UNIVERSITY OF VICTORIA

Department of Mechanical Engineering

MECH 242 – Dynamics

Pendulum Oscillation Project Report

To: Dr. Flavio Firmani

Group 07

Dylan Kerr - V00942921

Nicholas Warren - V00936614

Blaine Tubungbanua - V00918128

Maya Berglof - V00874070

Abstract

This project attempts to model a pendulum that is based upon a real physical pendulum. This is done by using 3 different models, each being more involved then the next. We first model a simple pendulum with a weighted bob, then progress to a solid rod and finally we have a compound rigid body pendulum. We model these theoretical pendulums using Matlab. The theoretical compound rigid body pendulum is based on a real compound pendulum that will be used to calibrate a friction coefficient through linear regression of experimental and theoretical data. This data was gathered from the real pendulum via two experiments, one to analyse the period and another to analyse the acceleration, velocity and angular position. The matlab analysis makes use of plots to graph our theoretical finding and experimental data. Finally a conservation of energy analyses is done to find the maximum velocity for comparison of the models.

1.0 Introduction

A pendulum is a simple mechanical system composed of a hanging weight suspended from a fulcrum which is allowed to oscillate freely. When the weight is rotationally displaced from its equilibrium position, it experiences a restorative force due to gravity that will cause the weight to accelerate back towards its initial position. The weight will continue to swing back and forth around the initial position as the acceleration from the restoring force changes direction. Historically, these systems have been used in timekeeping instruments and scientific devices, such as accelerometers and seismometers. They have also been used to measure the acceleration of gravity and were once used as a standard of length.

The objective of this project is to gain an understanding of the dynamic relations related to a pendulum system through an analysis of particular kinematic parameters. Project teams were instructed to design, model, simulate, and verify the oscillation of a pendulum through a hands-on experimentation session. Analytical models of a simple, physical, and compound rigid body pendulum were used to create MATLAB simulations of the oscillation path. These simulations would then be compared to the movement of the handmade pendulum. By comparing the experimental angular and linear kinematic vectors of the group's pendulum to the theoretically derived simulations, groups can verify the relation between mathematical and practical pendulum models.

1.1 Simple Pendulum

A simple pendulum system, also known as a pendulum bob, is defined to have a point mass **m** that is suspended by a massless unstretchable string of length **L**. In this setup, the only active forces on the point mass are the force from gravity and the tension from the string. Though a real pendulum is a more complex system, this simplified model will provide the foundation for our full-scale theoretical model. By applying Newton's second law for rotation, we can derive the system's equation of motion. The simple pendulum coordinate diagram and the corresponding free body diagram (FBD) is shown below in Figure 1.

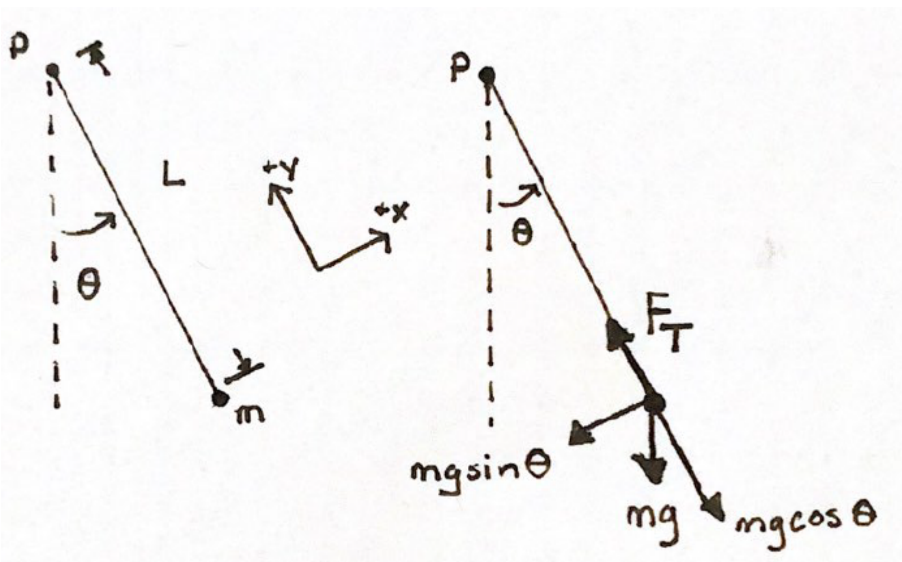

Figure 1: Pendulum Bob Coordinate and FBD Diagrams

The torque developed around the fulcrum point P can be represented by the expression:

