

# Classical & Nonlinear Dynamics

## 3.1 Chapter Overview

The first half of this chapter focuses on nonlinear oscillations. The prime tool is the numerical solution of ordinary differential equations (§1.7), which is both easy and precise. We look at a variety of systems, with emphases on behaviors in phase-space plots and bifurcation diagrams. In addition, the solution for the realistic pendulum gets new life since we can actually evaluate elliptic integrals using the integration techniques of §1.5. We then analyze the output from the simulations using the discrete Fourier transform of §2.4. The second half of the chapter examines projectile motion, bound states of three body systems, and Coulomb and chaotic scattering. We also look at some of the unusual behavior of billiards, which are a mix of scattering and bound states. (The quantum version of these same billiards is examined in §6.8.4.) Problems related to Lagrangian and Hamiltonian dynamics then follow, with the actual computation of Hamilton's principle. Finally, we end the chapter with the problem of several weights connected by strings; a simple problem that requires a complex solution involving both a derivative algorithm and a search algorithm, as discussed in Chapters 1 and 2.

Note that Chapter 8 contains a number of problems dealing with the several discrete maps that lead to chaotic behavior in biological systems. These materials, as well as the development of the predator-prey models in that chapter, might well be included in a study of classical dynamics.

## 3.2 Oscillators

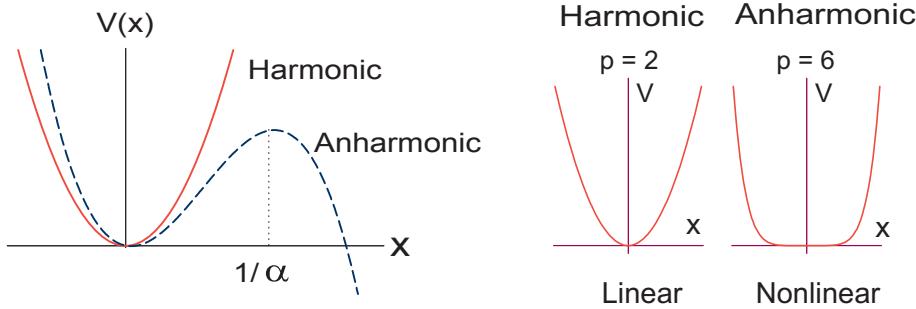
### 3.2.1 First a Linear Oscillator

1. Consider the 1-D harmonic (linear) oscillator with viscous friction:

$$\frac{d^2x}{dt^2} + \kappa \frac{dx}{dt} + \omega_0^2 x = 0 \quad (3.1)$$

- a. Verify by hand or by using a symbolic manipulation package (§3.2.7) that

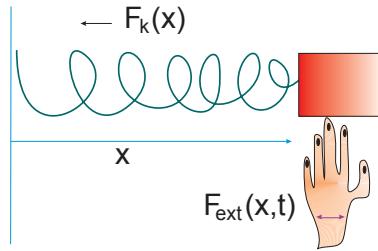
$$x(t) = e^{at} [x_0 \cos \omega t + (p_0/m\omega) \sin \omega t] \quad (3.2)$$



**Figure 3.1.** *Left:* The potentials of an harmonic oscillator (solid curve) and of an anharmonic oscillator (dashed curve). If the amplitude becomes too large for the anharmonic oscillator, the motion becomes unbound. *Right:* The shapes of the potential energy function  $V(x) \propto |x|^p$  for  $p = 2$  and  $p = 6$ . The “linear” and “nonlinear” labels refer to the restoring force derived from these potentials.

is a solution of (3.1).

- b. Determine the constants  $\omega$ ,  $x_0$ , and  $p_0$  in (3.2) in terms of initial conditions.
  - c. Plot the phase-space portrait  $[x(t), p(t)]$  for  $\omega_0 = 0.8$  and several values of  $p(0)$ . (Phase space portraits are discussed in §3.3.3.)
2. Do a number of things to check that your ODE solver is working well and that you know the proper integration step size needed for high precision.
- a. Choose initial conditions corresponding to a frictionless oscillator initially at rest, for which the analytic solution is:
- $$x(t) = A \sin(\omega_0 t), \quad v = \omega_0 A \cos(\omega_0 t), \quad \omega_0 = \sqrt{k/m}. \quad (3.3)$$
- b. Pick values of  $k$  and  $m$  such that the period  $T = 2\pi/\omega = 10$ .
  - c. Start with a time step size  $h \simeq T/5$  and make  $h$  smaller until the solution looks smooth, has a period that remains constant for a large number of cycles, and agrees with the analytic result. As a general rule of thumb, we suggest that you start with  $h \simeq T/100$ , where  $T$  is a characteristic time for the problem at hand. You should start with a large  $h$  so that you can see a bad solution turn good.
  - d. Make sure that you have exactly the same initial conditions for the analytic and numerical solutions (zero displacement, nonzero velocity) and then plot the two solutions together. Also make a plot of their difference versus time since graphical agreement may show only 2-3 places of sensitivity.
  - e. Try different initial velocities and verify that a *harmonic* oscillator is *isochronous*, that is, that its period does *not* change as the amplitude is made large.



**Figure 3.2.** A mass  $m$  (the block) attached to a spring with restoring force  $F_k(x)$  driven by an external time-dependent driving force (the hand).

### 3.2.2 Nonlinear Oscillators

Figure 3.2 shows a mass  $m$  attached to a spring that exerts a restoring force  $F_k(x)$  toward the origin, as well as a hand that exerts a time-dependent external force  $F_{ext}(x, t)$  on the mass. The motion is constrained to one dimension and so Newton's second law provides the equation of motion

$$F_k(x) + F_{ext}(x, t) = m \frac{d^2x}{dt^2}, \quad (3.4)$$

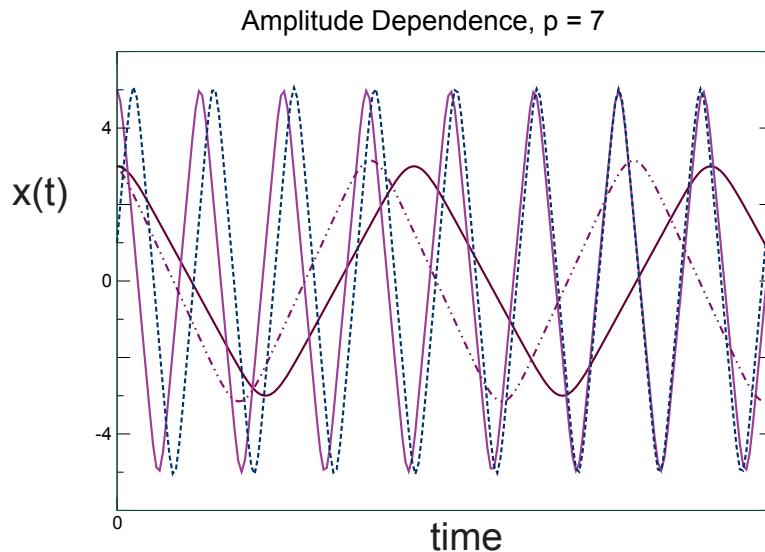
Consider two models for a nonlinear oscillator:

$$V(x) \simeq \frac{1}{2}kx^2 \left(1 - \frac{2}{3}\alpha x\right), \quad \text{Model 1,} \quad (3.5)$$

$$V(x) = \frac{1}{p}kx^p, \quad \text{Model 2 (}p\text{ even).} \quad (3.6)$$

Model 1's potential is quadratic for small displacements  $x$ , but also contains a perturbation that introduces a nonlinear term to the force for large  $x$  values: If  $\alpha x \ll 1$ , we would expect harmonic motion, though as  $x \rightarrow 1/\alpha$  the anharmonic effects should increase. Model 2's potential is proportional to an arbitrary  $p$  of the displacement  $x$  from equilibrium, with the power  $p$  being even for this to be a restoring force. Some characteristics of both potentials can be seen in Figure 3.1.

1. Modify your harmonic oscillator program to study anharmonic oscillations for strengths in the range  $0 \leq \alpha x \leq 2$ . Do *not* include any explicit time-dependent forces yet.
2. Test that for  $\alpha = 0$  you obtain simple harmonic motion.
3. Check that the solution remains periodic as long as  $x_{max} < 1/\alpha$  in model 1 and for all initial conditions in model 2.



**Figure 3.3.** The position versus time for oscillations within the potential  $V \propto x^7$  for four different initial amplitudes. Each is seen to have a different period.

4. Check that the maximum speeds always occur at  $x = 0$  and that the velocity vanishes at the maximum  $x$ 's.
5. Verify that nonharmonic oscillators are *nonisochronous*, that is, that vibrations with different amplitudes have different periods (Figure 3.3).
6. Describe how the shapes of the oscillations change for different  $\alpha$  or  $p$  values.
7. In Model 2, for what values of  $p$  and  $x$  will the potential begin to look like a square well? Note that for large values of  $p$ , the forces and accelerations get large near the turning points, and so you may need a smaller time step  $h$  to track the rapid changes in motion.
8. Devise an algorithm to determine the period  $T$  of the oscillation by recording times at which the mass passes through the origin. Note that because the motion may be asymmetric, you must record at least three times to deduce the period.
9. Verify that the oscillators are *nonisochronous*, that is, that vibrations with different amplitudes have different periods.
10. Construct a graph of the deduced period as a function of initial amplitude.
11. Verify that the motion is oscillatory, though not harmonic, as the energy approaches  $k/6\alpha^2$ , or for  $p \neq 2$ .

12. Verify that for oscillations with energy  $E = k/6\alpha^2$ , the motion in potential 1 changes from oscillatory to translational.
13. For Model 1, see how close you can get to the *separatrix* where a single oscillation takes an infinite amount of time.

### 3.2.3 Assessing Precision via Energy Conservation

It is important to test the precision and reliability of a numerical solution. For the present cases, as long as there is no friction and no external forces, we expect energy to be conserved. Energy conservation, which follows from the mathematics and not the algorithm, is hence an independent test of our algorithm.

1. Plot for 50 periods the potential energy  $\text{PE}(t) = V[x(t)]$ , the kinetic energy  $\text{KE}(t) = mv^2(t)/2$  and the total energy  $E(t) = \text{KE}(t) + \text{PE}(t)$ .
2. Check the long-term *stability* of your solution by plotting

$$-\log_{10} \left| \frac{E(t) - E(t=0)}{E(t=0)} \right| \simeq \text{number of places of precision} \quad (3.7)$$

for a large number of periods. Because  $E(t)$  should be independent of time, the numerator is the absolute error in your solution, and when divided by  $E(0)$ , becomes the relative error. If you cannot achieve 11 or more places, then you need to decrease the value of  $h$  or debug.

3. Because a particle bound by a large- $p$  oscillator is essentially “free” most of the time, you should observe that the average of its kinetic energy over time exceeds its average potential energy. This is actually the physics behind the Virial theorem for a power-law potential [Marion & Thornton(03)]:

$$\langle \text{KE} \rangle = \frac{p}{2} \langle \text{PE} \rangle. \quad (3.8)$$

Verify that your solution satisfies the Virial theorem and computed the effective value of  $p$ .

### 3.2.4 Models of Friction

Three simple models for frictional force are *static*, *kinetic*, and *viscous* friction:

$$F_f^{(\text{static})} \leq -\mu_s N, \quad F_f^{(\text{kinetic})} = -\mu_k N \frac{v}{|v|}, \quad F_f^{(\text{viscous})} = -bv, \quad (3.9)$$

where  $N$  is the *normal force* on the object under consideration,  $\mu$  and  $b$  are parameters, and  $v$  is the velocity<sup>1</sup>.

---

<sup>1</sup>The effect of air resistance on projectile motion is studied §3.6.

1. Extend your harmonic oscillator code to include the three types of friction in (3.9) and observe how the motion differs for each.
  - a. For the simulation with static plus kinetic friction, each time the oscillator has  $v = 0$  you need to check that the restoring force exceeds the static frictional force. If not, the oscillation must end at that instant. Check that your simulation terminates at nonzero  $x$  values.
  - b. For your simulations with viscous friction, investigate the qualitative changes that occur for increasing  $b$  values:

*Underdamped:*  $b < 2m\omega_0$  Oscillate within decaying envelope

*Critically:*  $b = 2m\omega_0$  Nonoscillatory, finite decay time

*Over damped:*  $b > 2m\omega_0$  Nonoscillatory, infinite decay time

### 3.2.5 Linear & Nonlinear Resonances

A periodic external force of frequency  $\omega_f$  is applied to an oscillatory system with natural frequency  $\omega_0$ . As the frequency of the external force passes through  $\omega_0$ , a *resonance* may occur. If the oscillator and the driving force remain in phase over time, the amplitude of oscillation will increase continuously unless there is some mechanism, such as friction or nonlinearity, that limits the growth. If the frequency of the driving force is close to, though not exactly equal to  $\omega_0$ , a related phenomena, *beating*, may occur in which there is interference between the natural vibration and the driven vibrations:

$$x \simeq x_0 \sin \omega_f t + x_0 \sin \omega_0 t = \left( 2x_0 \cos \frac{\omega_f - \omega_0}{2} t \right) \sin \frac{\omega_f + \omega_0}{2} t. \quad (3.10)$$

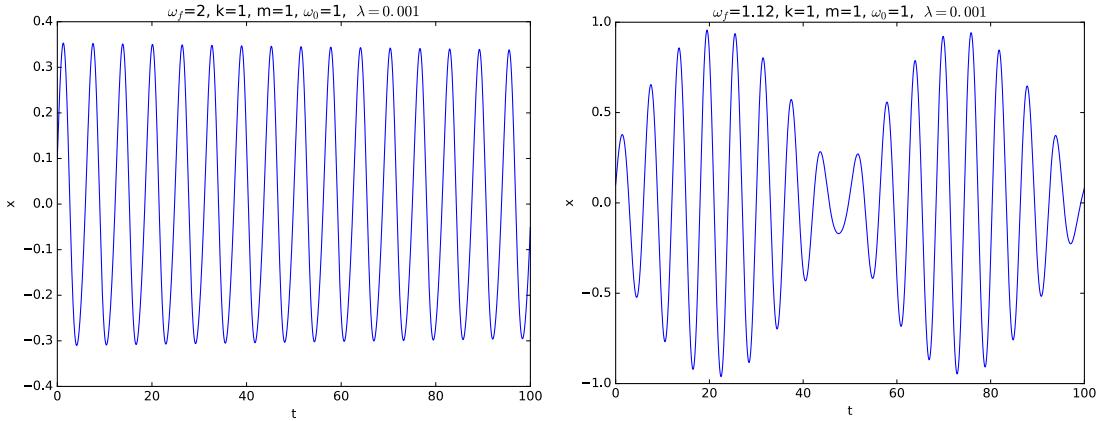
The resulting motion resembles the natural oscillation of the system at the average frequency  $(\omega_f + \omega_0)/2$ , however with an amplitude  $2x_0 \cos(\omega_f - \omega_0)/2t$  that varies slowly with the *beat frequency*  $(\omega_f - \omega_0)/2$ .

