

Concepts of this Module

- Periodic and simple harmonic motion.
- Simple harmonic motion as one component of uniform circular motion.
- Pendulums when the small angle approximation is not valid.
- Double pendulums, spring-mass systems with an oscillating support point, and chaotic systems.

The Activities



**Course
Concepts**

Activity 1

Here is the URL of a Flash animation of two systems executing harmonic motion.

<http://www.upscale.utoronto.ca/PVB/Harrison/Flash/ClassMechanics/SHM/TwoSHM.html>

The above link is to a fixed size animation which works nicely if only one person is viewing it. For use in the Practical itself a version which can be resized to be larger so that the entire Team can see it is better. Here is a link to such a version:

<http://www.upscale.utoronto.ca/PVB/Harrison/Flash/ClassMechanics/SHM/TwoSHM.swf>

Open the animation. In your own words describe which characteristics of these two systems are the same. In your own words describe what is different for these two systems.



**Course
Concepts**

Activity 2

Here is the URL of a Flash animation comparing uniform circular motion to simple harmonic motion:

<http://faraday.physics.utoronto.ca/PVB/Harrison/Flash/ClassMechanics/Circular2SHM/Circular2SHM.html>

The above link is to a fixed size animation which works nicely if only one person is viewing it. For use in the Practical itself a version which can be resized to be larger so that the entire Team can see it is better. Here is a link to such a version:

<http://faraday.physics.utoronto.ca/PVB/Harrison/Flash/ClassMechanics/Circular2SHM/Circular2SHM.swf>

Open and run the animation.

- A. For an object in uniform circular motion, the angle θ as a function of time changes according to:

$$\theta = \omega t$$

where ω is called the angular **velocity**. What are the units of ω ?

- B. For an object in uniform circular motion, the y coordinate of a point on the object changes as a function of time according to:

$$y = r \sin(\omega t)$$

What is the meaning of r ? What are its units?

- C. An object executing *Simple Harmonic Motion* is described by:

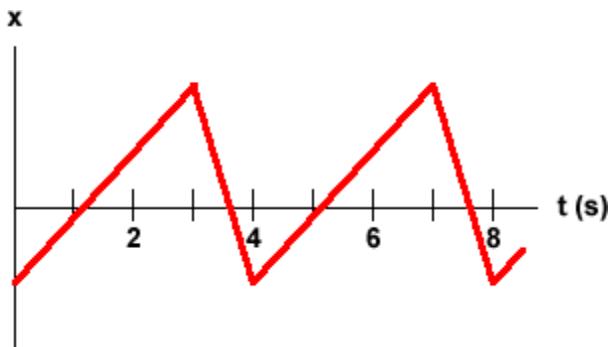
$$y = r \sin(\omega t)$$

Note that this is the same equation as in Part B. However in this case ω is called the angular **frequency**. In this case what are the units of ω ? In your own words explain why two somewhat different names are used for the same symbol ω . For Simple Harmonic Motion what name is usually used for r ? What are the units of r ?