$$= \ -mgLsin\theta$$

The moment of inertia of the point mass about the pivot point is known to be:

$$I_P \ = \ mL^2$$

By substituting these expressions into the conversation of energy equation, we can determine the rotational equation of motion:

$$= \ I_P\alpha \ = \ I_P\frac{d^2\theta}{dt^2}$$

$$-\ mgLsin\theta \ = \ mL^2\frac{d^2\theta}{dt^2}$$

$$0 \ = mL^2\frac{d^2\theta}{dt^2} \ + mgLsin\theta$$

If we assume that the amplitude of the pendulum's oscillation is small, we can use the small amplitude approximation to simplify the rotational equation of motion. For angles below 15°, we can substitute sinθ ≈ θ to obtain:

$$0 \ = mL^2\frac{d^2\theta}{dt^2} \ + mgL\theta$$

This expression now takes the form of a second-order linear differential equation which can be further simplified and evaluated explicitly.

1.2 Simple Rigid Body Pendulum

Similar to a pendulum bob, a simple rigid body pendulum consists of an object whose oscillations are similar to those of a simple pendulum but the inertial forces of the object's mass distribution must be included into the rotational motion equation. Instead of gravity working at the center of the point mass, we now have a system where gravity works at the center of mass on the object. Following the same procedure outlined in section 1.1, we can again derive the system's equation of motion through torque analysis. The simple rigid body pendulum coordinate and FBD diagrams are shown below in Figure 2. In accordance with the design of the handmade pendulum, a long thin rod was chosen as the generic simple body as the location of the center of mass and the moment of inertia were readily obtainable.

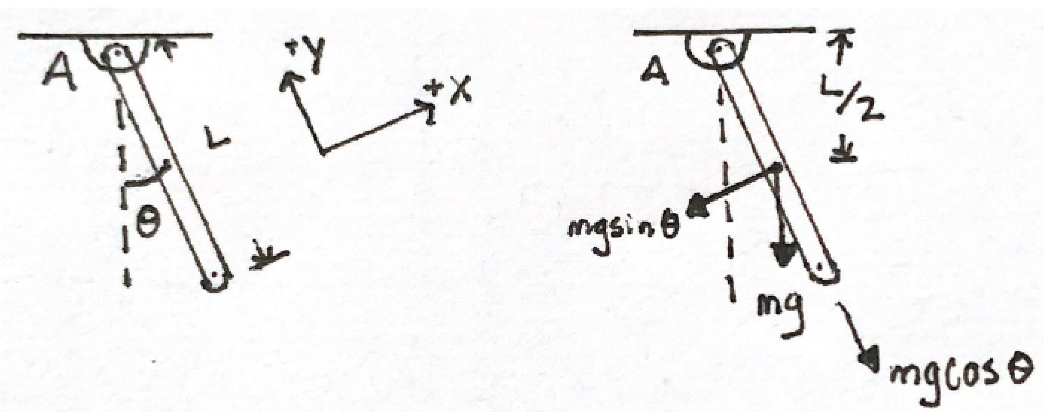

Figure 2: Simple Rigid Body Pendulum Coordinate and FBD Diagrams

The torque developed around pivot point A can be represented by the expression:

$$= \ - mg\frac{L}{2}sin\theta$$

The moment of inertia of the simple rigid body around the fulcrum point A can be obtained using the parallel-axis theorem:

$$I_P \ = I_G + \ mL^2$$

$$I_P \ = \frac{1}{12}mL^2 + \ \frac{1}{2}mL^2 \ = \ \frac{1}{3}mL^2$$

Using these expressions and applying the small angle approximation, we can derive the rotational equation of the simple rigid body pendulum:

$$= \ I_P\alpha \ = \ I_P\frac{d^2\theta}{dt^2}$$

$$- mg\frac{L}{2}sin\theta \ = \ \frac{1}{3}mL^2\frac{d^2\theta}{dt^2}$$

$$0 \ = \frac{1}{3}mL^2\frac{d^2\theta}{dt^2} \ + mg\frac{L}{2}\theta$$

This rotational equation was slightly more complex to determine than the one representing a pendulum bob, though it also takes the form of a second-order linear differential equation which can now be evaluated explicitly.

