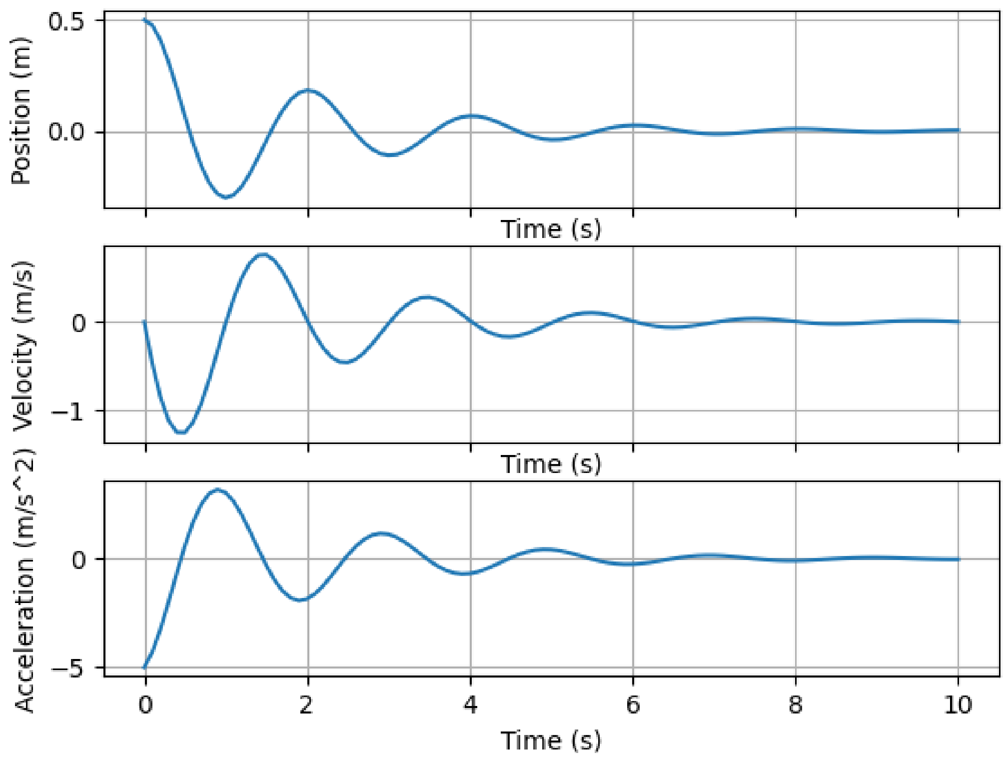
Dynamic Systems & Controls Lab 3 – Simulation of dynamic systems using Python