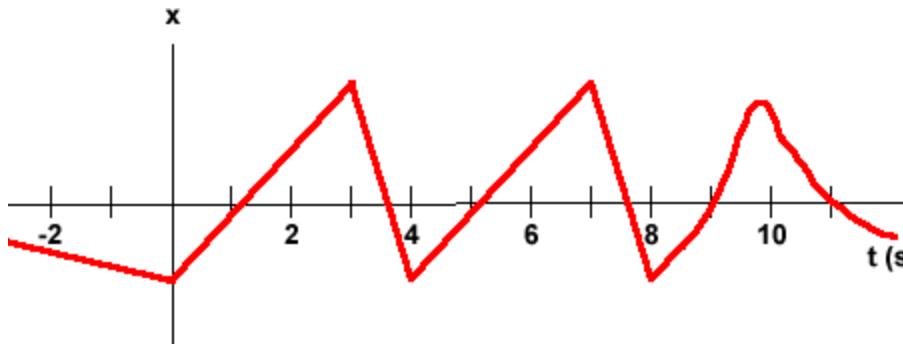


Activity 3

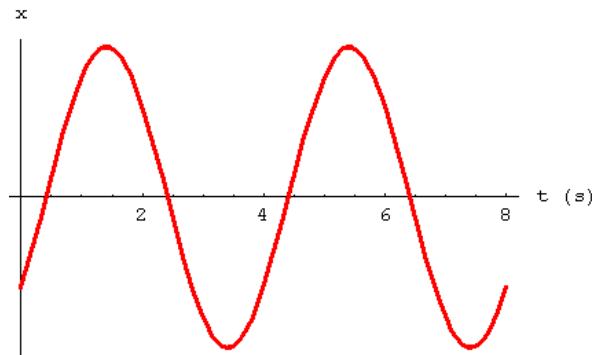
- A. Here is a plot of the position versus time for a particle. Does the motion appear to be *periodic*? Does the motion appear to be *Simple Harmonic*? What is the period of the motion? What is the frequency of the motion?



- B. Here is another position versus time plot. For values of the time between 0 and 8 s it is identical to Part A. Does this motion appear to be periodic? For the particle of Part A, if you only have data for times between 0 and 8 s can you tell the difference between that motion and the motion shown here?



- C. Here is another plot of the position versus time for a particle. Does the motion appear to be *periodic*? Does the motion appear to be *Simple Harmonic*? What is the period of the motion? What is the frequency of the motion? What is the angular frequency of the motion?

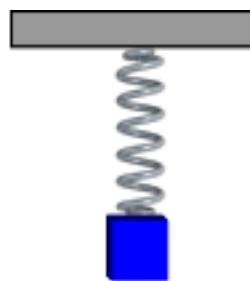


- D. Does it make sense to talk about the angular frequency of the motion of Part A? Explain.



Expt Activity 4

Mount the supplied spring on the support and hang the supplied mass from it. Gently lift the mass a small amount and release it, so that it bobs up and down. The oscillations should be small enough so that the motion is smooth and the spring is always vertical and stretched. Position the Motion Sensor under the mass and pointing up at it, as shown. You will connect the Motion Sensor to the U of T DAQ Device. You may have used this same setup in Mechanics Module 5 Activity 8.



Be aware that the Motion Sensor has a switch which adjusts the width of the beam; one of the switch settings will probably result in better data than the other. Recall that the Motion



Sensor can only measure distances greater than 0.15 m. This means that for the given coordinate system, if the Motion Sensor is at $x = 0$ the bottom of the mass must always have a value of $x > 0.15\text{m}$.

- Set the vertical position of the mass-spring position so that when the mass is oscillating the minimum distance from the Motion Sensor is close to but greater than 0.15 m. Try to have the mass moving only up and down.
- Set the Motion Sensor for the wide beam. On some units this is indicated by an icon of a person.
- After starting the *MotionSensor.vi* software, set the sample rate to about 60 samples per second. You may wish to try a different sampling rate to see if it improves the quality of your data.
- Collect data for a few oscillations.

Now the Activities:

- A. Does the motion appear to be Simple Harmonic? What is the period, frequency, angular frequency, and amplitude of the motion?
- B. What physical characteristics must the spring-mass system have for the motion to be truly Simple Harmonic and not just approximately so? Do you think this spring-mass system has those characteristics?



Expt Activity 5

In Activity 4 a Motion Sensor is used to measure the position versus time for a mass oscillating on a spring. A similar Activity is Mechanics Module 5 Activity 8. There were about 60 samples every second and, depending on the particular spring and mass used the period of the oscillation was about 0.5 s.