1.3 Compound Rigid Body Pendulum

The simple rigid body pendulum model more accurately portrays the experimental pendulum design than the pendulum bob model, though it does not take into account the presence of the accelerometer placed at the end of the rod. This device adds additional restorative forces acting on the object's oscillating movement, and therefore must be considered in the theoretical model. We can represent this system through a compound rigid body model in which the arm and bob are two fixed bodies rotating around the same pivot point. Shown below in Figure 3 is the coordinate and FBD diagram of the compound rigid body model.

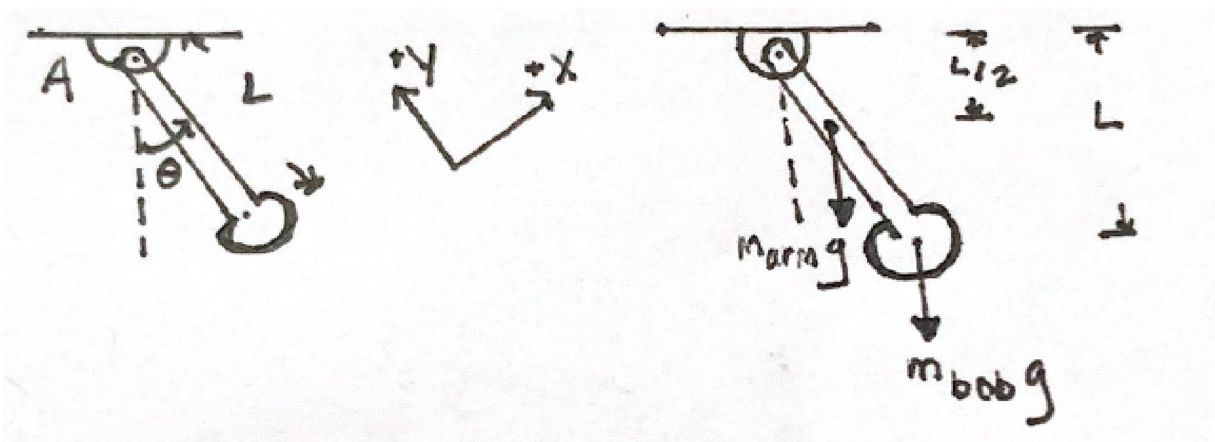

Figure 3: Compound Body Pendulum Coordinate and FBD Diagrams

In this system, we need to account for the rotation vectors originating from the center of the arm and from the center of the bob. The torque developed around the fulcrum point A can be represented by the expression:

$$= \; -(\frac{L}{2}m_{arm} + Lm_{bob})gsin\theta$$

The moment of inertia of the compound rigid body around pivot point A can be obtained by applying the parallel axis theorem like before. This time, we need to treat each body separately and determine their respective moments, then sum the results to obtain an expression for the overall inertial force. We can use the expressions previously obtained to simplify this process:

$$I_P = I_{rod} + I_{bob}$$

$$I_{rod} = \frac{1}{3}m_{arm}L^2$$

$$I_{bob} = m_{bob}L^2$$

$$I_P = \frac{1}{3}m_{arm}L^2 + m_{bob}L^2$$

For the purpose of keeping this model as simple as possible, a bob with the same moment of inertia as a point pass derived in section 1.1 was chosen as the object to represent the accelerometer. Substituting these expressions into the conversation of energy equation and applying the small angle approximation will result in the system's rotational motion equation:

$$= I_P\alpha = I_P\frac{d^2\theta}{dt^2}$$

$$-(\frac{L}{2}m_{arm} + Lm_{bob})gsin\theta = (\frac{1}{3}m_{arm}L^2 + m_{bob}L^2)\frac{d^2\theta}{dt^2}$$

$$0 = (\frac{1}{3}m_{arm}L^2 + m_{bob}L^2)\frac{d^2\theta}{dt^2} + (\frac{1}{2}m_{arm} + m_{bob})gL\theta$$

Just as before, this expression takes the form of a second-order linear differential equation which can be evaluated explicitly. Besides the frictional forces present at the fulcrum, this model takes into account all the corresponding force vectors which will affect the oscillation patterns of the handmade pendulum.

