

Pendulum Module Student Guide

Introduction

Not all oscillating motion is Simple Harmonic. The motion of a pendulum that swings through large angles is an example of non-simple harmonic motion. This week we will explore 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 Eq.[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:

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

The solution to Eq.[3] is simple harmonic motion with a small maximum amplitude $\theta_{max} \lesssim 10^\circ$ and period:

$$T_0 = 2\pi \sqrt{\frac{L}{g}} \quad [5]$$

The reason why we usually approximate Eq.[1] with Eq.[3] is that an exact analytical solution of Eq.[1] does not exist! If we want to analyze a pendulum with large oscillations, there are three ways:

1. Build and measure a real pendulum.
2. Write a numerical approximation in Python.
3. Do lots more math.

In Activity 1 you will build and measure a real pendulum. In Activity 2 you will modify and use a Python code to simulate your pendulum.

Since this is an experimental physics Practical, we won't pursue the third option of doing lots more math today. But, if you're interested, it does turn out that from Eq.[1] one can determine that the period of the pendulum with a maximum amplitude of θ_{\max} is given by:

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

Unfortunately, this is an elliptic integral, which cannot be expressed in terms of elementary functions. So, to get an actual period, you will have to numerically compute it with a python code, for example. If you wish to see the sort of mathematics necessary to get a formula that can be solved *without* a computer code, 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://fy.chalmers.se/~f7xiz/TIF080/Period.pdf> .

Activity 1: Period of a Real Pendulum at Home (35 minutes)

Construct a pendulum in your house. You will need some light string, such as dental floss. Tie one end of the string to a small heavy object. I used a little Hotwheels car (shown) but you might try a small bottle of nail polish, an ornament, a piece of clay, or a piece of jewelry. It should be much more massive than the string, and no bigger than 3 or 4 cm across, so that the effect of air resistance is reduced. Tie the other end of the string to some fixed, elevated point under which you can make the object swing in a wide arc back and forth. The string should be between 50 and 110 cm long.



Question A: Experiment Setup

Take a photo of your object when it is hanging straight down, including the pivot point at the top. Measure the length L , between the fixed point where you tied the string to the centre of mass of the swinging object. Use Eq.[5] to compute T_0 , the predicted period for Small Angle oscillations.

Question B: Small Angle Video Tracker

Set your pendulum swinging back and forth in Small Angle oscillations (θ_{\max} between 5 and 10°). Take a short video of your pendulum swinging back and forth. The video should be taken from the side so that the pendulum remains the same distance from your camera as it swings from side to side. The pivot point at the top where the string is tied to a fixed point must be included in the video. Take the video for 2 or 3 complete circles. Load the video into Tracker (<https://physlets.org/tracker/>). In order to calibrate the distance scale, add a calibration stick which is the length of the string, L , that you measured before. Move the Coordinate Axes to coincide with the pivot point at the

top. Click Create and add a Point Mass. Frame by frame, press shift while you click the mouse button to add position-time measurements for the pendulum as it travels in its vertical circular path. Include tracking points for at least one complete back and forth oscillation. Include a screenshot of your tracking points and the graph that Tracker creates of $x(m)$ versus $t(s)$.

Question C: Small Angle Tracker Analysis

Go to File→Export, and Export a Data File. Do this for the Point Mass data you collected, with Comma Delimiter. You can name it something like filename.csv, where the csv extension stands for “comma separated values”. Open the data with a spreadsheet like Excel or Google Sheets. You should notice that the x values oscillate back and forth around zero, and that y is mostly negative, and is most negative when $x = 0$. Compute θ in a new column, based on x and y [Note that if you are using Excel or Google Sheets this could be done with an inverse trig function, such as =ATAN()]. Make a scatter plot of theta versus time. Estimate the amplitude, θ_{\max} , and the period, T . Compare with theoretical prediction T_0 for Small Angle oscillations.

Question D: Large Angle Video Tracker

Repeat Step B, but for a Large Angle oscillation. Try to make it swing with θ_{\max} as close to 90° as you can easily get it (ie larger than 60° , if you can).

Question E: Large Angle Tracker Analysis

Repeat Step C for your Large Angle oscillation data. Has the period changed significantly?

Activity 2: Python Simulation of a Pendulum (30 minutes)

Find the code pendulum.py on the Quercus Practicals page, and save it to your computer. If you haven't used Python in a while, it might be useful to go review the Python intro document at <https://computation.physics.utoronto.ca/getting-started-python-3-computational-physics-u-t/> .

Question A: Running code with default values

Start the Spyder python-editing program, which should be on your desktop, and open the copy of pendulum.py. Use Run / Run Module or press the F5 key on your keyboard to start the simulation. Record the default values of L , θ_{\max} , and the output period. Also make a note of the time-step, dt , which sets the precision of the period output.

Question B: Simulation of Activity 1 Small Angle experiment

Modify the code so that it simulates the results of your Small Angle data you took in Activity 1, parts B and C. Make a scatter plot based of the theta vs time data in pendulum_Output.csv (which is a “comma separated values” file). Find the period. How do the results of your numerical approximation compare to the data you collected?

Question C: Simulation of Activity 1 Large Angle experiment

Modify the code so that it simulates the results of the Large Angle data you took in Activity 1, parts D and E. Make a scatter plot based of the theta vs time data in pendulum_Output.csv , and find the period. How do the results of your numerical approximation compare to the data you collected?

Question D: Simulation of $\theta_{\max} > 90^\circ$

Use the python code to simulate a pendulum for which $\theta_{\max} > 90^\circ$. This is possible if, instead of a string, the mass is connected to the frictionless pivot by a light, thin, inflexible rod. Try a value of θ_{\max} that is quite high, but still less than 180° , and show a plot of theta versus time. Does this look like Simple Harmonic Motion. What happens if $\theta_{\max} = 180^\circ$?