Imagine we have an ideal spring-mass system oscillating with maximum amplitude = 1 m and angular frequency $\omega = 6 \text{ s}^{-1}$. The motion is:

$$y = A \sin(\omega t) = \sin(6t)$$

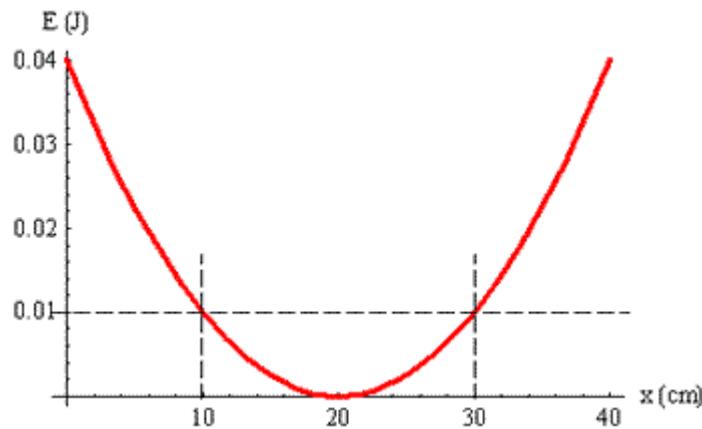
- A. What is the period of the oscillation?
- B. Imagine we use a Motion Sensor to measure the position once per second and take the first measurement at $t = 0$ s. Remember that in the above formula the argument to the sine function is in radians. What will be the measured values of y for the first few measurements? Is your results reasonable? Explain. You may find it helpful to draw a rough sketch of y versus t for a few periods of the oscillation and locate on the graph the points where you would have measured the values of y .



Activity 6

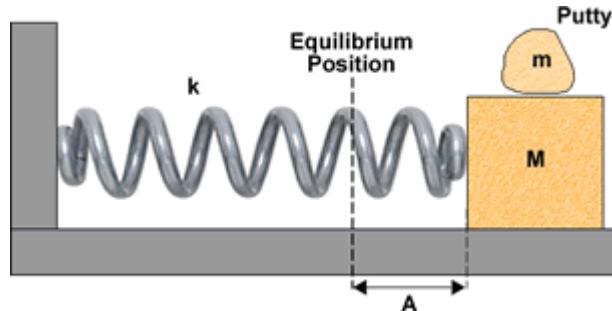
The figure shows the potential energy of a particle oscillating on a spring. Note that the horizontal axis has units of cm.

- What is the spring's equilibrium length? What is the value of the spring constant k ?
- The turning points of the particle are at 10 cm and 30 cm. What is the maximum amplitude of the particle's motion? What is the particle's maximum kinetic energy?
- Sketch a graph of the particle's kinetic energy as a function of x . What is the shape of the sketch?
- The particle has a mass of 2.0 kg. What is the particle's maximum speed? Sketch a graph of the particle's speed as a function of x . What is the shape of the sketch?
- If the total energy of the particle is tripled, what will be the value of the maximum speed of the particle?
- If the total energy of the particle of Parts A - D is tripled, what will be its maximum amplitude?



Activity 7

A block of mass M is attached to a spring of spring constant k , and oscillates back and forth with amplitude A on a frictionless surface. At the moment shown the block is at its maximum amplitude and a lump of putty of mass m is dropped from a very small height and sticks to the block. The mass of the spring is negligible.



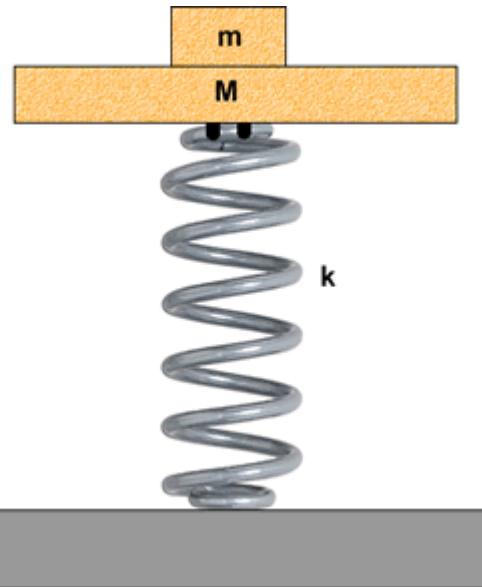
What are the new amplitude and frequency of the block plus putty system?