2.0 Differential Analysis

The rotational equations of motion for the three models created in section 1 are all second-order linear differential equations. The closed form solution to each equation can be explicitly determined, and will take the form of:

$$\theta(t) = Acos(\omega_0 t) + Bsin(\omega_0 t)$$

where ω_0 represents the oscillation angular frequency of the pendulum and is a function of the design parameters of the pendulum model. These solutions can be easily determined using symbolic computation in MATLAB.

2.1 Simple Pendulum Simulation

The angular frequency of the simple pendulum was determined using symbolic derivation methods. The MATLAB code presented in Figure 4 was used to produce the closed-form solution.

```
%Pendulum Bob Differential Analysis

clear all;%Clears all variables that may be stored in the 'workspace'
close all;%Closes all plots that may be open
clc;%Clears command window

syms Q(t) Q0 m L g
DQ = diff(Q);
ode = m*L^2*diff(Q,t,2)+m*g*L*Q;
cond1 = Q(0) == Q0;
cond2 = DQ(0) == 0;
conds = [cond1 cond2];
QSol(t) = dsolve(ode, conds);
QSol = simplify(QSol);
```

Figure 4: MATLAB Symbolic Derivation Code for Pendulum Bob Rotational Motion Equation

The closed form solution was determined to be:

$$\theta(t) = \theta_0 cos(\sqrt{\frac{g}{L}}t)$$

where θ_0 is the initial release angle.

Using the expression for angular frequency, we can further solve for the period of the system:

$$\omega_0 = \sqrt{\frac{g}{L}}$$

$$T = \frac{2\pi}{\omega_0} = 2\pi\sqrt{\frac{L}{g}}$$

A capture of the actual command window output for this code is provided in Figure 22 of Appendix A.

2.2 Simple Rigid Body Pendulum Simulation

The angular frequency of the simple rigid body pendulum was determined using the same symbolic derivation method used for the pendulum bob. The only difference between this block of code and the one used for the pendulum bob model is the expression for the ordinary differential equation function, as this model has identical initial condition constraints. The MATLAB code presented in Figure 5 was used to produce the closed-form solution.

```
%Simple Rigid Body Differential Analysis

clear all;%Clears all variables that may be stored in the 'workspace'
close all;%Closes all plots that may be open
clc;%Clears command window

syms Q(t) Q0 m L g
DQ = diff(Q);
ode = (1/3)*m*L^2*diff(Q,t,2)+m*g*L*0.5*Q;
cond1 = Q(0) == Q0;
cond2 = DQ(0) == 0;
conds = [cond1 cond2];
QSol(t) = dsolve(ode, conds);
QSol = simplify(QSol);
```

Figure 5: MATLAB Symbolic Derivation Code for Simple Rigid Body Pendulum Rotational Motion Equation

The closed form solution was determined to be:

$$\theta(t) = \theta_0 cos(\sqrt{\frac{6g}{L}\frac{t}{2}}) = \theta_0 cos(\sqrt{\frac{3g}{2L}}t)$$

The angular frequency and period of the system can now be solved:

$$\omega_0 = \sqrt{\frac{3g}{2L}}$$

$$T = \frac{2\pi}{\omega_0} = 2\pi\sqrt{\frac{2L}{3g}}$$

A capture of the actual command window output for this code is provided in Figure 23 of Appendix A.

