Physics 411: Homework 7

1. **The Lorenz equations:** One of the most celebrated sets of differential equations in physics is the Lorenz equations:

$$\frac{\mathrm{d}x}{\mathrm{d}t} = \sigma(y - x), \qquad \frac{\mathrm{d}y}{\mathrm{d}t} = rx - y - xz, \qquad \frac{\mathrm{d}z}{\mathrm{d}t} = xy - bz,$$

where σ, r, and b are constants. (The names σ, r, and b are odd, but traditional—they are always used in these equations for historical reasons.)

These equations were first studied by Edward Lorenz in 1963, who derived them from a simplified model of weather patterns. The reason for their fame is that they were one of the first incontrovertible examples of *deterministic chaos*, the occurrence of apparently random motion even though there is no randomness built into the equations. We encountered a different example of chaos in the logistic map on the first midterm exam.

(a) Write a program to solve the Lorenz equations for the case $\sigma = 10$, $r = 28$, and $b = \frac{8}{3}$ in the range from $t = 0$ to $t = 50$ with initial conditions $(x, y, z) = (0, 1, 0)$. Have your program make a plot of y as a function of time. Note the unpredictable nature of the motion. (Hint: If you base your program on previous ones you've written, or on programs from the book, be careful. This problem has parameters r and b with the same names as variables in previous programs—make sure to give your variables new names, or use different names for the parameters, to avoid introducing errors into your code.)

(b) Modify your program to produce a plot of z against x. You should see a picture of the famous "strange attractor" of the Lorenz equations, a lop-sided butterfly-shaped plot that never repeats itself.

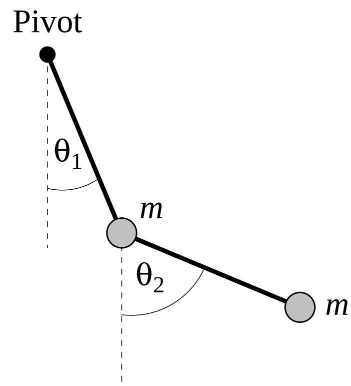
☑ **For full credit** turn in a copy of your final program and the two plots you made.

2. **The pendulum and the double pendulum:** In class we wrote a short program using the fourth-order Runge–Kutta method to calculate the motion of the nonlinear (simple) pendulum.

(a) Write your own version of that program, or recreate the one we made in class—most of the details are in Example 8.6 on page 349 of the book if you want a reminder. Then extend your program to create an animation of the motion of the pendulum. Your animation should, at a minimum, include a representation of the moving pendulum bob and the pendulum arm. (Hint: You will probably find the function `rate` discussed in Section 3.5 of the book useful for making your animation run at a sensible speed. Also, you may want to make the step-size for your Runge–Kutta calculation smaller than the framerate of your animation, i.e., do several Runge–Kutta steps per frame on screen. This is certainly allowed and may help to make your calculation more accurate.)

Once you have mastered the simple pendulum, your next challenge is to write a program to do the *double pendulum*, which is significantly more complicated.

Although it is nonlinear, the simple pendulum's movement is nonetheless perfectly regular and periodic—there are no surprises. A double pendulum, on the other hand, is completely the opposite—chaotic and unpredictable. A double pendulum consists of a normal pendulum with another pendulum hanging from its end. For simplicity let us ignore friction, and assume that both pendulums have bobs of the same mass m and massless arms of the same length ℓ. Thus the setup looks like this:



The position of the arms at any moment in time is uniquely specified by the two angles θ_1 and θ_2. The equations of motion for the angles can be derived using the Lagrangian formalism, as follows.

The heights of the two bobs, measured from the level of the pivot are

$$h_1 = -\ell \cos\theta_1, \qquad h_2 = -\ell(\cos\theta_1 + \cos\theta_2),$$

so the potential energy of the system is

$$V = mgh_1 + mgh_2 = -mg\ell(2\cos\theta_1 + \cos\theta_2),$$

where g is the acceleration due to gravity. The (linear) velocities of the two bobs are given by

$$v_1 = \ell\dot\theta_1, \qquad v_2^2 = \ell^2\big[\dot\theta_1^2 + \dot\theta_2^2 + 2\dot\theta_1\dot\theta_2\cos(\theta_1 - \theta_2)\big],$$

where $\dot\theta$ means the derivative of θ with respect to time t. (If you don't see where the second velocity equation comes from, it's a good exercise to derive it for yourself from the geometry of the pendulum.) Now the total kinetic energy is

$$T = \tfrac{1}{2}mv_1^2 + \tfrac{1}{2}mv_2^2 = m\ell^2\big[\dot\theta_1^2 + \tfrac{1}{2}\dot\theta_2^2 + \dot\theta_1\dot\theta_2\cos(\theta_1 - \theta_2)\big],$$

and the Lagrangian of the system is

$$\mathscr{L} = T - V = m\ell^2\big[\dot\theta_1^2 + \tfrac{1}{2}\dot\theta_2^2 + \dot\theta_1\dot\theta_2\cos(\theta_1 - \theta_2)\big] + mg\ell(2\cos\theta_1 + \cos\theta_2).$$