Activity 8

A vertical massless spring with spring constant k has a platform of mass M fixed to it. A mass m is sitting on top of the platform, but is not fixed to it. The two masses are oscillating vertically with amplitude A . There is a vertical position, the *equilibrium point*, where the total force acting on the two masses as a single system, their weight plus the force exerted by the spring, is zero.

- A. Imagine that the oscillation is so extreme that the mass m just loses contact with the platform. Where is this going to occur and why?
- B. What is the value of A when for which the mass m just loses contact with the platform?



So far, except for Activities 3A and 3B, all the motions we have investigated were at least approximately Simple Harmonic. In Activities 9 and 10 we will explore two pendulums for which the deviations from Simple Harmonic are not negligible. For both we concentrate on the how the period of the pendulum varies with the maximum amplitude.

Newton's Laws for a simple pendulum of length L gives us a differential equation:

$$\frac{d^2\theta}{dt^2} + \frac{g}{L} \sin \theta = 0 \quad (1)$$

In classes and the textbook, we almost always restrict ourselves to the case where the angle θ is always small, so that we can approximate that:

$$\sin \theta \approx \theta \quad (2)$$

where θ is in radians. Then we can approximate Eqn (1) as:

$$\frac{d^2\theta}{dt^2} + \frac{g}{L} \theta = 0 \quad (3)$$

Except for the differences in symbols, this is the same equation as for a spring mass system:

$$\frac{d^2x}{dt^2} + \frac{k}{m}x = 0 \quad (4)$$

You may have studied how to do an approximation of this system in the *Numerical Approximation Module*.

The solution to Eqn (3) is simple harmonic motion with a maximum amplitude θ_{\max} and period:

$$T_0 = 2\pi\sqrt{\frac{L}{g}} \quad (5)$$

The reason why we usually approximate Eqn (1) with Eqn (3) is that an exact analytical solution of Eqn (1) does not exist!

With considerable effort and sophisticated mathematics, from Eqn (1) one can determine that the period of the pendulum with a maximum amplitude of θ_{\max} is given by:

$$T = \frac{\sqrt{2}}{\pi} T_0 \int_0^{\theta_{\max}} \frac{d\theta}{\sqrt{\cos\theta - \cos\theta_{\max}}} \quad (6)$$

This is not terribly helpful, since the integral involves *elliptical integrals* which are not analytically solvable either, so one must either numerically approximate to find the value of T or look up the integral in some table. These sorts of integrals are so common that considerable effort has been expended to develop computer algorithms that are fast and accurate. Table 1 gives some values of T/T_0 .¹

Table 1

θ_{\max} (rads)	T/T_0
0.01000	1.00001
0.10000	1.00063
$\pi/4$	1.03997
$\pi/2$	1.18034
2.000000	1.32890
3.000000	2.57123
π	∞

There are a few ways to avoid becoming mired in all the mathematics:

¹ These values were calculated with *Mathematica*.

1. Use numerical approximation.
2. Build and measure a real pendulum.
3. Do even more mathematics to simplify Eqn (6) into an approximate form that can be solved with a simple calculator.

In Activity 9 we will use numerical approximation, and Activity 10 we will use a real pendulum. If you wish to see the sort of gyrations necessary to get a formula that can be solved with a calculator, see for example F.M.S. Lima and P. Arun, “An accurate formula for the period of a simple pendulum oscillating beyond the small angle regime”, American Journal of Physics **74**(10), 892 – 895 (2006), <http://scitation.aip.org/getabs/servlet/GetabsServlet?prog=normal&id=AJPIAS00007400010000892000001&idtype=cvips&gifs=yes>.