1. Include the time-dependent external force  $F \cos(\omega_f t)$  in your rk4 ODE solver. You can modify the `rk4Call.py` program given earlier in Listing 1.12 which uses `Visual`, or `ForcedOscillate.py` in Listing 3.1 which uses `Matplotlib`.

2. Start with a harmonic oscillator with these parameters and the initial conditions:

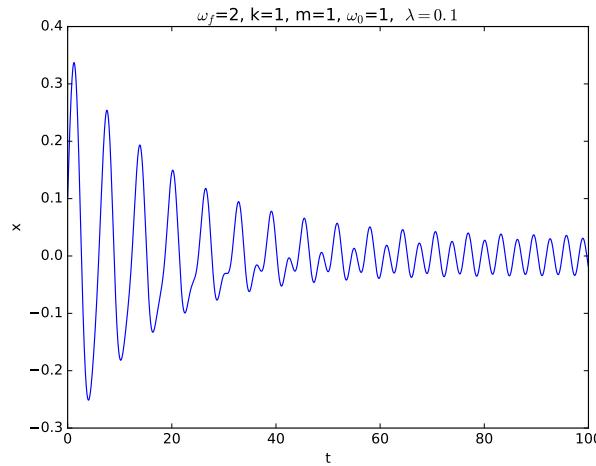
$$p = 2, \quad k = 1, \quad m = 1, \quad \mu = 0.001, \quad \omega_f = 2, \quad y^{(0)}(0) = 0.1, \quad y^{(1)}(0) = 0.3.$$

3. Starting with a large value for the magnitude of the driving force  $F_0$  should lead to *mode locking* in which the system is overwhelmed by the driving force, and, after the transients die out, will oscillate in phase with the driver. See if you can reproduce a behavior like that found on the left of Figure 3.4.
4. Why don't the oscillations appear damped?



**Figure 3.4.** *Left:* Position versus time for an oscillator with  $p = 2$ ,  $k = 1$ ,  $m = 1$ ,  $\mu = 0.001$ ,  $\omega_f = 2$ , and  $\omega_0 = 1$ . *Right:* Position versus time for  $k = 1$ ,  $m = 1$ ,  $\lambda = 0.001$ ,  $\omega_f = 1.12$ , and  $\omega_0 = 1$ .

5. With the same constants as before, change to  $\omega_f = 1.12$ , which is close to the natural frequency  $\omega_0 = 1$ . You should obtain oscillations similar to that on the right of Figure 3.4.
6. Verify that for  $p = 2$ , the beat frequency, that is, the number of variations in intensity per unit time, equals the frequency difference in cycles per second  $(\omega_f - \omega_0)/2\pi$ . With the same constants as in Figure 3.4, see the effects of a large viscosity.
7. Return to parameters that gave you distinct beating and make a series of runs in which you progressively increase the frequency of the driving force in the range  $\omega_0/10 \leq \omega_f \leq 10\omega_0$ . Plot up the response for a number of interesting runs.
8. Make a plot of the maximum amplitude of oscillation *versus* the driver's  $\omega_f$ . This should exhibit a resonance peak at  $\omega_0$ .
9. Explore what happens when you drive a nonlinear system. Start with a system being close to harmonic, and verify that you obtain beating in place of the blowup that occurs for the linear system.
10. Explain the origin of the beating in nonlinear resonances.
11. Investigate how the inclusion of viscous friction modifies the curve of maximum amplitude *versus* driver frequency. You should find that friction broadens the curve.



**Figure 3.5.** Position versus time for an oscillator with  $p = 2$ ,  $k = 1$ ,  $m = 1$ ,  $\mu = 0.1$ ,  $\omega_f = 2$  and  $\omega_0 = 1$ .

12. Explain how the character of the resonance changes as the exponent  $p$  in model 2 is made progressively larger. You should find that at large  $p$  the mass effectively “hits” the wall and falls out of phase with the driver, thereby making the driver less effective at pumping in energy.

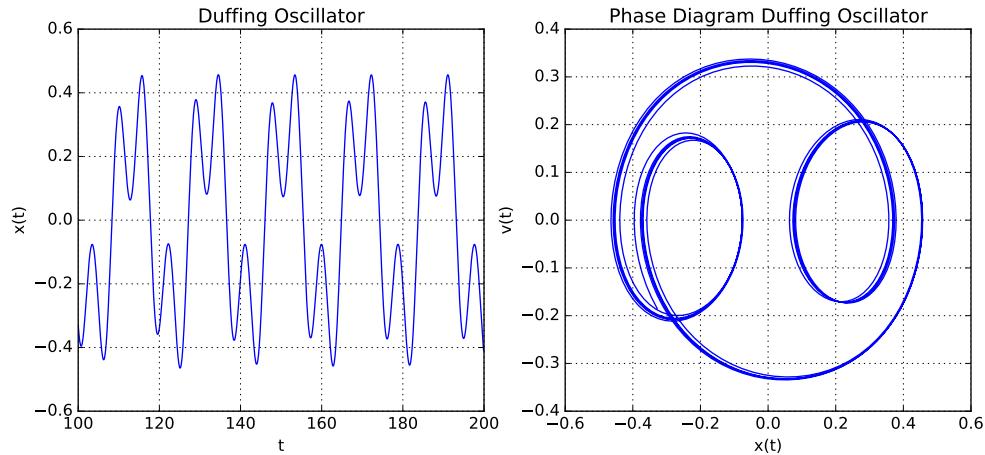
### 3.2.6 Famous Nonlinear Oscillators

1. The nonlinear behavior in once-common objects such as vacuum tubes and metronomes is described by the **van der Pool Equation**,

$$\frac{d^2x}{dt^2} + \mu(x^2 - x_0^2) \frac{dx}{dt} + \omega_0^2 x = 0. \quad (3.11)$$

- a. Explain why you can think of (3.11) as describing an oscillator with  $x$ -dependent damping.
- b. Create some phase space plots of the solutions to this equation, that is, plots of  $\dot{x}(t)$  versus  $x(t)$ .
- c. Verify that this equation produces a limit cycle in phase space, that is orbits internal to the limit cycle spiral out until they reach the limit cycle, and those external to it spiral in to it.

The **Duffing oscillator** is another example of a damped, driven nonlinear oscil-



**Figure 3.6.** A period three solution for a forced Duffing oscillator. *Left:*  $x(t)$ , *Right:*  $v(t)$  versus  $x(t)$ .

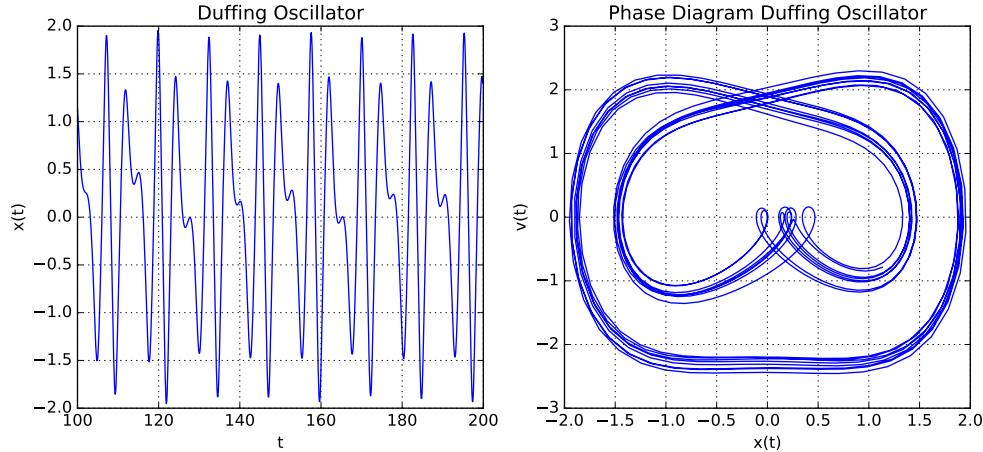
lator. It is described by the differential equation [Kov(11), Enns(01)]:

$$\frac{d^2x}{dt^2} = -2\gamma \frac{dx}{dt} - \alpha x - \beta x^3 + F \cos \omega t. \quad (3.12)$$

In Listing 1.13 we gave as an example the code `rk4Duffing.py` that solves a simple form of this equation.

1. Modify your ODE solver program to solve (3.12).
2. First choose parameter values corresponding to a simple harmonic oscillator and verify that you obtain sinusoidal behavior for  $x(t)$  and a closed elliptical phase space plot.
3. Include a driving force, wait 100 cycles in order to eliminate transients, and then create a phase space plot. We used  $\alpha = 1.0$ ,  $\beta = 0.2$ ,  $\gamma = 0.2$ ,  $\omega = 1.$ ,  $F = 4.0$ ,  $x(0) = 0.009$ , and  $\dot{x}(0) = 0$ .
4. Search for a period–three solution like those in Figure 3.6, where we used  $\alpha = 0.0$ ,  $\beta = 1.$ ,  $\gamma = 0.04$ ,  $\omega = 1.$ , and  $F = 0.2$ .
5. Change your parameters to  $\omega=1$  and  $\alpha=0$  in order to model an *Ueda* oscillator. Your solution should be similar to Figure 3.7.
6. Consider a nonlinear perturbed harmonic oscillator with friction:

$$p = \dot{q}, \quad \dot{p} = q - q^3 - p. \quad (3.13)$$



**Figure 3.7.** The Ueda oscillator *Left:*  $x(t)$ , *Right:*  $v(t)$  versus  $x(t)$ .

- Create several phase space portraits for this system.
  - Determine analytically the Liapunov coefficients and from these the position and nature of the critical points.
  - Does your analytic analysis agree with your computed results?
7. Investigate the simplified version of the Lorenz attractors developed by **Roessler** [Becker(86)]:

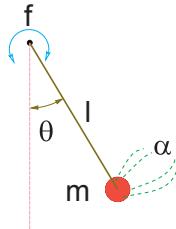
$$\dot{x} = -y - z \quad \dot{y} = x + ay \quad \dot{z} = b + xz - cz \quad (a, b, c) = (0.2, 0.2, 5.7). \quad (3.14)$$

- Compute and plot  $x(t)$ ,  $y(t)$ , and  $z(t)$  as functions of time.
- Plot projections of your solutions onto the  $(x, y)$  and  $(x, \dot{x})$  planes.
- Make a Poincaré mapping of the transverse section  $\dot{x} = 0$ . (A Poincaré mapping is the intersection of a periodic orbit in the phase space with a lower-dimensional subspace.)
- When  $\dot{x} = 0$ ,  $x$  has an extremum. Plot the value of the extrema  $x_{i+1}$  as a function of the previous extremum  $x_i$ .

### 3.2.7 Solution via Symbolic Computing

- Repeat the study of the damped and driven harmonic oscillator using a symbolic manipulation package. Listing 3.2 presents a direct solution of the differential equation using SymPy (see Chapter 1 for discussion of Python packages) and produces the output

```
ODE to be solved:  
Eq(kap*Derivative(f(t), t) + w0**2*f(t) + Derivative(f(t), t, t), 0)
```



**Figure 3.8.** A pendulum of length  $l$  driven through resistive air (dotted arcs) by an external sinusoidal torque (semicircle). The strength of the external torque is given by  $f$  and that of air resistance by  $\alpha$ .

```
Solution of ODE:  
Eq(f(t), C1*exp(t*(-kap - sqrt(kap**2 - 4*w0**2))/2) + C2*exp(t*(-kap + sqrt(kap**2
```

In turn, Listing 3.2 presents a determination of the parameters in the solution by evaluating the initial conditions, and produces the output

```
Soltn:  
(x0*cos(t*w) + p0*sin(t*w)/(m*w))*exp(alf*t)  
Derivatives:  
kap*(alf*(x0*cos(t*w) + p0*sin(t*w)/(m*w))*exp(alf*t) + (-w*x0*sin(t*w)  
+ p0*cos(t*w)/m)*exp(alf*t)) + w0**2*(x0*cos(t*w)  
+ p0*sin(t*w)/(m*w))*exp(alf*t) + (alf**2*(x0*cos(t*w) + p0*sin(t*w)/(m*w))  
- 2*alf*(w*x0*sin(t*w) - p0*cos(t*w)/m) - w*(w*x0*cos(t*w) + p0*sin(t*w)/m))*exp(alf*t)  
Initial value y2:  
y2 = alf**2*x0 + 2*alf*p0/m + kap*(alf*x0 + p0/m) - w**2*x0 + w0**2*x0  
Coefficients of p0/m, A = {-kap/2}  
W = {-sqrt(alf**2 + alf*kap + w0**2), sqrt(alf**2 + alf*kap + w0**2)}  
frequency w = {-sqrt(-kap**2 + 4*w0**2)/2, sqrt(-kap**2 + 4*w0**2)/2}
```

2. As you can see from the output, the analytic solution is output as exponentials. How do you reconcile these results compared with the previous ones in terms of sines and cosines?
3. Use a symbolic manipulation package to solve the equations of motion for a nonlinear oscillator.