Then the equations of motion are given by the Euler–Lagrange equations

$$\frac{\mathrm{d}}{\mathrm{d}t}\left(\frac{\partial\mathscr{L}}{\partial\dot\theta_1}\right) = \frac{\partial\mathscr{L}}{\partial\theta_1}, \qquad \frac{\mathrm{d}}{\mathrm{d}t}\left(\frac{\partial\mathscr{L}}{\partial\dot\theta_2}\right) = \frac{\partial\mathscr{L}}{\partial\theta_2},$$

which in this case give

$$
\begin{aligned}
2\ddot{\theta}_1 + \ddot{\theta}_2\cos(\theta_1-\theta_2) + \dot{\theta}_2^2\sin(\theta_1-\theta_2) + 2\frac{g}{\ell}\sin\theta_1 &= 0,\\
\ddot{\theta}_2 + \ddot{\theta}_1\cos(\theta_1-\theta_2) - \dot{\theta}_1^2\sin(\theta_1-\theta_2) + \frac{g}{\ell}\sin\theta_2 &= 0,
\end{aligned}
$$

where the mass m has canceled out.

These are second-order equations, but we can convert them into first-order ones by the usual method, defining two new variables, ω_1 and ω_2, thus:

$$
\dot{\theta}_1 = \omega_1, \qquad \dot{\theta}_2 = \omega_2.
$$

In terms of these variables our equations of motion become

$$
\begin{aligned}
2\dot{\omega}_1 + \dot{\omega}_2\cos(\theta_1-\theta_2) + \omega_2^2\sin(\theta_1-\theta_2) + 2\frac{g}{\ell}\sin\theta_1 &= 0,\\
\dot{\omega}_2 + \dot{\omega}_1\cos(\theta_1-\theta_2) - \omega_1^2\sin(\theta_1-\theta_2) + \frac{g}{\ell}\sin\theta_2 &= 0.
\end{aligned}
$$

Finally we have to rearrange these into the standard form with a single derivative on the left-hand side of each one, which gives

$$
\begin{aligned}
\dot{\omega}_1 &= -\frac{\omega_1^2\sin(2\theta_1-2\theta_2) + 2\omega_2^2\sin(\theta_1-\theta_2) + (g/\ell)\bigl[\sin(\theta_1-2\theta_2) + 3\sin\theta_1\bigr]}{3-\cos(2\theta_1-2\theta_2)},\\
\dot{\omega}_2 &= \frac{4\omega_1^2\sin(\theta_1-\theta_2) + \omega_2^2\sin(2\theta_1-2\theta_2) + 2(g/\ell)\bigl[\sin(2\theta_1-\theta_2) - \sin\theta_2\bigr]}{3-\cos(2\theta_1-2\theta_2)}.
\end{aligned}
$$

(This last step is quite tricky and involves some trigonometric identities. If you're not certain of how the calculation goes you may find it useful to go through the derivation for yourself.)

These two equations, along with the equations $\dot{\theta}_1 = \omega_1$ and $\dot{\theta}_2 = \omega_2$, give us four first-order equations which between them define the motion of the double pendulum.

(b) Derive an expression for the total energy $E = T + V$ of the system in terms of the variables θ_1, θ_2, ω_1, and ω_2, plus the constants g, ℓ, and m.

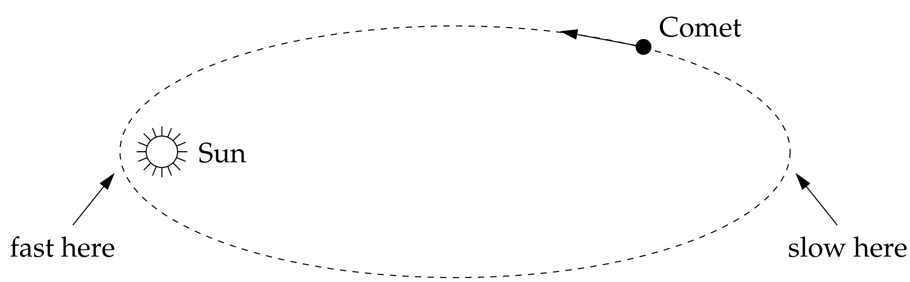
(c) Write a program using the fourth-order Runge–Kutta method to solve the equations of motion for the case where $\ell = 40\,\text{cm}$, with the initial conditions $\theta_1 = \theta_2 = 90°$ and $\omega_1 = \omega_2 = 0$. Use your program to calculate the total energy of the system assuming that the mass of the bobs is 1 kg each, and make a graph of energy as a function of time from $t = 0$ to $t = 100$ seconds.

