SWINGING GATE PENDULUM SEISMOMETER

CASPAR LANT

Intermediate Experimental Physics II

Section:	002
Date Performed:	February 16, 2016
Date Due:	February 23, 2016
Partner:	Neil Saddeler
Professor:	Prof. Andrew Kent
Instructor:	David Mykytyn

Date: February 22, 2016.

The Objective of this week's experiment was to put our vast theoretical knowledge of lenses to application. It is always a remarkable think to see what was once pure abstraction validated though rigorous scientific experimentation.

1. Theoretical Background/ Abstract

Lenses have been of interest to us humans for a long time now. Convex lenses, concave lenses, even planar mirrors have all been the subjects of frenzied study through the ages. This should come as no surprise, given how fantastically useful they are to creatures who experience the world, mainly, through sight. We are mainly interested in what happens to the light from objects in front of the lens.

$$\frac{1}{s_i} + \frac{1}{s_o} = \frac{1}{f} \tag{1}$$

$$\frac{1}{f} = (n-1)\Big(\frac{1}{R_1} - \frac{1}{R_2}\Big) \tag{2}$$

2. Experimental Procedure

(1) Level the seismometer by adjusting the thumbscrew on its base. It turns out that the best way to do this is not with the provided bubble level, but by eye. You should dial adjust the screw to a height such that the device's pendulum finds its equillibrium position at the middle of the seismometer base.

(2) Turn on data studio and set up a display of pendulum position versus time.

(3)

(4)

(5)

(6)

(7)

(8)

(9)

(10)

(11)

(12)

(13)

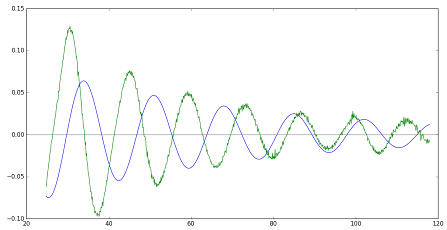
(14)

```
# -*- coding: utf-8 -*-
# Reads in data from data file, plots it and compares it to
# an exponentially decaying sinusoidal signal
import numpy as np
import matplotlib.pyplot as plt
from datetime import datetime
import os

FILENAME = "blow5.txt"
TIMESTAMP = datetime.now().strftime('%H:%M:%S')
OUTPUT = ''+ os.path.splitext(FILENAME)[0] + '_' + TIMESTAMP + '.png'
print OUTPUT

# Function defining exponentially decaying sinusoidal
def thetaU(t, A, w0, Q, phi):
    wu = w0 * np.sqrt(1.0-1.0/(4.0*Q*Q))
    return A * np.exp(-0.5*w0*t/Q) * np.sin(wu*t + phi)
# User sets these parameters to best match the data
A = 0.25
T = 14.25
Q = 9
# Amplitude
# Period (w0 = 2 pi / T)
# Q = quality factor (determines damping)
phase = 40                    # phase in degrees
# Convert phase in degrees to phase in radius for use in function
phi = np.pi * phase / 180.0      # phase in radians
time, theta = np.loadtxt(FILENAME, skiprows=130 , unpack = True)
fit = thetaU(time, A, 2.0*np.pi/T, Q, phi)
plt.plot(time, fit)
plt.axhline(color="gray")
plt.plot(time, theta)
plt.show()

fig = plt.figure()
fig.savefig(OUTPUT)
# format='png',
        # transparent=True, bbox_inches=None, pad_inches=0.1,
        # frameon=None)
```



hello

FIGURE 1. blow 5

3. Graphs and Tables

4. Questions

(1) *Question?*

Answer

(2) *Repeat your measurements both with and without the vibration isolation mounts for the wooden table. Do you see a difference? Why or why not?*

Answer

5. Error Analysis