### 3.3 Realistic Pendula

We call a pendulum without a small angle approximation “realistic” or “nonlinear”, and a realistic pendulum with a periodic driving torque “chaotic”. The chaotic pendulum in Figure 3.8 is described by the ordinary differential equation,

$$\frac{d^2\theta}{dt^2} = -\omega_0^2 \sin \theta - \alpha \frac{d\theta}{dt} + f \cos \omega t, \quad \omega_0 = \frac{mgl}{I}, \quad \alpha = \frac{\beta}{I}, \quad f = \frac{\tau_0}{I}. \quad (3.15)$$

Here  $\omega_0$  is the natural frequency, the  $\alpha$  term arises from friction, and the  $f$  term measures the strength of the driving torque. The difficulty with the computer study of this system is that the four parameters space  $(\omega_0, \alpha, f, \omega)$  is immense, and the results may be hypersensitive to the exact values used for them. So you may have to adjust somewhat the suggested parameter values to obtain the predicted behaviors.

1. Consider the ODE for an undriven realistic pendulum without friction:

$$\frac{d^2\theta}{dt^2} = -\omega_0^2 \sin \theta. \quad (3.16)$$

- a. Use conservation of energy to show that the pendulum's velocity as a function of position is

$$\frac{d\theta}{dt}(\theta) = 2\sqrt{\frac{g}{l}} [\sin^2(\theta_0/2) - \sin^2(\theta/2)]^{1/2}, \quad (3.17)$$

where  $\theta_0$  is the angle of displacement for a pendulum released from rest.

- b. Solve this equation for  $dt$  and then integrate analytically to obtain the integral expression for the period of oscillation as a function of  $\theta_0$ :

$$\frac{T}{4} = \frac{T_0}{4\pi} \int_0^{\theta_m} \frac{d\theta}{[\sin^2(\theta_m/2) - \sin^2(\theta/2)]^{1/2}} = 4\sqrt{\frac{L}{g}} K(\sin^2 \frac{\theta_0}{2}). \quad (3.18)$$

The  $K$  function in (3.18) is an *elliptic integral* of the first kind, and in §3.3.1 we discuss its numerical evaluation.

2. Again consider the ODE for an undriven realistic pendulum without friction (3.16), though now solve it numerically.
  - a. To ensure that you can solve the ODE for the realistic pendulum with high accuracy, start by plotting the total energy  $E(t)$  as a function of time. Adjust (decrease) the integration step size in rk4 until the relative energy of your solution  $E(t)/E(0)$  varies by less than  $10^{-6}$ , even for exceedingly large times.
  - b. It may be easiest to start the pendulum at  $\theta = 0$  with  $\dot{\theta}(0) \neq 0$ , and gradually increase  $\dot{\theta}(0)$  to increase the energy of the pendulum. Check that for all initial conditions your solution is periodic with unchanging amplitude.
  - c. Verify that as the initial KE approaches  $2mgl$ , the motion remains oscillatory but with ever-increasing period.
  - d. At  $E = 2mgl$  (the *separatrix*), the motion changes from oscillatory to rotational (“over the top” or “running”). See how close you can get your solution to the separatrix and hence to an infinite period.
  - e. Convert your different oscillations to sound and hear the difference between harmonic motion (boring) and anharmonic motion containing overtones (interesting). Some ways to do this is discussed in §3.4.2.

### 3.3.1 Elliptic Integrals

Conservation of energy permits us to solve for the period of a realistic pendulum released from rest with initial displacement of  $\theta_0$ :

$$T = 4\sqrt{\frac{L}{g}} K(\sin^2 \frac{\theta_0}{2}), \quad (3.19)$$

$$\simeq T_0 \left[ 1 + \left( \frac{1}{2} \right)^2 \sin^2 \frac{\theta_m}{2} + \left( \frac{1 \cdot 3}{2 \cdot 4} \right)^2 \sin^4 \frac{\theta_m}{2} + \dots \right]. \quad (3.20)$$

Here  $K$  is the incomplete elliptic integral of the first kind,

$$K(m) = \int_0^1 \frac{dt}{\sqrt{(1-t^2)(1-mt^2)}} = \int_0^{\pi/2} \frac{d\theta}{\sqrt{1-m \sin^2 \theta}}. \quad (3.21)$$

Tabulated values for elliptic integrals are available, or they can be evaluated directly using, for instance, Gaussian quadrature. In a mathematical sense, an infinite power series provides an exact representation of a function. However, it is often not good as an algorithm because it may converge slowly and because round-off error may dominate when there are many terms summed or when there is significant cancellation of terms. On the other hand, a polynomial approximation, such as [Abramowitz & Stegun(72)]

$$K(m) \simeq a_0 + a_1 m_1 + a_2 m_1^2 - [b_0 + b_1 m_1 + b_2 m_1^2] \ln m_1 + \epsilon(m),$$

$$m_1 = 1 - m, \quad 0 \leq m \leq 1, \quad |\epsilon(m)| \leq 3 \times 10^{-5},$$

$$\begin{array}{lll} a_0 = 1.38629 \ 44 & a_1 = 0.11197 \ 23 & a_2 = 0.07252 \ 96 \\ b_0 = 0.5 & b_1 = 0.12134 \ 78 & b_2 = 0.02887 \ 29 \end{array}.$$

provides an approximation of known precision with only a few terms, and is often very useful in its own right or as a check on numerical quadrature.

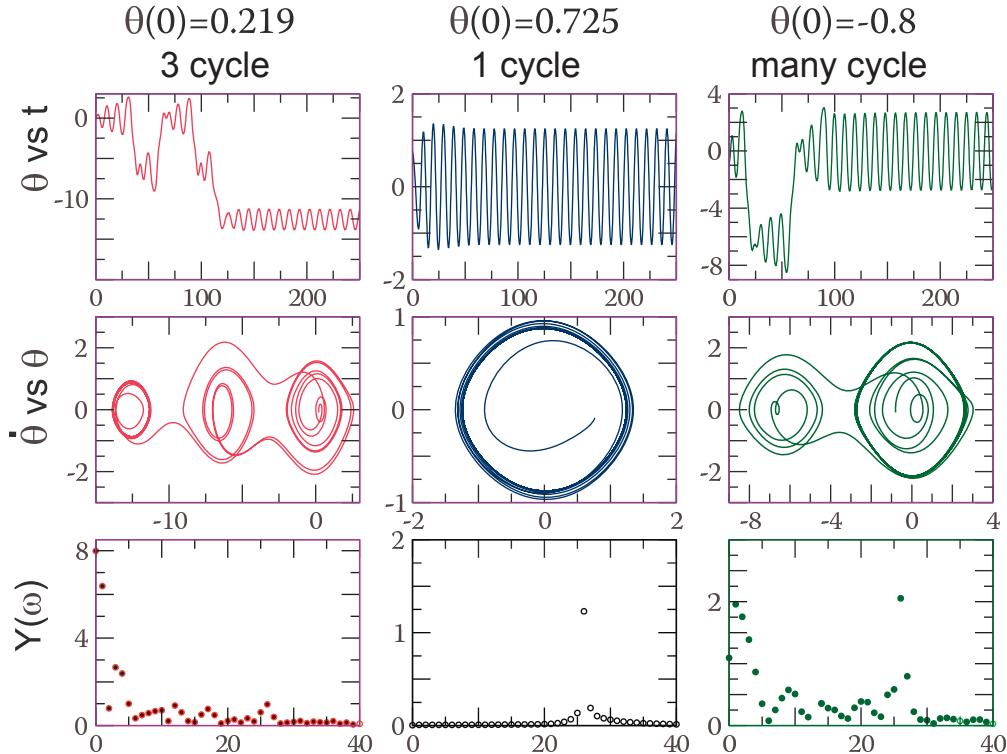
1. Compute  $K(m)$  by evaluating its integral representation numerically. Tune your evaluation until you obtain agreement at the  $\leq 3 \times 10^{-5}$  level with the polynomial approximation.
2. Use numerical quadrature to determine the ratio  $T/T_0$  for five values of  $\theta_m$  between 0 and  $\pi$ . Show that you have attained at least four places of accuracy by progressively increasing the number of integration points until changes occur only in the fifth place, or beyond.
3. Now use the power series (3.20) to determine the ratio  $T/T_0$ . Continue summing terms until changes in the sum occur only in the fifth place, or beyond and note the number of terms needed.
4. Plot the values you obtain for  $T/T_0$  versus  $\theta_m$  for both the integral and power series solutions. Note that any departure from 1 indicates breakdown of the familiar small-angle approximation for the pendulum.

### 3.3.2 Period Algorithm

1. Devise an algorithm to determine the period  $T$  of a realistic pendulum or of a nonlinear oscillator by recording times at which the pendulum passes through the origin  $\theta = 0$ . Because the oscillations may be asymmetric, you will need to record at least three times to deduce the period.
2. Verify that realistic pendula are *nonisochronous*, that is, that oscillations with different initial amplitudes have different periods.
3. Construct a graph of the deduced period *versus* initial amplitude.
4. Compare the graph of the deduced period *versus* initial amplitude deduced from your simulation to that predicted by numerical evaluation of the elliptic integral in (3.18).

### 3.3.3 Phase Space Orbits

1. Plot  $[\theta(t), d\theta/dt]$  for a large range of time  $t$  values for the linear and for the nonlinear pendulum (no torque, no friction). The geometric figures obtained are called phase space portraits or orbits.
  - a. Obtain orbits for small energies as well as for energies large enough for the pendulum to go over the top. (Figure 3.9)
  - b. Indicate the hyperbolic points, that is, points through which trajectories flow in and out.
  - c. Plot the gravitational torque on the pendulum as a function of  $\theta$  and relate it to your phase-space plot. Align vertically the phase-space plot with a plot of the torque *versus* angle so that both have the same abscissa.
2. Use your numerical solution to produce the phase-space orbits of the nonlinear pendulum with friction, though no driving force and compare to frictionless solutions.
3. Determine analytically the value of  $f$  for which the *average* energy dissipated by friction during one cycle is balanced by the energy put in by the driving force during that cycle. This is a stable configuration.
4. Show by computation that when the above condition is met, there arises a *limit cycle* near the phase space origin.
5. Sometimes you may have position data for a dynamical system, but not velocity data (or the positions are too chaotic to attempt forming derivatives). In cases like these, you can produce an alternative phase-space plot by plotting  $q(t + \tau)$  *versus*  $q(t)$ , where  $t$  is the time and  $\tau$  is a convenient lag time chosen as some fraction of a characteristic time for the system [Abarbanel et al.(93)]. Create



**Figure 3.9.** From top to bottom, position *versus* time, phase space plot, and Fourier spectrum for a chaotic pendulum with  $\omega_0 = 1$ ,  $\alpha = 0.2$ ,  $f = 0.52$ , and  $\omega = 0.666$  and, from left to right, three different initial conditions. (The leftmost column displays three dominant cycles, the center column only one, while the rightmost column has multiple cycles.)

a phase space plot from the output of your realistic or chaotic pendulum by plotting  $\theta(t + \tau)$  *versus*  $\theta(t)$  for a large range of  $t$  values.

- Explore how the graphs change for different values of the lag time  $\tau$ .
- Compare your results to the conventional phase space plots you obtained previously for the same parameters.

6. Extend your ODE solution to the chaotic pendulum with parameters

$$f = 0.52, \quad \alpha = 0.2, \quad \omega_0 = 1. \quad (3.22)$$

- Using your previously tested ODE solver, create phase-space orbits by plotting  $[\theta(t), d\theta/dt(t)]$  for long time intervals (Figure 3.9).
- Indicate which part of the orbits are transients.

- c. Correlate phase-space structures with the behavior of  $\theta(t)$  by also plotting  $\theta$  versus  $t$  (preferably next to  $d\theta/dt$  versus  $\theta$ ).
  - d. Gain some physical intuition about the flow in phase space by watching how it builds up with time.
7. For the second part of the chaotic pendulum study, use the same parameters as in first part, though now sweep through a range of  $\omega$  values.
- a. Use initial conditions:  $d\theta(0)/dt = 0.8$ , and  $\theta(0) = -0.0888$ .
  - b. Verify that for  $\omega \simeq 0.6873$  there is a period-three limit cycle where the pendulum jumps between three orbits in phase space.
  - c. Verify that for  $\omega \simeq 0.694 - 0.695$  there are running solutions in which the pendulum goes over the top multiple times. Try to determine how many rotations are made before the pendulum settles down.
  - d. For  $\omega \simeq 0.686$  and long times, the solutions for very slightly different initial conditions, tend to fill in bands in phase space. If needed, decrease your time step and try to determine how the bands get filled, in particular, just how small a difference in  $\omega$  values separates the regular and the chaotic behaviors.
8. Create a Poincaré map for the chaotic pendulum.

### 3.3.4 Vibrating Pivot Pendulum

As an alternative to what we have called the chaotic pendulum, repeat the pendulum analysis for another version of the chaotic pendulum, this one with a vibrating pivot point (in contrast to our usual sinusoidal external torque):

$$\frac{d^2\theta}{dt^2} = -\alpha \frac{d\theta}{dt} - (\omega_0^2 + f \cos \omega t) \sin \theta. \quad (3.23)$$

Essentially, the acceleration of the pivot is equivalent to a sinusoidal variation of  $g$  or  $\omega_0^2$  [Landau & Lifshitz(77), DeJong(92), Gould et al.(06)]. The scatterplot in Figure 3.10 displays a sampling of  $\dot{\theta}$  as a function of the magnitude of the vibrating pivot point.