2.3 Compound Rigid Body Pendulum Simulation

The compound body rigid pendulum rotational motion equation was also manipulated using symbolic derivation in MATLAB like the other two models using the same initial condition constraints. Like before, the only part of the code block that was changed was the expression for the ordinary differential equation function. The code presented in Figure 6 was used to produce the solution:

```
%Compound Rigid Body Differential Analysis

clear all;%Clears all variables that may be stored in the 'workspace'
close all;%Closes all plots that may be open
clc;%Clears command window

syms Q(t) Q0 m_arm m_bob L g
DQ = diff(Q);
ode = (((1/3)*m_arm*L^2) + (m_bob*L^2))*diff(Q,t,2)+(((1/2)*m_arm) + (m_bob))*g*L*Q;
cond1 = Q(0) == Q0;
cond2 = DQ(0) == 0;
conds = [cond1 cond2];
QSol(t) = dsolve(ode, conds);
QSol = simplify(QSol);
```

Figure 6: MATLAB Symbolic Derivation Code for Compound Rigid Body Pendulum Rotational Motion Equation

The closed form solution was determined to be:

$$\theta(t) = \theta_0 cos(\frac{\sqrt{6g(m_{arm} + 2m_{bob})(m_{arm} + 3m_{bob})}}{(2m_{arm} + 6m_{bob})\sqrt{L}} t)$$

The angular frequency and period of the system can now be solved for this system:

$$\omega_0 = \frac{\sqrt{6g(m_{arm} + 2m_{bob})(m_{arm} + 3m_{bob})}}{(2m_{arm} + 6m_{bob})\sqrt{L}}$$

$$T = \frac{2\pi}{\omega_0} = 2\pi(2m_{arm} + 6m_{bob})\sqrt{\frac{L}{6g(m_{arm} + 2m_{bob})(m_{arm} + 3m_{bob})}}$$

A capture of the actual command window output for this code is provided in Figure 24 of Appendix A.

3.0 Pendulum Design

The designed pendulum was made from wood, aluminium wire, tape, eye hook and a phone. The finish pendulum can be seen below in Figure 7. The rigid rod was 1.00m long with an eye hook attached at one end. A hole was drilled on one end to allow for the wire to be attached securely to the rod by fishing the wire through the hole and using tape to stop any "bob" only movement. The wire was bent a certain way to allow the phone to be securely held. Table 1 has the relevant measured parameters of the pendulum necessary for completing the kinematic analysis.

	Mass (g)	Length (m)
Rod	307	1.00
Bob	210	0.0762
Total	517	1.0762

Table 1: Parameters of the Pendulum

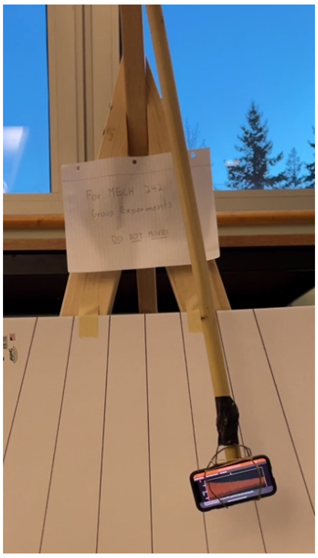

Figure 7: Pendulum mid swing

4.0 Kinematic Analysis

Using the measured pendulum parameters from section 3.0, the angular and linear kinematic vectors describing the pendulum's motion were analyzed. The following section provides further context on the calculation and simulation processes used.

4.1 Angular Displacement, Velocity and Acceleration Analysis

Using the differential equations solved in section 2.0, the simple pendulum, rigid-body pendulum, and compound-rigid-body pendulum were modelled in MATLAB. A capture of the code block used to determine the kinematic vectors is shown below in Figure 8.

```
g = 9.81; %gravity
Q0 = 15; %Initial theta
L = 1 + 0.0762/2; %Length of pendulum + length of bob/2 (length to COM)
m_arm = 0.307;
m_bob = 0.210;

startTime = 0;
endTime = 60;
dT = 0.1;
t=[startTime:dT:endTime]';




% 4.2 Angular Displacement, Velocity, and Acceleration
% Simple pendulum
theta_s = Q0*cos((g^(1/2)*t)/L^(1/2));
omega_s = diff(theta_s)/dT;
alpha_s = diff(diff(theta_s))/dT;
% Rigid Body
theta_rb = Q0*cos((6^(1/2)*g^(1/2)*t)/(2*L^(1/2)));
omega_rb = diff(theta_rb)/dT;
alpha_rb = diff(diff(theta_rb))/dT;
% Compound Rigid Body
theta_crb = Q0*cos((6^(1/2)*g^(1/2)*t*(m_arm + 2*m_bob)^(1/2)*(m_arm + 3*m_bob)^(1/2))/(L^(1/2)*(2*m_arm + 6*m_bob)));
omega_crb = diff(theta_crb)/dT;
alpha_crb = diff(diff(theta_crb))/dT;
```