Expt Activity 9

Here are some values of the period of a simple pendulum for a length $L = 1.0000$ m taking $g = 9.8000$ m/s² using Eqn. 5 and the values of Table 1.

Table 2

θ_{\max} (rads)	T (s)
0.01000	2.00710
0.10000	2.00835
$\pi/4$	2.08732
$\pi/2$	2.36905
2.000000	2.66723
3.000000	5.16070
π	∞

- A. A listing of the code for `Pendulum.py` appears in Appendix 1. We assume that the mass is connected to the support by a massless rigid rod. The algorithm used to approximate this system is very similar to the one used for the spring-mass system investigated in the *Numerical Approximation Module*. Examine the code and describe in your own words how the algorithm approximates the motion of the pendulum.
- B. The code also estimates the period of the oscillation. Describe in your own words how the estimation is done.
- C. The “master” copy of `Pendulum.py` is located at:

Feynman:Public/Modules/Oscillations

Copy the file to your Team’s area on the server. Start the IDLE for VPython

- program and open the copy of `Pendulum.py`. Use Run / Run Module or press the F5 key on your keyboard to start the animation. How does the estimated value of the period compare to the value in Table 1? Can you think of a better way to determine the period of the simulation?
- D. Modify the code so that when the animation starts it prints the maximum amplitude and the timestep of the approximation. Test your changes to make sure that they work. You will wish to know that when you change the code and run the animation *VPython* will over-write the file with the new version.

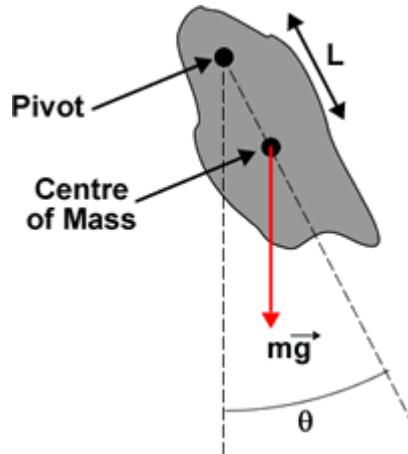
A tip: when you start the animation, two windows are opened, a *Python Shell* which shows the results of all `print` statements in the code and a window of the animation. To stop the animation close the animation window, but leave the *Python Shell* window open. This will be useful for Part E.

- E. Does the numerical approximation agree with the values of Table 2? How does the approximation do for different values of the timestep? You may wish to print the window of the data for the estimated periods for various maximum amplitudes and timesteps and staple it into your lab book.



Expt Activity 10

For a real pendulum, things are not quite so simple. In the figure we show a real pendulum. The distance between the pivot point and the centre of mass is L , the moment of inertia of the object about the pivot is I , and the total mass of the object is m . Often this is called a *physical pendulum*.



Using the usual convention that positive torques cause counter-clockwise rotations, the torque exerted on the pendulum is:

$$\tau = -mgL\sin\theta \quad (7)$$

Newton's 2nd Law for rotational motion is:

$$\alpha \equiv \frac{d^2\theta}{dt^2} = \frac{1}{I}\tau \quad (8)$$

Thus, the equation of motion is:

$$\frac{d^2\theta}{dt^2} + \frac{mgL}{I} \sin \theta = 0 \quad (9)$$

Note that except for differences in the symbols this has the same mathematical structure as Eqn (1) for a simple pendulum. Therefore within symbol changes the solution is also the same. In particular, in the *small angle approximation* the motion is *simple harmonic* and the period of oscillation is:

$$T_0 = 2\pi \sqrt{\frac{I}{mgL}} \quad (10)$$

Similarly, for oscillations where the maximum amplitude is not small Eqn (6) is also true:

$$T = \frac{\sqrt{2}}{\pi} T_0 \int_0^{\theta_{\max}} \frac{d\theta}{\sqrt{\cos \theta - \cos \theta_{\max}}} \quad (11)$$

Therefore the ratios of T/T_0 from Table 1 are also true for this case.