## 3.4 Fourier Analysis of Oscillations

1. Consider a particle oscillating in the nonharmonic potential of (3.6):

$$V(x) = \frac{1}{p} k|x|^p, \quad p \neq 2. \quad (3.24)$$

While nonforced oscillations in this potential are always periodic, they are not sinusoidal.

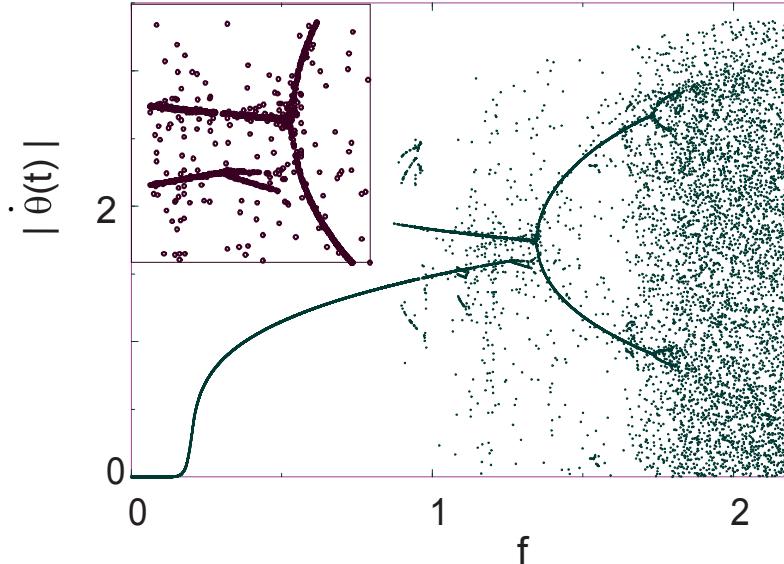
- a. For  $p = 12$ , decompose the solution  $x(t)$  into its Fourier components.

- b. Determine the number of components that contribute at least 10%.
  - c. Check that resuming the components reproduces the input  $x(t)$ .
2. Recall the perturbed harmonic oscillator (3.5):
- $$V(x) = \frac{1}{2}kx^2(1 - \frac{2}{3}\alpha x) \quad \Rightarrow \quad F(x) = -kx(1 - \alpha x). \quad (3.25)$$
- For small oscillations ( $x \ll 1/\alpha$ ),  $x(t)$  should be well approximated by solely the first term of the Fourier series.
- a. Fix your value of  $\alpha$  and the maximum amplitude of oscillation  $x_{\max}$  so that  $\alpha x_{\max} \simeq 10\%$ . Plot up resulting  $x(t)$  along with a pure sine wave.
  - b. Decompose your numerical solution into a discrete Fourier spectrum.
  - c. Make a semilog plot of the power spectrum  $|Y(\omega)|^2$  as a function of  $x_{\max}$ . Because power spectra often varies over several orders of magnitude, a semi-log plot is needed to display the smaller components.
  - d. As always, check that summation of your transform reproduce the signal.
3. For cases in which there are one-, three-, and five-cycle structures in phase space (Figure 3.9), store your post-transients solutions for the chaotic pendulum, or for the double pendulum.
4. Perform a Fourier analysis of  $x(t)$ . Does it verify the statement that “the number of cycles in the phase-space plots corresponds to the number of major frequencies contained in  $x(t)$ ”?
5. See if you can deduce a relation among the Fourier components, the natural frequency  $\omega_0$ , and the driving frequency  $\omega$ .
6. Examine your system for parameters that give chaotic behavior and plot the power spectrum in a semi-logarithmic plot. Does this verify the statement that “a classic signal of chaos is a broad Fourier spectrum”?

### 3.4.1 Pendulum Bifurcations

Fourier analysis and phase-space plots indicate that a chaotic system contains a number of dominant frequencies, and that the system tends to “jump” from one frequency to another. In contrast to a linear system in which the Fourier components occur simultaneously, in nonlinear systems the dominant frequencies may occur sequentially. Thus a sampling of the instantaneous angular velocity  $\dot{\theta} = d\theta/dt$  of the chaotic pendulum for a large number of times indicates the frequencies to which the system is *attracted*, and accordingly should be related to the system's Fourier components.

1. Make a scatter plot of the sampled  $\dot{\theta}$ s for many times as a function of the magnitude of the driving torque.



**Figure 3.10.** A bifurcation diagram for the damped pendulum with a vibrating pivot (see also the similar diagram for a double pendulum, Figure 3.12 right). The ordinate is  $|d\theta/dt|$ , the absolute value of the instantaneous angular velocity at the beginning of the period of the driver, and the abscissa is the magnitude of the driving torque  $f$ . Note that the heavy line results from the overlapping of points, not from connecting the points (see enlargement in the inset).

2. For each value of  $f$ , wait 150 periods of the driver before sampling to permit transients to die off. Then sample  $\dot{\theta}$  for 150 times at the instant the driving force passes through zero.
3. Plot values of  $|\dot{\theta}|$  versus  $f$  as unconnected dots.

### 3.4.2 Sonification

Human ears respond to air vibrations in the approximate frequency range 20–20,000Hz, which our brain interprets as sound. So to create a sonification of an oscillation we need to map the oscillation's frequencies into the range of human hearing. Python and Matlab have utilities to create sound files from arrays, as do third party software packages such as *Audacity*. In Python, the function `scipy.io.wavfile` is used to create a `wav` file, for example,

```
import numpy as np
from scipy.io.wavfile import write
data = np.random.uniform(-1,1,44100)           # Random samples -1 < r < 1
```

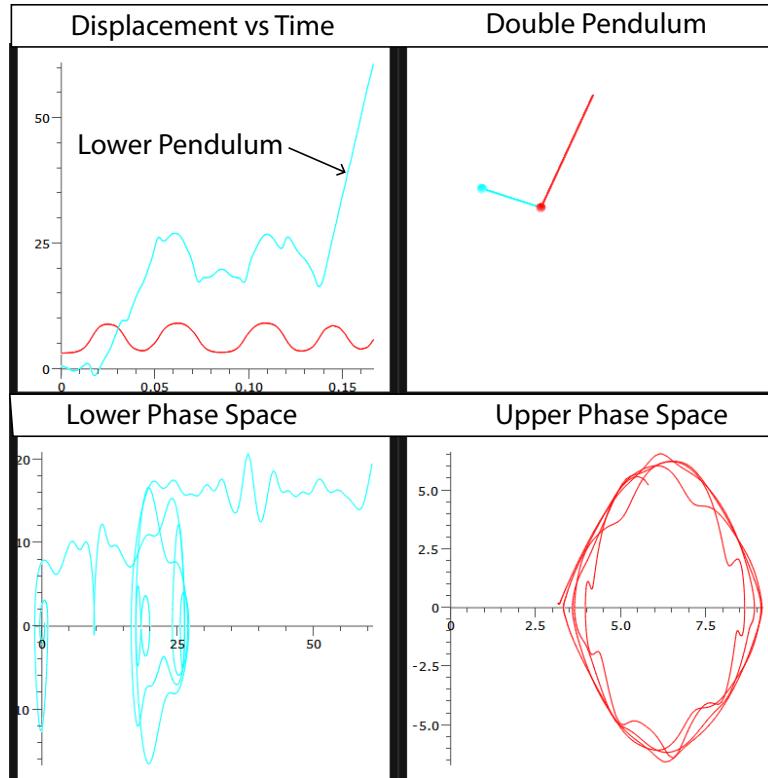


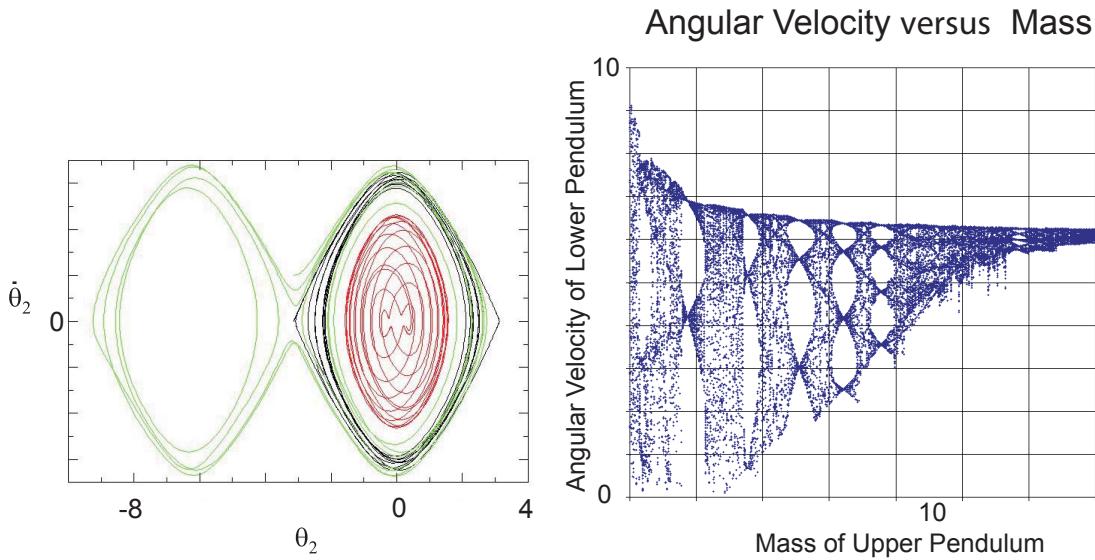
Figure 3.11. A large-angle, chaotic double pendulum (lower bob started at top).

```
scaled = np.int16(data/np.max(np.abs(data)) * 32767)
write('test.wav', 44100, scaled)
```

Sonify an harmonic and a nonharmonic oscillation, and listen to the differences. The overtones (higher harmonics) in the nonharmonic oscillation should make it sound more interesting.

### 3.5 The Double Pendulum

Repeat the preceding study of the chaotic pendulum but now do it for the double pendulum with no small angle approximations. This is a compound system in which a lower pendulum is attached to the bottom of an upper pendulum (Figure 3.11 top right). On the left of Figure 3.12 we see a phase space diagram for one bob undergoing fairly regular motion. Because each pendulum acts as a driving force for the other, we need not include an external driving torque to produce a chaotic



**Figure 3.12.** *Left:* Phase space trajectories for a double pendulum with  $m_1 = 10m_2$  and with two dominant attractors. *Right:* A bifurcation diagram for the double pendulum displaying the instantaneous velocity of the lower pendulum as a function of the mass of the upper pendulum. (Both plots courtesy of J. Danielson.)

system. In Figure 3.11 (and the animation DoublePend.mp4) we show visualizations for a chaotic double pendulum in which the lower pendulum started off upright.

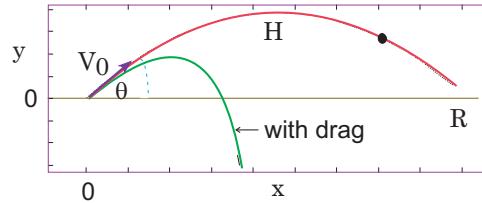
1. Show that the Lagrangian for the double pendulum is:

$$\begin{aligned} L = \text{KE} - \text{PE} = & \frac{1}{2}(m_1 + m_2)l_1^2\dot{\theta}_1^2 + \frac{1}{2}m_2l_2^2\dot{\theta}_2^2 \\ & + m_2l_1l_2\dot{\theta}_1\dot{\theta}_2 \cos(\theta_1 - \theta_2) + (m_1 + m_2)gl_1 \cos \theta_1 + m_2gl_2 \cos \theta_2. \end{aligned} \quad (3.26)$$

2. Use this Lagrangian to show that the equations of motion are

$$\begin{aligned} (m_1 + m_2)L_1\ddot{\theta}_1 + m_2L_2\ddot{\theta}_2 \cos(\theta_2 - \theta_1) &= m_2L_2\dot{\theta}_2^2 \sin(\theta_2 - \theta_1) - (m_1 + m_2)g \sin \theta_1 \\ L_2\ddot{\theta}_2 + L_1\ddot{\theta}_1 \cos(\theta_2 - \theta_1) &= -L_1\dot{\theta}_1^2 \sin(\theta_2 - \theta_1) - g \sin \theta_2. \end{aligned} \quad (3.27)$$

3. Deduce the equations of motion for small displacement of each pendulum from its equilibrium position (usually what's found in text books).
4. Deduce analytically the frequencies of slow and fast modes for the small angle oscillations.



**Figure 3.13.** Schematics of the trajectories of a projectile fired with initial velocity  $V_0$  in the  $\theta$  direction. The lower curve includes air resistance.

5. Solve the equations of motion numerically without making any small angle approximations.
6. Verify that your numerical solution has a slow mode in which  $\theta_1 = \theta_2$  and a fast mode in which  $\theta_1$  and  $\theta_2$  have opposite signs.
7. Reproduce the phase space plots on the left in Figure 3.12 describing the motion of the lower pendulum for  $m_1 = 10m_2$ . When given enough initial kinetic energy to go over the top, the trajectories are seen to flow between two major attractors with energy being transferred back and forth between the pendula.
8. Reproduce the bifurcation diagram for the double pendulum shown on the right in Figure 3.12. This is created by sampling the set of instantaneous angular velocity  $\dot{\theta}_2$  of the lower pendulum as it passes through its equilibrium position, and plotting the set as a function of the mass of the upper pendulum. The resulting structure is fractal with the bifurcations indicative of the dominant Fourier components.
9. Compute the Fourier spectrum for the double pendulum with the same parameters used for the bifurcation plot. Do the two plots correlate?

