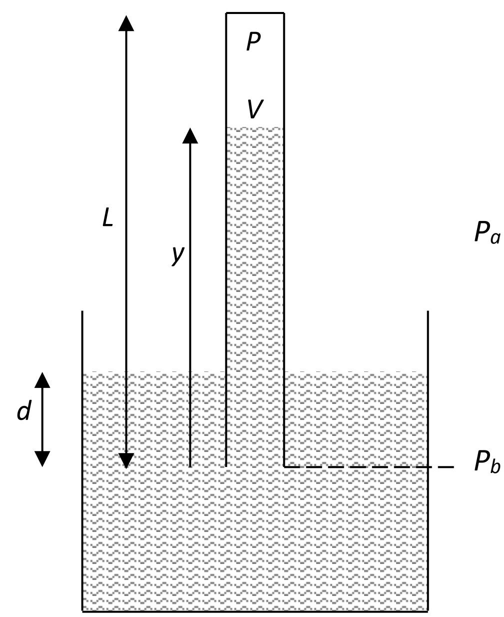
Water-Column Oscillations



Consider a water column beneath trapped air, as shown in the figure. The tube containing the water column is open at the bottom and immersed in a large container of water. When the tube is moved up and down, the height of the water column varies, and the equilibrium pressure and volume of trapped air must vary in response. This observation is the foundation of a recently proposed method for determining atmospheric pressure (J. Brody, K. Rohald, and A. Sutton, "Determining atmospheric pressure with a eudiometer and glycerol," Journal of Chemical Education **87**, 1367-1368 (2010)).

While developing this method for determining atmospheric pressure, a further observation was made: when the tube is displaced vertically, the water column oscillates for several seconds before settling into its new equilibrium position. The water column behaves like a mass on a spring, and yet no spring is present! What plays the role of a spring in this system? What determines the frequency of oscillation? Can we develop a theoretical model to test experimentally?

Your goal is to derive an equation for the oscillation frequency as a function of any necessary parameters (tube length, equilibrium height of the water column, distance from the bottom of the tube to the top of the water in the reservoir, etc.). ***The derivation isn't trivial!*** Perhaps your experimental results can help guide your theory, which is how real science works (and how I derived an expression that agrees with measured data). If you need any hints, please ask!

Your **theoretical expression** for **frequency** should be compared with **experimental results** for **various equilibrium heights** in **various tubes**. To take data, vigorously displace the tube vertically, and then immobilize it in a clamp. Record the oscillations with a digital camera, held sideways (right side up, left side down). Use Fine mode (30 frames per second). If you can see the (refracted light from the) metal pole through the water column, you'll easily be able to find the meniscus in the recorded images. Measure all important distances. Be sure to record a distance between two fixed points that are identifiable in the recorded images. You will need this to calibrate the video-processing software.

Using the Tracker software

Run the Tracker software from http://physlets.org/tracker/.

Under the Video menu, select Import, and choose your file.

Drag the "slider" below the video to select a starting frame. Right click on the image, choose Clip Settings, and enter the number of the starting frame (the red number in the lower left corner).

Drag the slider toward the end of the video, stopping at a frame in which the water column has come to equilibrium. The origin of our coordinate system will be center of the meniscus at equilibrium. Click on the "axes button" (to the left of **Create**) and drag the origin to the center of the meniscus. Right click on the +x axis and deselect Visible.

Click the "calibration button" (to the left of the axes button) and choose **calibration stick**. Manipulate the stick until it spans a known distance. Click on the number above the stick and enter the actual distance (your choice of units).

Click Create and choose Point Mass. Go to your starting frame, hold Shift, and left click on the center of the meniscus. Right click on the symbol that appears and deselect Visible. The frame has automatically advanced. Continue holding Shift and clicking on the meniscus until you reach equilibrium.

Curve-fitting in Python

To export the data to Python, right click on the data, and choose Copy Data -> Full Precision. Paste into Excel (as an intermediate step). The first two rows are titles.

Then open Spyder and run the following Python script by copying it into the “Editor pane” on the left and clicking the green “Run” arrow (if you want Spyder on your computer, download Anaconda from https://www.continuum.io/downloads):

```
import numpy as np, matplotlib.pyplot as plt, pandas as pd

from scipy.optimize import curve_fit

from sklearn.metrics import r2_score

print ('Your x data and y data should appear as columns in Excel\n')

input("Copy your x data to clipboard and press Enter.")

x=pd.read_clipboard(header=None)

x=np.array(x).T[0]

input("Copy your y data to clipboard and press Enter.")

y=pd.read_clipboard(header=None)

y=np.array(y).T[0]

def func(t, a, b, c, d):

	return a*np.exp(-b*t)*np.cos(c*t+d)

popt, pcov = curve_fit(func, x, y)

plt.plot(x,y,'o')

yfit=func(x,*popt)

plt.plot(x,yfit)

perr = np.sqrt(np.diag(pcov))

r2=r2_score(y, yfit)

print("The fitting parameters are ",popt)

print("The uncertainties in the fitting parameters are ",perr)

print("The coefficient of determination is ", r2)
```

The fit gives you the best estimate of the observed frequency. (The angular frequency is the third fitting parameter, c, in the Python script above.) Compare with your theoretical expression. Alter the amount of water in the tube. Repeat.