Kieran Cosgrove

TA: Seong-Hyo Ahn

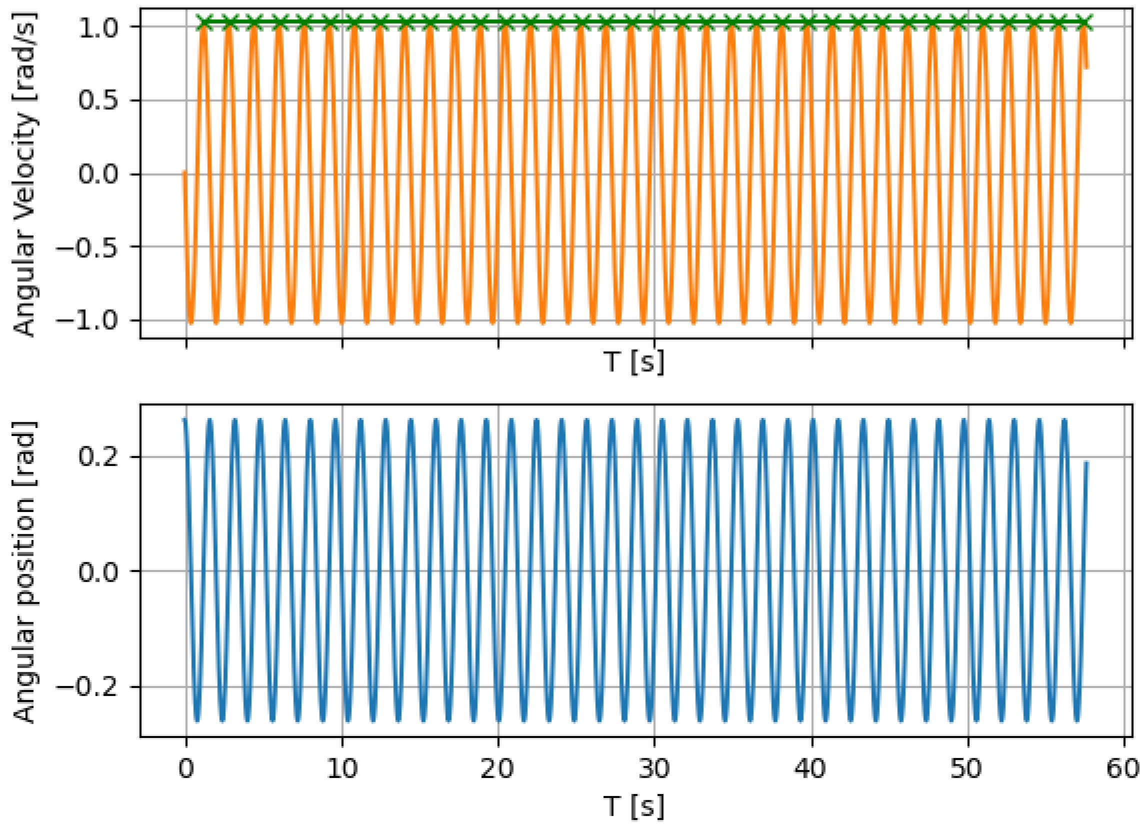
1. Mass-spring-damper simulation



The main change to allow for these graphs to be created is calculating the acceleration given the position of the mass, which is calculated using the following code changes:

```
accels = []
for i in range(len(t)):
    xdot, y = mkb(sol[i,:], t[i], m, k, b)
    accels.append(y)
```

2. Simulation results for bifilar pendulum



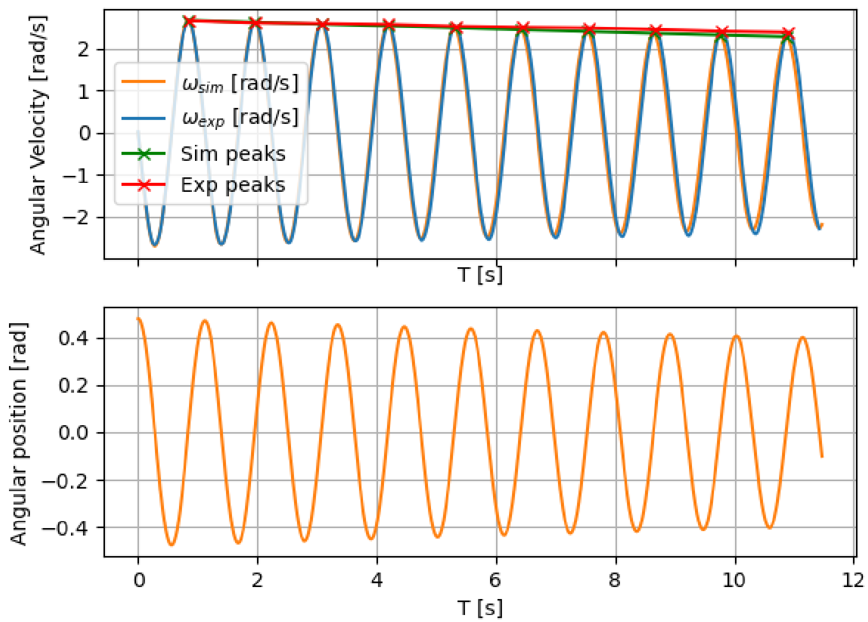
Key changes I made to my code to allow for this include putting in the theoretical moment of inertia of my phone. Also, adding modifying the forcing function from a mass to a moment of inertia was required from the pendulum model.

```
def bifilar_pend(x, t, m, g, R, L, J, b):
    xdot1 = x[1]
    xdot2 = (1/J)*(-(R**2*m*g/L)*np.sin(x[0]) - b*x[1])
    # specify outputs
    y = 0

    return np.array([xdot1, xdot2]), y
```

```
#----------------------------------------------------------------
g = 9.81          # [m/s^2], gravitational acc
L = 0.7112 # [m],  bifilar string length
R = 0.0445 # [m],  length between files/2
Lp = 0.1447 # [m],  length of phone (horizontal --> parallel to ground)
t = 0.008 # [m],  thickness of phone
m = 0.151 # [kg], phone mass
```

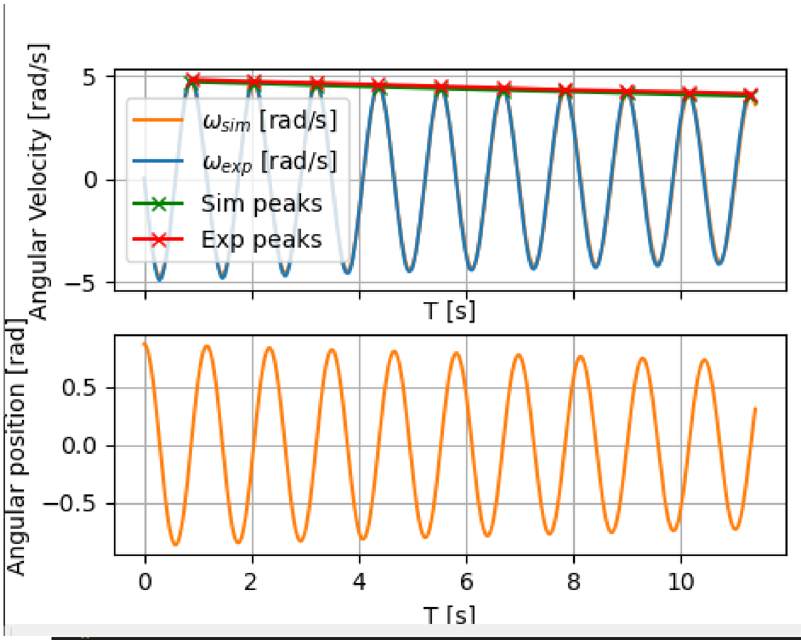
3. Simulation of bifilar pendulum with a small angle




```
#----------------------------------------------------------------
J   = 0.000218 # [kg*m^2]
b   = 0.000007    # [Solve for units], damping coefficient
x_0 = 27.5        # [deg]
```