### 3.6 Realistic Projectile Motion

Figure 3.13 shows trajectories for a projectile shot at inclination  $\theta$  and with an initial velocity  $V_0$ . If we ignore air resistance, the projectile has only the force of gravity acting on it and the trajectory will be a parabola with range  $R = 2V_0^2 \sin \theta \cos \theta / g$  and maximum height  $H = \frac{1}{2}V_0^2 \sin^2 \theta / g$ . Because a parabola is symmetric about its midpoint it does not describe what appears to be a sharp nearly vertical, drop off of baseballs and golf balls near the end of their trajectories.

1. Investigate several models for the frictional force:

$$\mathbf{F}^{(f)} = -k m |v|^n \frac{\mathbf{v}}{|v|}. \quad (3.28)$$

Here the  $-\mathbf{v}/|v|$  factor ensures that the frictional force is always in a direction opposite that of the velocity.

- Show that for our model of friction, the equations of motion are

$$\frac{d^2x}{dt^2} = -k v_x^n \frac{v_x}{|v|}, \quad \frac{d^2y}{dt^2} = -g - k v_y^n \frac{v_y}{|v|}, \quad |v| = \sqrt{v_x^2 + v_y^2}. \quad (3.29)$$

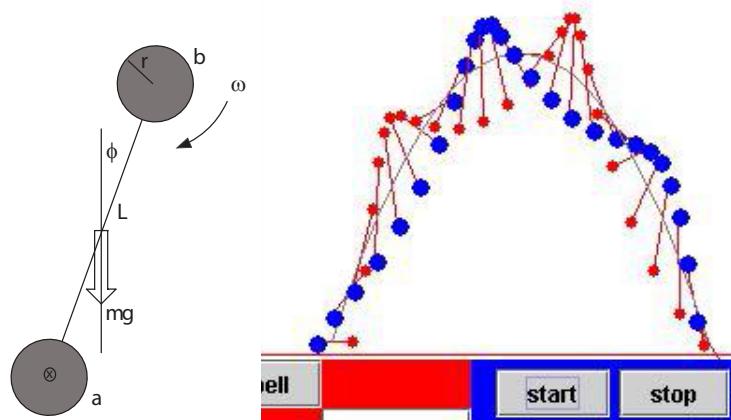
- Consider three values for  $n$ , each of which represents a different model for the air resistance: 1)  $n = 1$  for low velocities, 2)  $n = 3/2$  for medium velocities, and 3)  $n = 2$  for high velocities.
- Modify your `rk4` program so that it solves the simultaneous ODEs for projectile motion (3.29) with friction ( $n = 1$ ). In Listing 3.4 we present the program `ProjectileAir.py` that solves for the trajectory using a form of the velocity–Verlet algorithm accurate to second order in time. Here is its pseudo code:

```
# Pseudocode for ProjectileAir.py: Projectile motion with air resistance
Import visual library
Initialize variables
Compute analytic R, T, & H
Initialize 2 graphical curves
Define plotNumeric
    loop over times
        compute & output numeric (x,y)
Define plotAnalytic
    loop over times
        compute & analytic numeric (x,y)
Plot up numeric & analytic results
```

- What conclusions can you draw as to the effects of air resistance on the shape of the projectile?
- The model (3.28) with  $n = 1$  is applicable only for low velocities. Now modify your program to handle  $n = 3/2$  (medium-velocity friction) and  $n = 2$  (high-velocity friction). Adjust the value of  $k$  for the latter two cases such that the initial force of friction  $k v_0^n$  is the same for all three cases.
- Solve the equations of motion for several initial conditions and powers  $n$ .
- How does friction affect the range  $R$  and the time aloft  $T$ ?
- What conclusion can you draw regarding the observation of balls appearing to fall straight down out of the sky?

### 3.6.1 Trajectory of Thrown Baton

Classical dynamics describes the motion of the baton as the motion of an imaginary point particle located at the center of mass (CM), plus a rotation about the CM. Because the translational and rotational motions are independent, each may be determined separately. This is made easier in the absence of air resistance, since then there are no torques on the baton and the angular velocity  $\omega$  about the CM is constant.

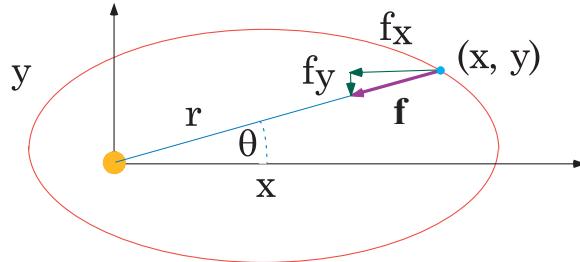


**Figure 3.14.** *Left:* The initial conditions for a baton as it is thrown. *Right:* The motion of the entire baton as its CM follows a parabola.

1. The baton in Figure 3.14 is let fly with an initial velocity corresponding to a rotation about the center of the lower sphere. Extend the program developed for projectile motion of a point particle to one that computes the motion of the baton as a function of time. Ignore air resistance, assume that the bar is massless, and choose values for all of the parameters.
  - a. Plot the position of each end of the baton as functions of time. Figure 3.14 shows a typical result.
  - b. Plot the translational kinetic energy, the rotational kinetic energy, and the potential energy of the baton, all as functions of time.
  - c. Solve for the motion of the baton with an additional lead weight at its center, again making separate plots of each end of the baton.

### 3.7 Bound States

In contrast to the analytic case, the numerical solutions to orbit problems are straightforward. A simple approach is to express the forces and equations of motion in Cartesian coordinates (Figure 3.15), and then to solve the resulting two simultaneous ODEs:



**Figure 3.15.** The gravitational force on a planet a distance  $r$  from the sun. The  $x$  and  $y$  components of the force are indicated.

$$\mathbf{F} = m\mathbf{a} = m \frac{d^2\mathbf{x}}{dt^2} \quad \Rightarrow \quad (3.30)$$

$$F_x = F^{(g)} \cos \theta = F^{(g)} \frac{x}{\sqrt{x^2 + y^2}}, \quad F_y = F^{(g)} \sin \theta = F^{(g)} \frac{y}{\sqrt{x^2 + y^2}}, \quad (3.31)$$

$$\frac{d^2x}{dt^2} = -GM \frac{x}{(x^2 + y^2)^{3/2}}, \quad \frac{d^2y}{dt^2} = -GM \frac{y}{(x^2 + y^2)^{3/2}}. \quad (3.32)$$

1. Show that in order to apply **rk4** to simultaneous ODEs, we need only increase the dimension of the vectors in the dynamical form of the equation of motion from two to three:

$$y^{(0)} = x(t), \quad y^{(1)} = \frac{dx(t)}{dt} = \frac{dy^{(0)}}{dt}, \quad (3.33)$$

$$y^{(2)} = y(t), \quad y^{(3)} = \frac{dy(t)}{dt} = \frac{dy^{(2)}}{dt}, \quad (3.34)$$

$$\Rightarrow \quad f^{(0)} = y^{(1)}(t), \quad f^{(1)} = \frac{F_x(\mathbf{y})}{m}, \quad f^{(2)} = y^{(3)}(t), \quad f^{(3)} = \frac{F_y(\mathbf{y})}{m}. \quad (3.35)$$

2. What are the explicit expressions for  $f^{(2)}$  and  $f^{(3)}$  in terms of the  $y^{(i)}$ s?
3. Modify your ODE solver program to solve the equations of motion.
  - a. Assume units such that  $GM = 1$  and the initial conditions

$$x(0) = 0.5, \quad y(0) = 0, \quad v_x(0) = 0.0, \quad v_y(0) = 1.63. \quad (3.36)$$

- b. Check that you are using small enough time steps by verifying that the orbits remain closed and fall upon themselves for long periods of time.
- c. Experiment with the initial conditions until you find the ones that produce a circular orbit (a special case of an ellipse).

- d. Progressively increase the initial velocity until the orbits become unbound.
  - e. Using the same initial conditions that produce elliptical orbits, investigate the effect of varying continuously the power in Newton's law of gravitation from two. Even small changes should cause the orbits to precess, as predicted by general relativity.
4. Consider the motion of a particle of mass  $m$  with angular momentum  $l$  in an inverse-square force field subjected to an inverse-cube perturbation:

$$\mathbf{F} = \left( \frac{-k}{r^2} + \frac{C}{r^3} \right) \hat{\mathbf{e}}_r \quad (3.37)$$

The solutions to the orbit equation fall into three classes depending on the constant  $C$ :

$$|C| < l^2/m, \quad |C| = l^2/m, \quad |C| > l^2/m. \quad (3.38)$$

- a. Solve the equations of motion numerically for each of the three conditions given in (3.38).
  - b. Indicate which of your solutions are bound, that is, the particle remains localized to some region of space.
  - c. What are the conditions for the solutions to look like ellipses with a slow rate of precession?
5. A mass is in a circular orbit about an attractive potential  $U(r)$ .
- a. Find an analytic expression for the frequency of oscillations for small radial perturbations about the circular orbit.
  - b. Consider now the potential

$$U(r) = \frac{-k}{r^{n-1}}, \quad (3.39)$$

with  $n$  an integer. Prove analytically that the angle through which the orbit rotates as  $r$  varies from its minimum to maximum value is  $\pi/\sqrt{3-n}$ .

- c. Solve numerically for the orbits for various values of  $n$  and plot your results.
- d. Is it true that the orbit returns to itself only for  $n = 2$ ?
- e. Plot the *phase space portraits* for various combinations of variables for oscillations about a circular orbit, and for various values of  $n$ .

6. A particle is confined to a 2-D square well potential of depth  $W$  and radius  $R$ ,

$$V(r) = -W\theta(R - r). \quad (3.40)$$

- a. Solve for the 2-D orbits of a particle of mass  $m = 1$  within this square well, either by deriving the appropriate orbit equation, or geometrically by using a ruler, a pencil, and a piece of paper.

- b. Explain why there are problems solving for a square well potential numerically. (*Hint:* Think about derivatives.)
- c. As an approximation to a square well, try the potential

$$V(r) = V_0 r^{10}, \quad (3.41)$$

which is small for  $r < 1$ , though gets large rapidly for  $r > 1$ .

- d. Start by looking at  $x(t)$  and  $y(t)$  and making sure that they are reasonable (like a free particle for small  $x$  or small  $y$ , and then like a particle hitting a wall for larger  $x$  or  $y$ ).
- e. Next look at the trajectories  $[x(t), y(t)]$  and see if they seem close to what you might expect for a mass reflecting off the walls of a circular cavity.
- f. Evaluate the angular momentum  $l(t)$  and the total energy of the mass  $E(t)$  as functions of time and determine their level of constancy.
- 7. Solve for the orbits of a particle of mass  $m = 1$  confined within a 2-D racetrack shaped (ellipse-like) potential,

$$V(r) = ax^n + by^n, \quad (3.42)$$

where you are free to choose values for the parameters  $a$ ,  $b$ , and  $n$ .

- a. Test your program for the two cases ( $b = 0, n = 2$ ) and ( $a = 0, n = 2$ ). You should obtain simple harmonic motion with frequency  $\omega_0 = \sqrt{a}$  and  $\sqrt{b}$ .
- b. Verify that the orbits are symmetric in the  $xy$  plane.
- c. Verify that the angular momentum and energy of the particle are constants as functions of time for all  $a = b$  values and for all values of  $n$ .
- d. Plot the orbits for several  $a \neq b$  cases.
- e. Check that for large  $n$  values the orbits look like internal reflections from the racetrack walls.
- f. Search for those combinations of  $(a, b)$  values for which the orbits close on themselves.
- g. Search for those combinations of  $(a, b)$  values for which the orbits precess slowly.
- h. Evaluate the energy and angular momentum as functions of time for an  $a \neq b$  case, and comment on their variability.

### 3.8 Three-Body Problems: Neptune, Two Suns, Hénon-Heiles

The planet Uranus was discovered in 1781 by William Herschel, and found to have an orbital period of approximately 84 years. Never the less, even before it had completed an entire orbit around the sun in the time since its discovery, it was observed that Uranus's orbit was not precisely that predicted by Newton's law of gravity. Accordingly, it was hypothesized that a yet-to-be-discovered and distant planet was perturbing Uranus's orbit. The predicted planet is called Neptune.