Here you will side-step trying to solve Eqn (11) by taking data on how the period of a real physical pendulum varies with its maximum amplitude. You will use a Pasco Rotary Motion Sensor, the U of T Data Acquisition Device (DAQ), and the *RMS* program. The *RMS* program is based on the *SignalExpress* software platform from National Instruments.

The Rotary Motion Sensor is mounted on a support, and has a rod with a mass on it mounted on it. The rod and mass will be the physical pendulum you will study. Also part of the rotating system are the plastic disc on which the rod is mounted and the axel connected to the Rotary Motion Sensor.

Setup

There are two phone plugs on cables connected to the Rotary Motion Sensor. These are plugged into the corresponding terminals on the Data Acquisition Device. On the left side of the Device are two pairs of terminals labeled *Digital Channels*; each pair has one labeled with a yellow circle and the other with a black circle. The corresponding yellow and black plugs from the sensor are plugged into pair labeled **0**, yellow to yellow and black to black.

When you start *RMS* the position of the Rotary Motion Sensor defines the angle to be zero. Make sure that the pendulum is stationary and start *RMS*.

Occasionally the hardware and software gets confused about the resolution of the measurement.

1. Rotate the pendulum by one full rotation. The angle on the graph should read 6.28 radians ($= 360^\circ$) either a positive or negative number.
2. If the angle is not correct, disconnect the Rotary Motion Sensor from the DAQ then plug it back in.
3. Stop and restart the *RMS* program and check that one full rotation reads 6.28 radians.
4. Stop *RMS*.

Trial Run

It is a good idea to do a trial run first. You should do a run with a maximum amplitude $\theta_{\max} \approx 1$ radian .

1. Make sure the pendulum is stationary.
2. Start *RMS*.
3. Rotate the pendulum to some about 1 radian. You can read the angle on the graph.
4. Release the pendulum. The graph will show the angle of the pendulum as a function of time.
5. After about 10 oscillations stop *RMS*.
6. Right click on the graph. Choose `Visible Items / Cursors`.
7. Drag the solid cursor to one of the early oscillations. The horizontal line of the cursor will track the data.
8. Drag the dotted cursor to one of the later oscillations.
9. At the bottom of the graph the `Dx` field is the time between the two cursors. `y1` and `y2` are the angles of the first and second cursors respectively, in radians, and `Dy` is the change in the angle.

Thus you may calculate the value of the period and the maximum amplitude.



Note: due to a mis-feature of *SignalExpress*, sometimes when you take another data set it is difficult to use the cursors. To fix, turn the cursors off and then back on using `Visible Items / Cursors`.

There are some issues that should be considered:

- A. The time resolution of the software is 0.05 s. You can confirm this by moving one of the cursors the minimum possible amount and seeing how much the value of the time changes. Considering that the measured value of the time when the pendulum is at maximum amplitude is unlikely to be the exact value when it actually occurred, a reasonable value for the error in the measurement of the time Δt is perhaps ± 0.03 s. Is this a reasonable estimate? If yes, explain why in your own words. If no, what is a more reasonable value?

- B. If you position the cursors at successive maximum values of the amplitude, then the time between them is the period T of oscillation. What is a reasonable estimate of the uncertainty in the period, ΔT ?
- C. Say you position the cursors for n oscillations. Call the final value of the time t_f and the initial value t_i . The software shows the value of $t_f - t_i$ as Dx at the bottom of the window. The value of period is $T = Dx/n$. Now what is the uncertainty in the period, ΔT ? Is this a better way to determine the period?
- D. Since this is not a perfect frictionless system, the maximum amplitude decreases with time. You can see this in your data. Thus the value of θ_{\max} is not well defined. What is a reasonable estimate of the error in its value? Why? How does the value depend on the number of oscillations between the cursors?
- E. Considering the results of Part C and Part D, how many oscillations should you use to determine the values of the period for a given maximum amplitude? Note that there is a trade-off: increasing the number of oscillations n reduces the error in the period but increases the error in the maximum amplitude. Explain your choice.