Because of energy conservation, the total energy should be constant over time (actually it should be zero for this particular set of initial conditions), but you will find that it is not perfectly constant because of the approximate nature of the solution of the differential equation. Choose a suitable value of the step size h to ensure that the variation in energy is less than 10^{-5} Joules over the course of the calculation.

(d) Modify your program to make an animation of the motion of the double pendulum over time. At a minimum, the animation should show the two arms and the two bobs. Hint: You will probably find that the value of h needed to get the required accuracy in your solution gives a frame-rate much faster than any that can reasonably be displayed in your animation, so you won't be able to display every time-step of the calculation in the animation. Instead you will have to arrange the program so that it updates the animation only once every several Runge–Kutta steps.

☑ **For full credit** turn in a printout of your final program for the double pendulum, your calculation from part (b), your graph of the energy as a function of time from part (c), and screenshots showing your two animations in action, for the simple pendulum and the double pendulum.

3. **Cometary orbits:** Many comets travel in highly elongated orbits around the Sun. For much of their lives they are far out in the solar system, moving very slowly, but on rare occasions their orbit brings them close to the Sun for a fly-by and for a brief period of time they move very fast indeed:



This is a classic example of a system for which an adaptive step size method is useful, because for the large periods of time when the comet is moving slowly we can use long time-steps, so that the program runs quickly, but short time-steps are crucial in the brief but fast-moving period close to the Sun.

The differential equation obeyed by a comet is straightforward to derive. The force between the Sun, with mass M at the origin, and a comet of mass m with position vector $\mathbf{r}$ is GMm/r^2 in direction $-\mathbf{r}/r$ (i.e., the direction towards the Sun), and hence Newton's second law tells us that

$$m\frac{\mathrm{d}^2\mathbf{r}}{\mathrm{d}t^2} = -\left(\frac{GMm}{r^2}\right)\frac{\mathbf{r}}{r}.$$

Canceling the m and taking the x component we have

$$\frac{\mathrm{d}^2 x}{\mathrm{d}t^2} = -GM\frac{x}{r^3},$$

and similarly for the other two coordinates. We can, however, throw out one of the coordinates because the comet stays in a single plane as it orbits. If we orient our axes so that this plane is perpendicular to the z-axis, we can forget about the z coordinate and we are left with just two second-order equations to solve:

$$\frac{d^2x}{dt^2} = -GM\frac{x}{r^3}, \qquad \frac{d^2y}{dt^2} = -GM\frac{y}{r^3},$$

where $r = \sqrt{x^2 + y^2}$.

(a) Turn these two second-order equations into four first-order equations, using the methods you have learned.

(b) Write a program to solve your equations using the fourth-order Runge–Kutta method with a *fixed* step size. You will need to look up the mass of the Sun and Newton's gravitational constant G. As an initial condition, take a comet at coordinates $x = 4$ billion kilometers and $y = 0$ (which is somewhere out around the orbit of Neptune) with initial velocity $v_x = 0$ and $v_y = 500\,\mathrm{m\,s^{-1}}$. Make a graph showing the trajectory of the comet (i.e., a plot of y against x).

Choose a fixed step size h that allows you to accurately calculate at least two full orbits of the comet. Since orbits are periodic, a good indicator of an accurate calculation is that successive orbits of the comet lie on top of one another on your plot. If they do not then you need a smaller value of h. Give a short description of your findings. What value of h did you use? What did you observe in your simulation? How long did the calculation take?

(c) Make a copy of your program and modify the copy to do the calculation using an adaptive step size. Set a target accuracy of $\delta = 1$ kilometer per year in the position of the comet and again plot the trajectory. What do you see? How do the speed, accuracy, and step size of the calculation compare with those in part (b)?

(d) Modify your program to place dots on your graph showing the position of the comet at each Runge–Kutta step around a single orbit. You should see the steps getting closer together when the comet is close to the Sun and further apart when it is far out in the solar system.

Calculations like this can be extended to cases where we have more than one orbiting body—see Exercise 8.16 in the book for an example. We can include planets, moons, asteroids, and others. Analytic calculations are impossible for such complex systems, but with careful numerical solution of differential equations we can calculate the motions of objects throughout the entire solar system.

☑ **For full credit** turn in a copy of your final (adaptive) program, your answers to the calculations and questions in parts (a), (b), and (c), and your plot from part (d).