Use these data for the calculation:

	Mass ( $10^{-5}$ Solar Masses)	Distance (AU)	Orbital Period (Years)	Angular Position (in 1690)
Uranus	4.366244	19.1914	84.0110	$\sim 205.64^\circ$
Neptune	5.151389	30.0611	164.7901	$\sim 288.38^\circ$

You may enter these data into your program much as we did:

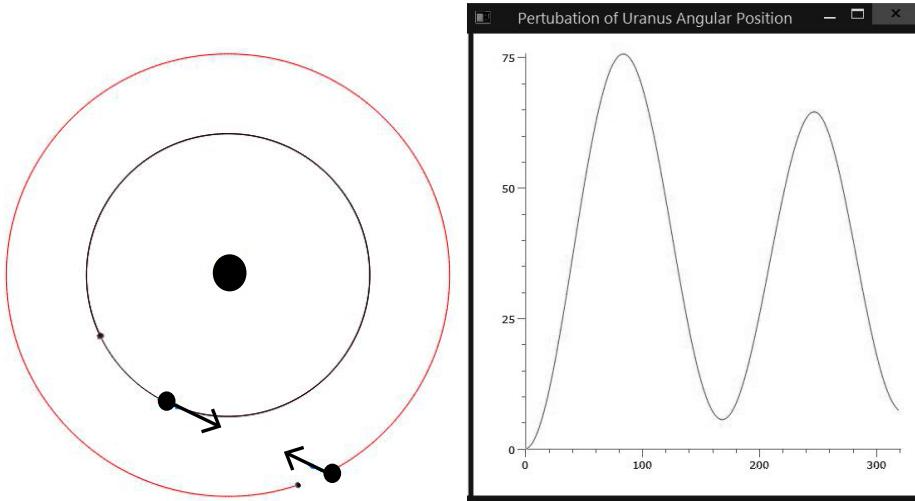
```
G = 4*pi*pi (AU, Msun=1) mu = 4.366244e-5 = Uranus mass
M = 1.0 = Sun mass mn = 5.151389e-5 = Neptmass
du = 19.1914 = Uran Sun dist dn = 30.0611 = N sun dist
Tur = 84.0110 = Uran Period Tnp = 164.7901 = N Period
omeur = 2*pi/Tur = U omega omennp = 2*pi/Tnp = N omega
omreal = omeur (UA/yr) urvel = 2*pi*du/Tur = Uran omegal
npvel = 2*pi*dn/Tnp = Nept omega UA/yr radur = 205.64*pi/180
urx = du*cos(radur) = init x 1690 ury = du*sin(radur) = init y 1690
urvelx = urvel*sin(radur) urvely = -urvel*cos(radur)
radnp = 288.38*pi/180. = Nept angular pos.
```

1. Use rk4 to compute the variation in angular position of Uranus with respect to the Sun as a result of the influence of Neptune during one complete orbit of Neptune. Consider only the forces of the Sun and Neptune on Uranus. Use astronomical units in which  $M_s = 1$  and  $G = 4\pi^2$ .
2. As shown in Figure 3.16, assume that the orbits of Neptune and Uranus are circular and coplanar, and that the initial angular positions with respect to the  $x$  axis are as in the table above.

### 3.8.1 Two Fixed Suns with a Single Planet

The three-body problem in which three particles move via pairwise interactions can be complicated and chaotic. Here we ask you to examine a simple version in which two heavy stars 1 and 2 are kept at a fixed separation along the  $x$  axis, while a lighter planet moves about them [Row 2004]. We use natural units  $G = 1$  to keep the calculations simpler, and treat all bodies as point particles. It is best to view the output as animations so that you can actually see the planet pass through a number of orbits. A characteristic of this kind of chaotic system is that there are periods with smooth precessions followed by chaotic behavior, and then smooth precession again. This means that we on earth are lucky having only one sun as this makes the year of reliably constant length [Liu(14)].

1. Start with  $M_1 = M_2 = 1$  and the planet at  $(x, y)_0 = (0.4, 0.5)$  with  $(v_x, v_y)_0 = (0, -1)$
2. Set  $M_2 = 2$  and see if the planet remains in a stable orbit about sun 2.



**Figure 3.16.** A screenshot of an animation from `UranusNeptune.py` showing: *Left:* The orbits of Uranus (inner circle) and of Neptune (outer circle) with the sun in the center. The arrow on the outer circle indicate Neptune's perturbation on Uranus. *Right:* The perturbation on Uranus as a function of angular position.

3. Return to the equal mass case and investigate the effect of differing initial velocities.
4. Make  $M_1$  progressively smaller until it acts as just a perturbation on the motion around planet 2, and see if the year now becomes of constant length.
5. What might be the difficulty of having life develop and survive in a two sun system?
6. Explore the effect on the planet of permitting one of the suns to move.

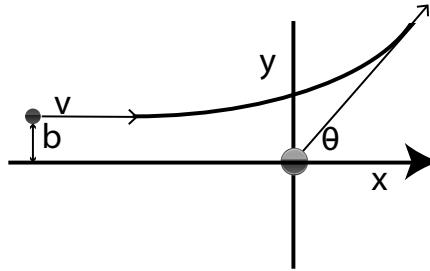
### 3.8.2 Hénon-Heiles Bound States

The Hénon-Heiles potential

$$V(x, y) = \frac{1}{2}x^2 + \frac{1}{2}y^2 + x^2y - \frac{1}{3}y^3 \quad (3.43)$$

is used to model the interaction of three close astronomical objects. The potential binds the objects near the origin though releases them if they move far out.

1. Show that the minimum in the potential for  $x = 0$  occurs at  $y = 1$
2. Show that the value of the potential at its minimum implies bound orbits occur for energies  $0 < E < 1/6$ .



**Figure 3.17.** A particle with impact parameter  $b$  incident along the  $y$  axis is scattered by a force center at the origin into a scattering angle  $\theta$ .

3. Show that the equations of motion following from Hamiltonian equations are:

$$\frac{dp_x}{dt} = -x - 2xy, \quad \frac{dp_y}{dt} = -y - x^2 + y^2, \quad \frac{dx}{dt} = p_x, \quad \frac{dy}{dt} = p_y. \quad (3.44)$$

4. Solve for the position  $[x(t), y(t)]$  for a number of initial conditions.
5. Verify that the orbits are bounded for energies less than  $1/6$ .
6. As on check on your numerics, verify that the Hamiltonian remains constant.
7. Produce a Poincare section by creating a  $(y, p_y)$  plot, adding a point each time  $x \simeq 0$ . With the energy fixed, make several plot for different initial conditions.
8. Isolate a smaller region of phase space and look for unusual structures.

## 3.9 Scattering

### 3.9.1 Rutherford Scattering

A particle of mass  $m = 1$  and velocity  $v$  is scattered by the force center

$$V(r) = \frac{\alpha}{r^2} \quad (3.45)$$

with  $\alpha$  positive. As seen in Figure 3.17, the particle starts at the left ( $x = -\infty$ ) with an impact parameter  $b$  (distance above the  $x$  axis) and is scattered into an angle  $\theta$ . Because the force center does not recoil, the magnitude of the initial and final velocities are the same, though their directions differ.

1. Calculate and plot the positions  $[x(t), y(t)]$  for a range of impact parameters  $b$  starting from negative values and proceeding to positive ones, and for a range of velocities  $v$ .

2. For fixed angular momentum, for what values of  $\alpha$  does the particle make one and two revolutions about the center of force before moving off?
3. See if you can find an orbit that collapses into the origin  $r = 0$ . This should be possible for angular momentum  $l^2 < 2m\alpha$ .
4. Use a central difference approximation to calculate the derivative  $d\theta/db$  as a function of  $\theta$ .
5. Deduce the scattering angle for a trajectory by examining the trajectory of the scattered particle at a large enough distance from the target so that the potential no longer has much effect, say  $|PE|/KE \leq 10^{-6}$ . The scattering angle is then deduced from the components of velocity,

$$\theta = \tan^{-1}(v_y/v_x) = \text{math.atan2}(y, x). \quad (3.46)$$

Here `atan2` computes the arctangent in the correct quadrant and eliminates the possibility of division by zero when computing  $\tan^{-1}(y/x)$ .

6. Calculate the differential scattering cross section from the dependence of the scattering angle upon the classical impact parameter  $b$  [Marion & Thornton(03)]:

$$\sigma(\theta) = \left| \frac{d\theta}{db} \right| \frac{b}{\sin \theta(b)}. \quad (3.47)$$

7. In units for which the Coulomb potential between a target of charge  $Z_T e$  and a projectile of charge  $Z_P e$  is

$$V(r) = \frac{Z_T Z_P e^2}{r}, \quad (3.48)$$

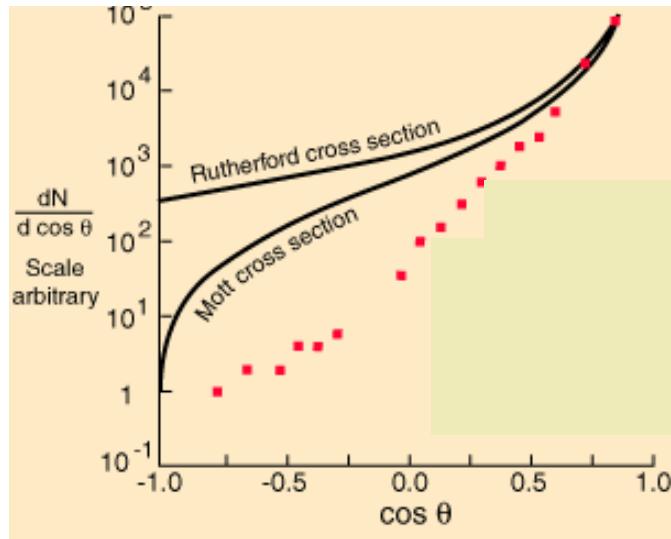
the cross section for pure Coulomb scattering is given by the Rutherford formula

$$\sigma(\theta)_R = \left( \frac{Z_T Z_P e^2}{4E \sin^2 \theta/2} \right)^2, \quad (3.49)$$

where  $E = p^2/2\mu$  is essentially the projectile's energy, with  $p$  the COM momentum, and  $\mu$  the reduced target-projectile mass (for a fixed scattering center, the target mass is infinite, and so  $\mu = m_P$ ) [Landau(96)]. Compare the  $\theta$  dependence of your computed results to that of the Rutherford cross section.

### 3.9.2 Mott Scattering

Rutherford scattering is appropriate for the scattering of two spinless particles (no magnetic moments). In many important applications, high energy electrons are scattered from the Coulomb field of heavy nuclei and the magnetic moment (spin) of the



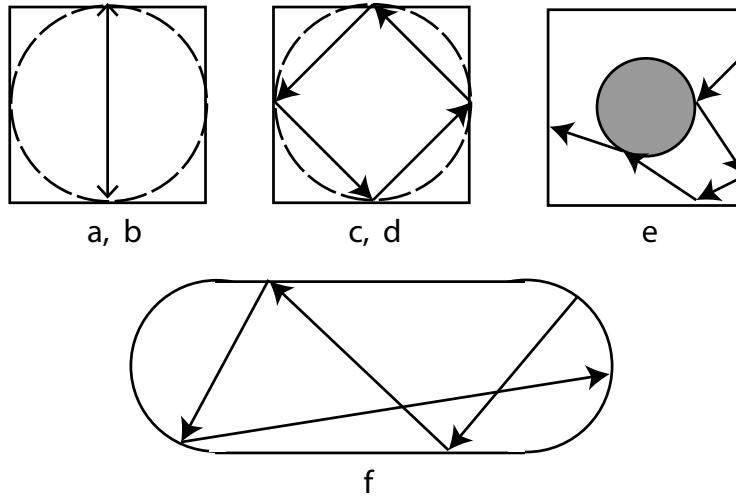
**Figure 3.18.** Experimental results for high-energy electron scattering from a nucleus showing deviations from assumption of a point nucleus.

electron interacts with the Coulomb field. This is called Mott scattering and leads to a multiplicative correction term being affixed to the Rutherford cross section:

$$\sigma(\theta)_{Mott} = \sigma(\theta)_R \left( 1 - \frac{v^2}{c^2} \sin^2 \theta / 2 \right). \quad (3.50)$$

1. Compute and plot the Mott and Rutherford differential cross sections for the scattering of 150 MeV electrons from gold nuclei.
2. At what angle does the difference reach 50%?
3. At what angle does the difference reach 10%?
4. It was found experimentally by Hofstadter *et al.* that the reduction in 150° cross section for 150 MeV electrons scattering was 1000 times greater than that predicted by the Mott formula. Apparently, there must be a reduction in the strength of the Coulomb potential from that given by (3.48). This reduction could arise from the electron penetrating into the nucleus, and accordingly not being affected by all of the nucleus's electric charge. Quantum mechanics tells us that the correction for the finite nuclear size is approximately

$$\sigma(\theta)_{finite} \simeq \sigma(\theta)|_{Mott} \left( 1 - \frac{q^2 R_{rms}^2}{6\hbar^2} \right)^2, \quad (3.51)$$



**Figure 3.19.** Square (a, c), circular (b, d), Sinai (e), and stadium billiards (f). The arrows are trajectories. The stadium billiard has two semicircles on the ends.

where  $q^2 = 2p^2(1 - \sin \theta)$  is the scattered electron's momentum transfer and  $R_{rms}$  is the root mean square radius of the nucleus. Based on this reduction in the cross section, what would you estimate as the size of the gold nucleus?

### 3.9.3 Chaotic Scattering

- One expects the scattering of a projectile from a force center to be a continuous process. Never the less in some situations the projectile undergoes multiple internal scatterings and ends up with a final trajectory that seems unrelated to the initial one. Specifically, consider the 2-D potential [Blehel et al.(90)]

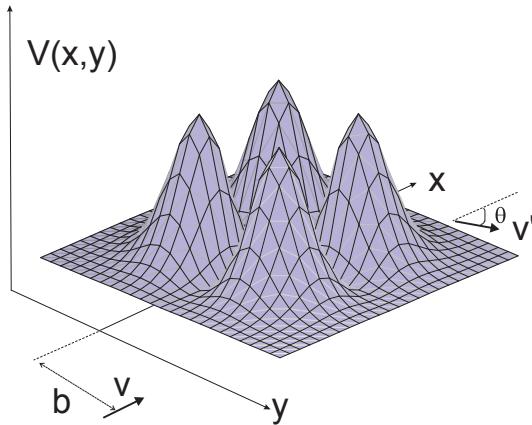
$$V(x, y) = \pm x^2 y^2 e^{-(x^2+y^2)}. \quad (3.52)$$

As seen in Figure 3.20, this potential has four circularly symmetric peaks in the  $xy$  plane. The two signs produce repulsive and attractive potentials, respectively.