Data Collection and Analysis

Collect data for the period for a number of values of the maximum amplitude.

Use *CreateDataSet* to create a dataset of values for the period and maximum amplitude, including their errors, and save it into your Team's area. You will want the values of the maximum amplitude to be the *Independent (x) Variable*, and the values of the period be the *Dependent (y) Variable*.

You can then use *ViewDataSet* to view your data. You may wish to print the window and staple it into your lab book.

Does the data look reasonable? Does it appear to be consistent with the values of Table 1?

Parameterising the Data

You have determined the period of the pendulum for measured values of the maximum amplitude. Here we explore how to use that data to determine the value of the period for maximum amplitudes that have not been measured.

The relation between the period and the maximum amplitude is a complicated function $f()$ involving elliptical integrals:

$$T = f(\theta_{\max}) = \frac{2}{\pi} T_0 \int_0^{\theta_{\max}} \frac{d\theta}{\sqrt{\cos \theta - \cos \theta_{\max}}} \quad (12)$$

However, it might be reasonable to approximate the function as a polynomial:

$$T = a(0) + a(1)\theta_{\max} + a(2)\theta_{\max}^2 + a(3)\theta_{\max}^3 + a(4)\theta_{\max}^4 + \dots \quad (13)$$

- A. Can you eliminate the odd coefficients $a(1)$, $a(3)$, etc. from the series using a physical argument? If yes, what argument can you use?
 - B. Use the *PolynomialFit* program to find the best fit the data. Do not include any terms that are physically unreasonable. You will find that adding some terms to the polynomial will not improve the quality of the fit: your goal is to find the minimum number of terms in the polynomial that provides a good fit to the data. Any fitted parameters $a(i)$ that are zero within errors should not be included in the fit.
 - C. From the best fit, what is the value of T_0 ?
 - D. How well does the best fit do in duplicating the values of Table 1? Does it do a better job for small angles than for large ones?
-

Activities 9 and 10 were concerned with pendulums whose motion was approximately but not exactly *Simple Harmonic*. In Activities 11 and 12 we will investigate some periodic systems whose motion is not even close to being Simple Harmonic. Both Activities are only brief looks at the two systems, and are just for your interest.

Both systems that are investigated are *chaotic*. Here are some characteristics that all chaotic systems have:

1. No analytic formula can even approximate the motion.
2. The motion will never repeat. Ever!
3. If two identical periodic systems are started with almost identical positions and speeds, soon their motions will be radically different from each other.
4. Chaotic systems are deterministic. If they start with exactly identical initial conditions their subsequent behavior will be exactly the same.

Characteristic 3 is sometimes called the *butterfly effect*. This is because if history were a chaotic system then the outcome of World War II could have been determined by whether or not a butterfly landed on a particular flower in the Himalaya Mountains in 1848. You may learn more about chaotic systems from:

<http://www.upscale.utoronto.ca/PVB/Harrison/Chaos/Chaos.html>



Expt Activity 11

Our first example of a non-harmonic pendulum is the *double pendulum*. It is an example of a chaotic system.

Start the IDLE for VPython program. Use File / Open ... to open the doublependulum.py program in the examples directory; this directory is the default one that is opened when IDLE is first started. Run the program. You are welcome to look at the code for this program, but will wish to know that it is written in terms of a sophisticated form of Newton's Laws called a Lagrangian formulation.

You may investigate the double pendulum further with a Java applet by Peter Selinger at Dalhousie University, Halifax Nova Scotia:

<http://www.mscs.dal.ca/~selinger/lagrange/doublependulum.html>

From the trace of the trajectory you can see characteristic #2 above illustrated. By clicking on the *Restart* button you can see characteristic #4 demonstrated.



