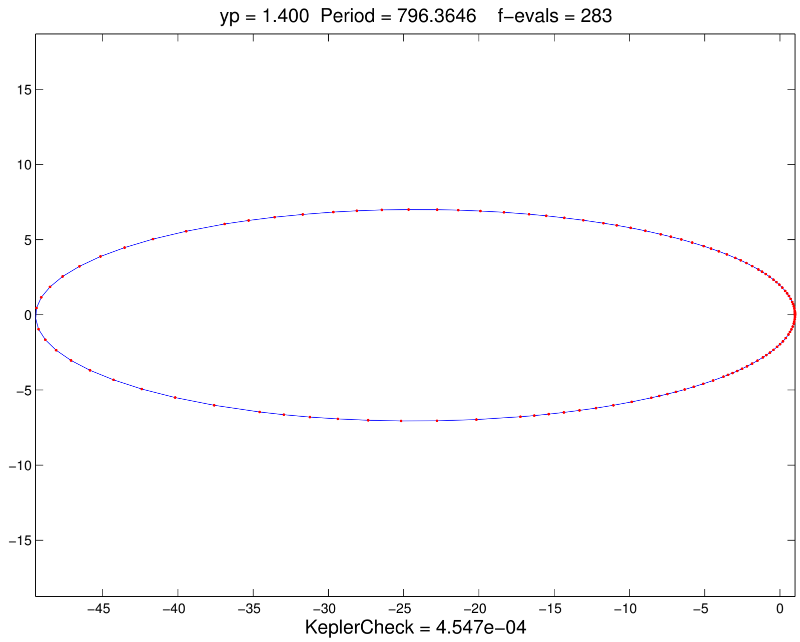
$$\ddot{x}(t) = -\frac{x(t)}{(x(t)^2 + y(t)^2)^{3/2}} \qquad x(0) = 1 \quad \dot{x}(0) = 0$$

$$\ddot{y}(t) = -\frac{y(t)}{(x(t)^2 + y(t)^2)^{3/2}} \qquad y(0) = 0 \quad \dot{y}(0) = y_p\,.$$

Think: Sun at (0,0) and planet at $(x(t), y(t))$. The integral

$$A(t_c, \Delta) \;=\; \int_{t_c}^{t_c+\Delta} (x(t)\dot{y}(t) - \dot{x}(t)y(t))dt \qquad (1)$$

is the area that is "swept out" from time $t = t_c$ to time $t = t_c + \Delta$ by the Sun-to-planet radius vector. Kepler's second law (the "equal area law") states that $A(t_c, \Delta)$ does not depend on t_c. You are to write a function `P3(yp)` that confirms this property. It displays the orbit and some key results. Sample output:



Details:

- Use `ode45` with an event function that forces termination after one period (approximately). You are free to play with the absolute error tolerance.
- The lecture m-file `Kepler` is of interest. You can count calls to `Kepler` by establishing a global variable that is incremented each call.
- Your function `P3` should plot the orbit and indicate the output points that are returned by the `ode45` call. Something like what is displayed above.
- To approximate $A(t_c, \Delta)$, approximate $x(t)$, $\dot{x}(t)$, $y(t)$, and $\dot{y}(t)$ with splines $S_x(t)$, $S_{xp}(t)$, $S_y(t)$, and $S_{yp}(t)$ respectively. Then you can use your favorite MATLAB quadrature procedure to estimate

$$\tilde{A}(t_c, \Delta) \;=\; \int_{t_c}^{t_c+\Delta} (S_x(t)S_{yp}(t) - S_{xp}(t)S_y(t))dt$$

- If P is the computed period then let $\Delta = P/100$ and define A_k by

$$A_k \;=\; \tilde{A}((k-1)\Delta, \Delta) \qquad k = 1{:}100.$$

  If everything was exact, then the A_k would all have the same value, i.e., one-hundredth of the area enclosed by the orbit. We can measure how far off we are by

$$\text{KeplerCheck} \;=\; \max |A_k - \mu|$$

  where $\mu = (A_1 + \cdots + A_{100})/100$.