- Show that the two simultaneous equations of motions are:

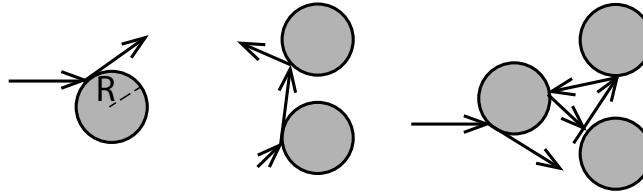
$$m \frac{d^2x}{dt^2} = \mp 2y^2 x(1 - x^2)e^{-(x^2+y^2)}, \quad m \frac{d^2y}{dt^2} = \mp 2x^2 y(1 - y^2)e^{-(x^2+y^2)}. \quad (3.53)$$

- Show that the force vanishes at the  $x = \pm 1$  and  $y = \pm 1$  peaks in Figure 3.20, that is, where the maximum value of the potential is  $V_{\max} = \pm e^{-2}$ . This sets the energy scale for the problem.



**Figure 3.20.** Scattering from the potential  $V(x,y) = x^2 y^2 e^{-(x^2+y^2)}$  which in some ways models a pinball machine. The incident velocity  $v$  is in the  $y$  direction, and the impact parameter ( $y$  value) is  $b$ . The velocity of the scattered particle is  $v'$  and its scattering angle is  $\theta$ .

- c. Apply the `rk4` method to solve the simultaneous second-order ODEs.
- d. The initial conditions are 1) an incident particle with only a  $x$  component of velocity and 2) an impact parameter  $b$  (the initial  $y$  value). You do not need to vary the initial  $x$ , though it should be large enough such that  $PE/KE \leq 10^{-6}$ , which means that the  $KE \simeq E$ .
- e. Good parameters to use are  $m = 0.5$ ,  $v_y(0) = 0.5$ ,  $v_x(0) = 0.0$ ,  $\Delta b = 0.05$ ,  $-1 \leq b \leq 1$ . You may want to lower the energy and use a finer step size once you have found regions of rapid variation in the cross section.
- f. Plot a number of trajectories  $[x(t), y(t)]$ , some being smooth and others being jumpy. In particular, try to show the apparently discontinuous changes in scattering angle
- g. Plot a number of phase space trajectories  $[x(t), \dot{x}(t)]$  and  $[y(t), \dot{y}(t)]$ . How do these differ from those of bound states?
- h. Compute the scattering angle  $\theta = \text{atan2}(Vx, Vy)$  by determining the velocity components of the scattered particle after it has left the interaction region ( $PE/KE \leq 10^{-6}$ ).
- i. Which characteristics of a trajectory lead to discontinuities in  $d\theta/db$  and thus in the scattering cross section  $\sigma(\theta)$  (3.47)?
- j. Run the simulations for both attractive and repulsive potentials and for a range of energies less than and greater than  $V_{max} = \exp(-2)$ .
- k. Another way to find unusual behavior in scattering is to compute the *time delay*  $T(b)$ , that is, the increase in the time it takes a particle to travel through the interaction region. Look for highly oscillatory regions in the semilog plot of  $T(b)$ , and once you find some, repeat the simulation at a finer scale by



**Figure 3.21.** One, two, and three stationary disks on a flat billiard table scatter point particles elastically, with some of the internal scattering leading to trapped, periodic orbits.

setting  $b \simeq b/10$  and observe the trajectories.

2. Figure 3.21 shows three setups in which one, two, and three hard disks are attached to the surface of a 2-D billiard table. In each case the disks scatter point particles elastically (no energy loss). The disks all have radius  $R$  and have center-to-center separations of  $a$ , with the three disks placed at the vertices of an equilateral triangle. Also shown in the figure are trajectories of particles scattered from the disks, with some of the internal scattering leading to trapped, periodic orbits in which the projectile bounces back and forth endlessly. For the two-disk case there is just a single trapped orbit, but for the three disk case there are infinitely many, which leads to chaotic scattering. In §6.8.3 we explore the quantum mechanical version of this problem. Note, the quantum version of this problem `3QMDisks.py` in Listing 6.18 has a potential subroutine that can be used to model the present disks.
3. Modify the program already developed for the study of scattering from the four-peaked Gaussian (3.52) so that it can be applied to scattering from one, two, or three disks.
4. Repeat the study of scattering from the four-peak Gaussian in §3.9 though now for the one, two, and three disk case.
  - a. Since infinite potential values would cause numerical problems, pick instead a very large value for the potential and verify that increasing the strength of the potential even further has no significant effect on the scattering.
  - b. Explore different geometries to find some that produce chaos. This should be near  $a/R \simeq 6$  for the three hard disks.

## 3.10 Billiards

Deriving its name from the once-popular parlor game, a mathematical *billiard* describes a dynamical system in which a particle moves freely in a straight line until it hits a boundary wall, at which point it undergoes specular reflection (equal incident and reflected angles), and then continues on in a straight line until the next collision. The

confining billiard table can be square, rectangular, circular, polygonal, or a combination of the preceding, and can be three dimensional or in curved spaces. These are Hamiltonian systems in which there is no loss of energy, in which the motion continues endlessly, and which often display chaos.

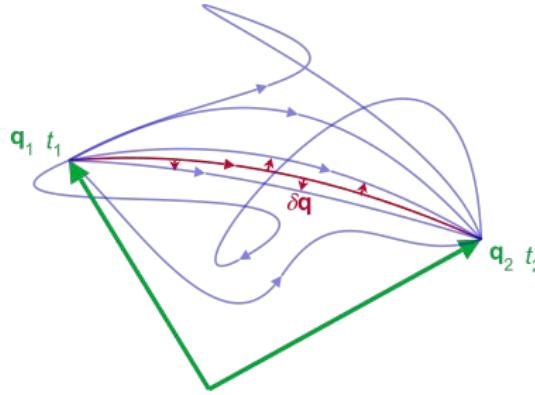
In Figure 3.19 we show square (a, c), circular (b, d), Sinai (e), and stadium billiards (f), with the arrows indicating possible trajectories. Note how right-angle collisions lead to two-point periodic orbits for both square and circular billiards, while in (c) and (d) we see how  $45^\circ$  collisions leads to four-point periodic orbits. Figures (e) and (d) show nonperiodic trajectories that are ergodic, that is, orbits that in time will pass with equal likelihood through all points in the allowed space. These latter orbits lead to chaotic behaviors.

1. The problems for this section are for you to write programs that compute the trajectories for any or all of these billiards and for a range of initial conditions. This is straight forward computationally (it's just geometry) and is an excellent way to study chaos. Preferably, your programs should produce animated output so you can view the trajectories as they occur. In Listing 3.3 we give a sample program for a square billiard that produces an animation using the Visual package (we do the quantum version of this problem in §6.8.3).
2. Have your program compute, keep track of, and plot the distance between successive collision points as a function of collision number. The plot should be simple for periodic motion, but show irregular behavior as the motion becomes chaotic. *Hint:* 20-30 collisions typically occur before chaos sets in.
3. Keep in mind that not all initial conditions lead to chaos, especially for circles, and so you may need to do some scanning of initial conditions.
4. Keep track of how many collisions occur before chaos sets in. You need at least this many collisions to test hypersensitivity to initial conditions.
5. For initial conditions that place you in the chaotic regime, explore the difference in behavior for a relatively slight ( $\leq 10^{-3}$ ) variation in initial conditions.
6. Try initial conditions that differ at the machine precision level to gauge just how sensitive are chaotic trajectories really are to initial conditions (be patient).
7. How much does the number of steps to reach chaos change for a  $10^{-3}$  variation?

## 3.11 Lagrangian and Hamiltonian Dynamics

### 3.11.1 Hamilton's Principle

As illustrated in Figure 3.22, Hamilton's principle states that the actual trajectory of a physical system described by a set of generalized coordinate  $\mathbf{q} = (q_1, q_2, \dots, q_N)$



**Figure 3.22.** A physical system evolves such that out of the many possibilities, the actual path taken through configuration space is that which produces a stationary action,  $\delta S = 0$  [Penrose(07)].)

between two specific states 1 and 2 is such that the action integral is stationary under variation in  $\mathbf{q}$ :

$$S[\mathbf{q}] = \int_{t_1}^{t_2} L(\mathbf{q}(t), \dot{\mathbf{q}}, t) dt, \quad \frac{\delta S}{\delta \mathbf{q}(t)} = 0, \quad (3.54)$$

where  $L$  is the Lagrangian of the system.

1. You are given the fact that a particle falls a known distance  $d$  in a known time  $t = \sqrt{2D/g}$ . Assume a quadratic dependence on distance and time,

$$d = \alpha t + \beta t^2. \quad (3.55)$$

Show that the action  $S = \int_0^{t_0} L dt$  for the particle's trajectory is an extremum only when  $\alpha = 0$  and  $\beta = g/2$ .

2. Consider a mass  $m$  attached to an harmonic oscillator with characteristic frequency  $\omega = 2\pi$  that undergoes one period  $T = 1$  of an oscillation:

$$x(t) = 10 \cos(\omega t). \quad (3.56)$$

- a. Propose a modification of the known analytic form that agrees with it at  $t = 0$  and  $t = T$ , though differs for intermediate values of  $t$ . Make sure that your proposed form has an adjustable parameter that does not change the  $t = 0$  or  $t = T$  values.
- b. Compute the action for an entire range of values for the parameter in your proposed trajectory and thereby verify that only the known analytic form yields a minimum action.

### 3.11.2 Lagrangian & Hamiltonian Problems

1. A bead of mass  $m$  moves without friction on a circular hoop of wire of radius  $R$ . The hoop's axis is vertical in a uniform gravitational field and the hoop is rotating at a constant angular velocity  $\omega$ .
  - a. Derive the Lagrange equations of motion for this system.
  - b. Determine the angle that the bead assumes at equilibrium, that is, an angle that does not change with time.
  - c. For small perturbations about this equilibrium configuration, what is the frequency  $\omega_0$  of oscillations?
  - d. Choose parameter values and solve Lagrange equations numerically.
  - e. Verify the analytic expression you have derived for the equilibrium angle and for the frequency of oscillations about the equilibrium position agrees with the numerical results.
  - f. Examine some initial conditions that lead to nonequilibrium positions of the bead.
  - g. Plot the time dependence of the bead's position and its trajectory in phase space for a wide range of initial conditions.

2. Consider a 1-D harmonic oscillator with displacement  $q$  and momentum  $p$ . The energy

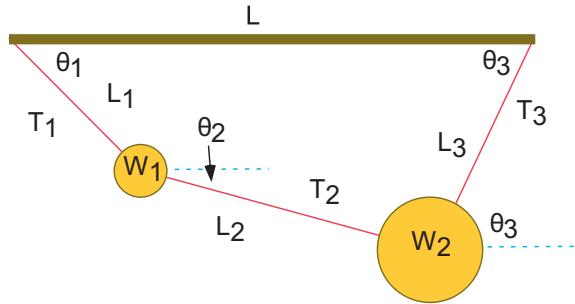
$$E(p, q) = \frac{p^2}{2m} + \frac{m\omega^2 q^2}{2} \quad (3.57)$$

is an integral of the motion, and the area of a periodic orbit is

$$A(E) = \oint p dq = 2 \int_{q_{min}}^{q_{max}} p dq. \quad (3.58)$$

- a. Use the analytic, or numeric, solution for simple harmonic motion to compute the area  $A(E)$ .
  - b. Use a central difference approximation to compute the derivative  $T = dA(E)/dE$ , and compare to the analytic answer.
  - c. Now repeat this problem using a nonlinear oscillator for which there is only a numerical solution. (Oscillators of the form  $V = kx^p$  with  $p$  even should work just fine.) You can determine the period from the time dependence of your solution, and then use your solution to compute  $A(E)$  for different initial conditions.
3. Verify **Liouville's Theorem** for the realistic (nonlinear) pendulum without friction. In particular, verify that the flow of trajectories in phase space is similar to that of an incompressible fluid.
    - a. The equations of motion for this problem are

$$\frac{dy^{(0)}}{dt} = y^{(1)}(t), \quad \frac{dy^{(1)}}{dt} = -\sin(y^{(0)}(t)). \quad (3.59)$$



**Figure 3.23.** A free body diagram for one weight in equilibrium. Balancing the forces in the  $x$  and  $y$  directions for all weights leads to the equations of static equilibrium.

You should already have worked out solutions to these equations in your study of nonharmonic motion.

- b. Start out with a set of initial configurations for the pendulum that fall upon a circle in the  $(y_1, y_2)$  phase space below the separatrix, and then follow the paths taken by different points higher up on the circle.
- c. Verify that different points on the starting circle and at the center of the circle take different times to orbit through phase space.
- d. Examine the orbits for at least 5 times:  $0, \tau/4, \tau/2, 3\tau/4, \tau$ , where  $\tau$  is the period for small oscillations.
- e. Verify that the domain of points that started on the original circle gets progressively more deformed as they proceed through phase space. Make sure also to include the point at the center of the circle.
- f. Verify that the density of points in phase space remains constant in time.
- g. Follow the passage through phase space for an initial circle of points that starts off with its center along the separatrix.
- h. Note the effect on the phase space orbits of including a frictional torque.

### 3.12 Weights Connected by Strings (Hard)

Two weights  $(W_1, W_2) = (10, 20)$  are hung from three pieces of string with lengths  $(L_1, L_2, L_3) = (3, 4, 4)$  and a horizontal bar of length  $L = 8$  (Figure 3.23). Determine the angles assumed by the strings and the tensions exerted by the strings. [An analytic solution exists, but it is painful, and in any case, the techniques used here are applicable to more complicated problems.]

1. Write out the equations for the geometric constraints that the horizontal length of the structure is  $L$  and that the left and right ends of the string are at the same height.