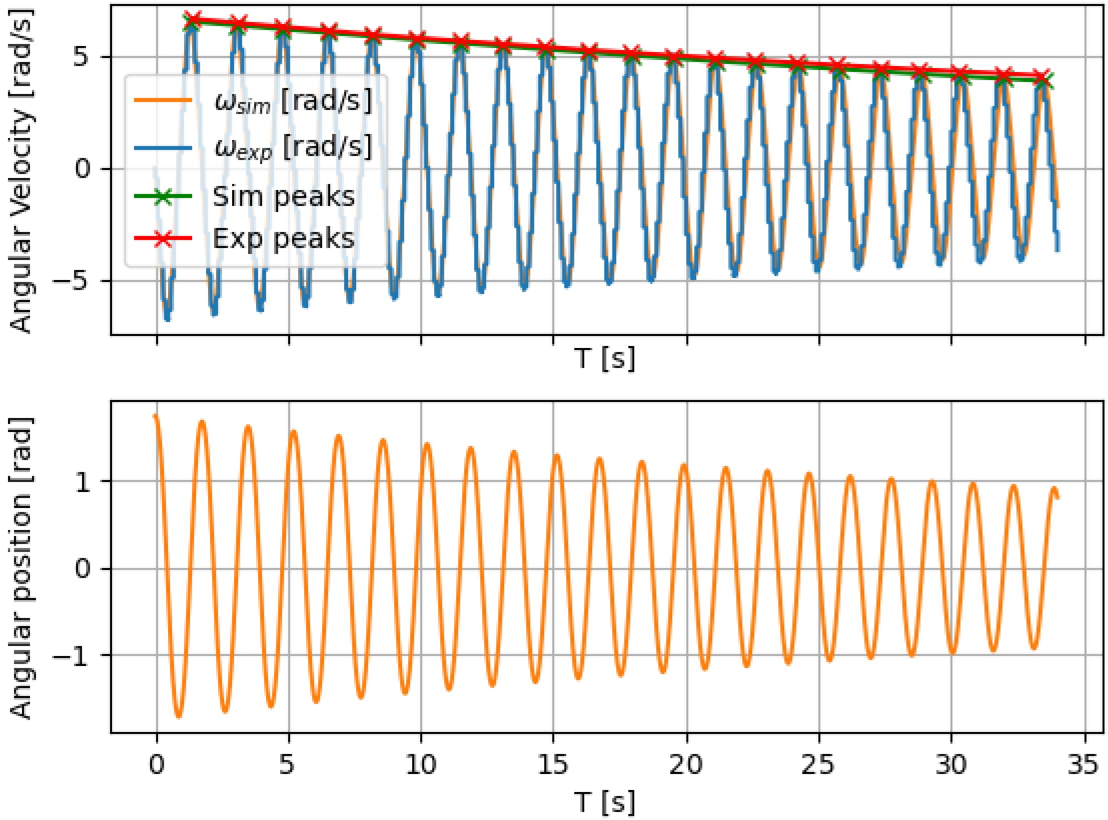
This momentum of inertia is much lower than the one calculated from lab2, which calculated it should be .000267 kg*m^2, which is the same as the theoretical value. Also, the initial angle & damping term were adjusted such that experimental and physical results would lineup.

4. Now that damping and moment of inertia has been found, large angles can be tested. Angles of about 45 and 90 degrees are tested below.




```
J   = 0.000223 # [kg*m^2]
b   = 0.000007    # [Solve for units], damping coefficient
x_0 = 50        # [deg]
```

An angle of 45 degrees found that just the moment of inertia had to be slightly adjusted, but the results were relatively close. Of note, the test setup was slightly different for the large angles than the small angles due to them being completed on different days with different length string.




```
#----------------------------------------------------------------
J   = 0.000218 # [kg*m^2]
b   = 0.0000075    # [Solve for units], damping coefficient
x_0 = 100        # [deg]
```

From the greatest angle, the inertia was the same as the small angle, but the damping has to be slightly adjusted from the small angle testing.

5. Experimental and Simulation Data Misalignment

The angle I observed always seemed to be smaller than the angle I put in code to get the amplitudes of the curves to match. But it was very hard to measure this, however it seemed to have very little influence in the moment of inertia and damping as they were relatively close across experiments. Another factor that probably affected my lab was the fan running in the background, as the airflow could have induced forces onto the phone to increase the amplitudes. It was also hard to release the phone without causing some initial motion, which caused data issues.