Expt Activity 12

In Activity 4 you may have used a Motion Sensor to track the position of a mass oscillating up and down on a spring. If the support of the spring is oscillating up and down, for some frequencies and amplitudes of oscillation this system too is chaotic.

FEED ME.

This Student Guide was written by David M. Harrison, Dept. of Physics, Univ. of Toronto in May 2008

Revision by David M. Harrison and Jason Harlow. Last revision: November 5, 2010.

Activity 3.A is based on Activity 14.1 Part 3 of Randall D. Knight, **Student Workbook with Modern Physics** (Pearson Addison-Wesley, 2008). Activity 6 is based on Activities 1.43 Parts 10, 11, and 12 of Knight's **Student Workbook**. Activities 7 are 8 are from David Harrison and William Ellis, **Student Activity Workbook**, 3rd ed. (Norton, 2008), Activities 15.8, 15.9.

Appendix 1 – Listing of Pendulum.py

```

# Solve the pendulum using numerical approximation
# Copyright (c) 2008 David M. Harrison

# The next line is an internal revision control id:
# $Date: 2008/05/10 10:33:45 $, $Revision: 1.1 $

# Import the visual library
from visual import *

# The initial angle in radians.
theta = pi/2.0

# The initial angular velocity
omega = 0

# Set g and the length of the pendulum
g = 9.80
L = 1.00

# These four lines control the size of the window of
# the animation and the scale. The details of these lines
# are not important for our purposes.
scene.autoscale = 0
scene.height = 600
scene.width = 600
scene.range = vector(2.0,2.0,2.0)

#
# Now we build the pendulum which we will animate.
#

# The support for the pendulum
support = cylinder( pos = (0, 0, -0.5), axis = (0,0,1), radius = 0.02)

# The "frame" construct groups two or more objects into a single one.
# Here we group the cylinder and the sphere into a single object
# which is the pendulum.
pendulum = frame()
cylinder(frame=pendulum, pos=(0,0,0), radius=0.01, length=1,
color=color.cyan)
sphere(frame=pendulum, pos=(1,0,0), radius=0.1, color=color.red)

# Position the pendulum.
pendulum.pos = (0,0,0)

# Rotate the pendulum about the z axis. Note that VPython measures
# angles with respect to the x (horizontal) axis. We are measuring
# angles with respect to the vertical (-y axis) so we subtract
# pi/2.0 radians from the angle.
pendulum.rotate(axis = (0,0,1), angle = theta - pi/2.0)

# The time
t = 0.

```

```

# Below we will want to store the old value of the time.
# Set it the "impossible" value of -1 initially.
t_old = -1.

# The time step
dt = 0.0005

# The value "1" is equivalent to true. So this causes the while
# loop to run forever.
while 1:

    # Set the rate of the animation in frames per second
    rate(1/dt)

    # The angular acceleration, i.e. the second derivative of the
    # angle with respect to time.
    alpha = -(g/L) * sin(theta)

    # The new value of the angular velocity
    omega = omega + alpha * dt

    # The change in the angle of the pendulum
    d_theta = omega * dt

    # A rough and ready way to estimate the period of the oscillation.
    # If the angle is positive and adding d_theta to it will make
    # it negative, then it is going through the vertical
    # from right to left.
    if(theta > 0 and theta + d_theta < 0) :
        # If t_old is > 0, then this is not the first cycle of
        # the oscillation. The difference between t and t_old
        # is the period within the resolution of the time step dt
        # and rounding errors. Print the period.
        if(t_old > 0):
            print "Estimated Period =", t - t_old, "s"
        # Store the current value of the time in t_old
        t_old = t

    # Rotate the pendulum about the z axis by the change in the angle
    pendulum.rotate(axis = (0,0,1), angle = d_theta)

    # Update the value of the angle
    theta = theta + d_theta

    # Update the time
    t = t + dt

```