2. Write out the equations resulting from balancing the horizontal and vertical forces to zero for each mass.
3. Why can't we assume the equilibrium of torques?
4. You now have a set of simultaneous, nonlinear equations to solve. You may treat  $\theta$  as a single unknown or you may treat  $\sin \theta$  and  $\cos \theta$  as separate unknowns with an additional identity relating them. In any case, you will need to search for a solution.
5. We recommend setting the problem up as a matrix problem with all of the unknowns as a vector, and using a Newton-Raphson algorithm to search in a multi-dimensional space for a solution. For example, our solution [LPB(15)] was of the form

$$\vec{f} + F' \vec{\Delta x} = 0, \Rightarrow F' \vec{\Delta x} = -\vec{f}, \quad (3.60)$$

$$\vec{\Delta x} = \begin{pmatrix} \Delta x_1 \\ \Delta x_2 \\ \ddots \\ \Delta x_9 \end{pmatrix}, \quad \vec{f} = \begin{pmatrix} f_1 \\ f_2 \\ \ddots \\ f_9 \end{pmatrix}, \quad F' = \begin{pmatrix} \partial f_1 / \partial x_1 & \cdots & \partial f_1 / \partial x_9 \\ \partial f_2 / \partial x_1 & \cdots & \partial f_2 / \partial x_9 \\ \ddots & & \ddots \\ \partial f_9 / \partial x_1 & \cdots & \partial f_9 / \partial x_9 \end{pmatrix}. \quad (3.61)$$

$$\Rightarrow \vec{\Delta x} = -F'^{-1} \vec{f},$$

where the inverse must exist for a unique solution.

6. At some stage derivatives are needed, and it is easiest to use a forward-difference approximation for them:

$$\frac{\partial f_i}{\partial x_j} \simeq \frac{f_i(x_j + \delta x_j) - f_i(x_j)}{\delta x_j}, \quad (3.62)$$

where  $\delta_j$  is an arbitrary small value.

### 3.13 Code Listings

```
# ForcedOsc.py Driven Oscillator with Matplotlib
import numpy as np, matplotlib.pyplot as plt
from rk4Algor import rk4Algor

F=1; m=1; mu=0.001; omegaF=2; k=1 # Constants
omega0 = np.sqrt(k/m) # Natural frequency
y = np.zeros((2))
tt = []; yPlot=[]
# Empty list init

def f(t,y): # RHS force function
```

```

freturn = np.zeros((2))                      # Set up 2D array
freturn[0] = y[1]
freturn[1] = 0.1*np.cos(omegaF*t)/m-mu*y[1]/m-omega0**2*y[0]
return freturn

y[0] = 0.1      # initial conditions: x
y[1] = 0.3      # init cond speed
f(0,y)          # call function for t=0 with init cons.
dt = 0.01;    i = 0
for t in np.arange(0,100,dt):
    tt.append(t)
    y = rk4Algor(t,dt,2, y, f)    # call rk4
    yPlot.append(y[0])            # Change to y[1] for velocity
    i = i+1
plt.figure()
plt.plot(tt,yPlot)
plt.title('$\omega_f=2, k=1, m=1, \omega_0=1, \lambda = 0.001$')
plt.xlabel('t')
plt.ylabel('x')
plt.show()

```

**Listing 3.1.** **ForcedOscillate.py** solves the ODE for forced oscillator using rk4 and plotting with Matplotlib.

```

# ODEsympy.py: symbolic soltn of HO ODE using sympy
from sympy import *
f, g = symbols('f g', cls=Function)      # makes f a function
t, kap, w0 = symbols('t kap w0')
f(t)
f(t).diff(t)

# The ODE
diffeq = Eq(f(t).diff(t,t) + kap*(f(t).diff(t)) + (w0*w0)*f(t))
print("\n ODE to be solved:")
print(diffeq)
print("\n Solution of ODE:")
ff = dsolve(diffeq,f(t))    # Solves ODE
F = ff.subs(t,0)
print(ff)

```

**Listing 3.2.** **ODEsympy.py** uses the symbolic manipulation package Sympy to evaluate the constants in the harmonic oscillator solution (3.2).

```

# SqBillardCM.py: Animated classical billiards on square table
from vpython import *
dt = 0.01;   Xo = -90.;   Yo = -5.4;
v = vector(13.,13.1,0)

```

```

r0 = r= vector(Xo,Yo,0);  eps = 0.1;  Tmax = 500; tp = 0
scene = canvas(width=500, height=500, range=120,\n
                background=color.white, foreground=color.black)
table = curve(pos=[(-100,-100,0),(100,-100,0),(100,100,0),\n
                  (-100,100,0),(-100,-100,0)])
ball = sphere(pos=vector(Xo,Yo,0),color=color.red,\n
              radius=0.1,make_trail=True)

for t in arange(0,Tmax,dt):\n
    rate(5000)\n
    tp = tp + dt\n
    r = r0 + v*tp\n
    if(r.x>= 100 or r.x<=-100):\n
        # Right and left walls\n
        v = vector(-v.x,v.y,0)\n
        r0 = vector(r.x,r.y,0)\n
        tp = 0\n
    if(r.y>= 100 or r.y<=-100):\n
        # Top and bottom walls\n
        v = vector(v.x,-v.y,0)\n
        r0 = vector(r.x,r.y,0)\n
        tp = 0\n
    ball.pos=r

```

**Listing 3.3.** **SqBilliardCM.py** Computes trajectories within a square billiard, and produces animated output using the Visual package.

```

# ProjectileAir.py: Order dt^2 projectile trajectory + drag\n
from vpython import *\n#from visual.graph import *\nfrom numpy import *\n\nv0 = 22.; angle = 34.; g = 9.8; kf = 0.8; N = 5\nv0x = v0*cos(angle*pi/180.); v0y = v0*sin(angle*pi/180.)\nT = 2.*v0y/g; H = v0y*v0y/2./g; R = 2.*v0x*v0y/g\nprint('No Drag T = ', T, ', H = ', H, ', R = ', R)\n\ngraph1 = graph(title='Projectile with & without drag',\n               xtitle='x', ytitle='y', xmax=R, xmin=-R/20., ymax=8, ymin=-6.0)\nfunct = gcurve(color=color.red)\nfunct1 = gcurve(color=color.green)\n\ndef plotNumeric(k):\n    vx = v0*cos(angle*pi/180.)\n    vy = v0*sin(angle*pi/180.)\n    x = 0.0\n    y = 0.0\n    dt = vy/g/N/2.\n    print("\n      With Friction   ") \n    print("      x           y")\n    for t in arange(0,0.7*T,0.1):\n        #rate(30)\n        dt=0.1\n        vx = vx - k*vx*dt\n        vy = vy - g*dt - k*vy*dt\n        x = x + vx*dt\n        y = y + vy*dt\n        funct.plot(pos=(x,y))\n
```

```

print( " %13.10f  %13.10f  %(x,y) )"

def plotAnalytic():
    v0x = v0*cos(angle*pi/180.)
    v0y = v0*sin(angle*pi/180.)
    dt = 2.*v0y/g/N
    print("\n      No Friction   ")
    print("      x          y")
    for t in arange(0,N,0.1):
        #rate(30)
        #t = i*dt
        x = v0x*t
        y = v0y*t -g*t*t/2.
        funct1.plot(pos=(x,y))
        print(" %13.10f  %13.10f %(x ,y) )

plotNumeric(kf)
plotAnalytic()

```

**Listing 3.4.** `ProjectileAir.py` solves for a projectile's trajectory using a form of the velocity–Verlet algorithm accurate to second order in time.

```

# UranusNeptune.py: Orbits of Neptune & Uranus

from visual.graph import *
scene = display(width=600,height=600,
    title = 'White Neptune & Black Uranus', range=40)
sun = sphere(pos=(0,0,0), radius=2, color=color.yellow)
escenau = gdisplay(x=600,width=400,height=400,
    title='Perturbation of Uranus Angular Position')
graphu = gcurve(color=color.cyan)

escenan = gdisplay(x=800,y=400,width=400,height=400)
graphn = gcurve(color=color.white)
rfactor = 1.8e-9
G = 4*pi*pi      # in units T in years, R AU, Msun=1
mu = 4.366244e-5 # mass Uranus in solar masses
M = 1.0           # mass Sun
mn = 5.151389e-5 # Neptune mass in solar masses
du = 19.1914      # distance Uranus Sun in AU
dn = 30.0611      # distance Neptune sun in AU
Tur = 84.0110     # Uranus Orbital Period yr
Tnp = 164.7901    # Neptune Orbital Period yr
omeur = 2*pi/Tur # Uranus angular velocity (2pi/T)
omennp = 2*pi/Tnp # Neptune angular velocity
omreal = omeur
urvel = 2*pi*du/Tur      # Uranus orbital velocity UA/yr
npvel = 2*pi*dn/Tnp      # Neptune orbital velocity UA/yr
# 1 Uranus at lon 2 gr 16 min sep 1821
radur = (205.64)*pi/180. # to radians in 1690 -wrt x-axis
urx = du*cos(radur)      # init x- pos ur. in 1690
ury = du*sin(radur)      # init y-pos ur in 1690
urvelx = urvel*sin(radur)
urvely = -urvel*cos(radur)
# 1690 Neptune at long.
radnp = (288.38)*pi/180. # 1690 rad neptune wrt x-axis
uranus = sphere(pos=(urx,ury,0), radius=0.5,color=(.88,1,1),

```

```

    make_trail=True)
urpert = sphere(pos=(urx ,ury ,0) , radius=0.5,color=(.88,1,1),
    make_trail=True)
fnu = arrow(pos=uranus.pos,color=color.orange, axis=vector(0,4,0))
npx = dn*cos(radnp)      #init coord x neptune 1690
npy = dn*sin(radnp)      #                      y
npvelx = npvel*sin(radnp)
npvely = -npvel*cos(radnp)
neptune = sphere(pos=(npx,npy,0) , radius=0.4,color=color.cyan,
    make_trail=True)
fun = arrow(pos=neptune.pos,color=color.orange, axis=vector(0,-4,0))
nppert = sphere(pos=(npx,npy,0) , radius=0.4,color=color.white, ←
    make_trail=True)
velour = vector(urvelx ,urvely ,0)   #initial vector velocity Uranus
velnp = vector(npvelx,npvely ,0)     #initial vector velocity Neptune
dt = 0.5                                # time increment in terrestrial year
r = vector(urx ,ury ,0)                  # initial position Uranus wrt Sun
rnp = vector(npx,npy,0)                  # initial position Neptune wrt Sun
veltot = velour
veltotnp = velnlp
rtot = r
rtotnp = rnp

def ftotal(r,rnp,i):    # i==1 Uranus  i==2 Neptune
    Fus = -G*M*mu*r/(du**3) # Force sun over URANUS
    Fns = -G*M*mn*rnp/(dn**3) # Force Sun over NEPTUNE
    dnu = mag(rnp-r)          # distance Neptune-Uranus
    Fnu = -G*mu*mn*(rnp-r)/(dnu**3) # force N on U
    Fun = -Fnu                 # force uranus on Neptune
    Ftotur = Fus+Fnu           # total force on U (sun + N)
    Ftotnp = Fns+Fun           # On Neptune F sun +F urn
    if i==1: return Ftotur
    else:   return Ftotnp

def rkn(r,veltot,rnp,m,i):    # on Neptune
    k1v = ftotal(r,rnp,i)/m
    k1r = veltot
    k2v = ftotal(r,rnp+0.5*k1r*dt,i)/m
    k2r = veltot+0.5*k2v*dt
    k3v = ftotal(r,rnp+0.5*k2r*dt,i)/m
    k3r = veltot+0.5*k3v*dt
    k4v = ftotal(r,rnp+k3r*dt,i)/m
    k4r = veltot+k4v*dt
    veltot = veltot+(k1v+2*k2v+2*k3v+k4v)*dt/6.0
    rnp = rnp+(k1r+2*k2r+2*k3r+k4r)*dt/6.0
    return r , veltot

def rk(r,veltot,rnp,m,i):    # on Uranus
    k1v = ftotal(r,rnp,i)/m
    k1r = veltot
    k2v = ftotal(r+0.5*k1r*dt,rnp,i)/m
    k2r = veltot+0.5*k2v*dt
    k3v = ftotal(r+0.5*k2r*dt,rnp,i)/m
    k3r = veltot+0.5*k3v*dt
    k4v = ftotal(r+k3r*dt,rnp,i)/m
    k4r = veltot+k4v*dt
    veltot = veltot+(k1v+2*k2v+2*k3v+k4v)*dt/6.0
    r = r+(k1r+2*k2r+2*k3r+k4r)*dt/6.0
    return r , veltot

for i in arange(0,320):# estaba 1240
    rate(10)
    rnewu ,velnewu = rk(r,velour,rnp,mu,1) # uranus
    rnewn ,velnewn = rkn(rnp,velnp,r,mn,2) # neptune

```

```

r = rnewu                      # uranus position
velour = velnewu                # uranus velocity
du = mag(r)
omeur = mag(velour)/du          # new abgykar velocity of uranus
degr = 205.64*pi/180 - omeur*i*dt # angular position uranus
rnp = rnewn                      # neptune pos
velnp = velnewn                 # neptune pos
dn = mag(rnp)
omenp = mag(velnp)/dn           # radians neptune
radnp = radnp-dt*omenp
npx = dn*cos(radnp)
npy = dn*sin(radnp)
rnp = vector(npx,npy,0)         # neptune position
deltaomgs = -omeur+omreal
graphu.plot(pos=(i,deltaomgs*180/pi*3600))
urpert.pos = r
fnu.pos = urpert.pos            # position of arrow on uranus
dnu = mag(rnp-r)                # distance Neptune-Uranus
fnu.axis = 75*norm(rnp-r)/dnu   # axes the arrow over uranus
neptune.pos = rnp               # radiovector Neptune
fun.pos = neptune.pos
fun.axis = -fnu.axis             # arrow on neptune

```

**Listing 3.5.** `UranusNeptune.py` solves for the orbits of Uranus and Neptune and their interaction.