Figure 8: Kinematic Analysis, Angular Calculations

These values were plotted over a span of 60 seconds, shown below in figures 9 and 10.

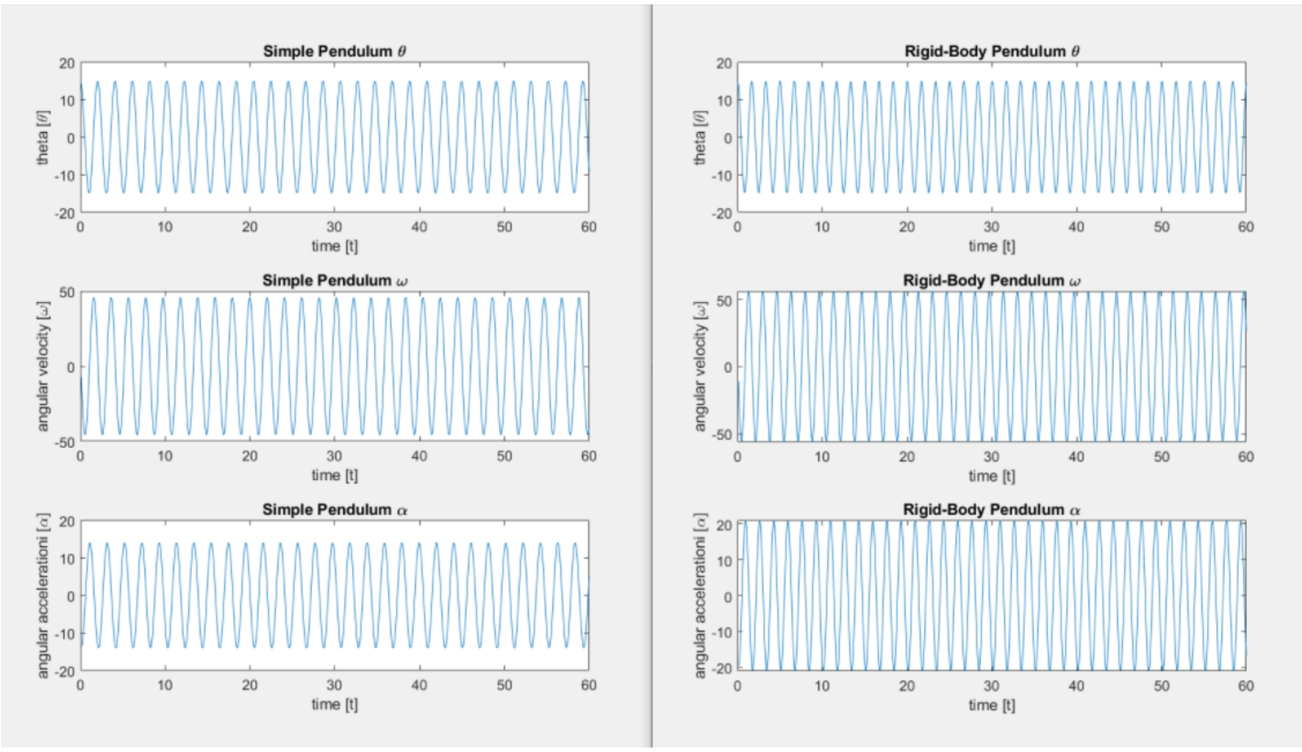

Figure 9: Simple and Rigid-Body Pendulum plots

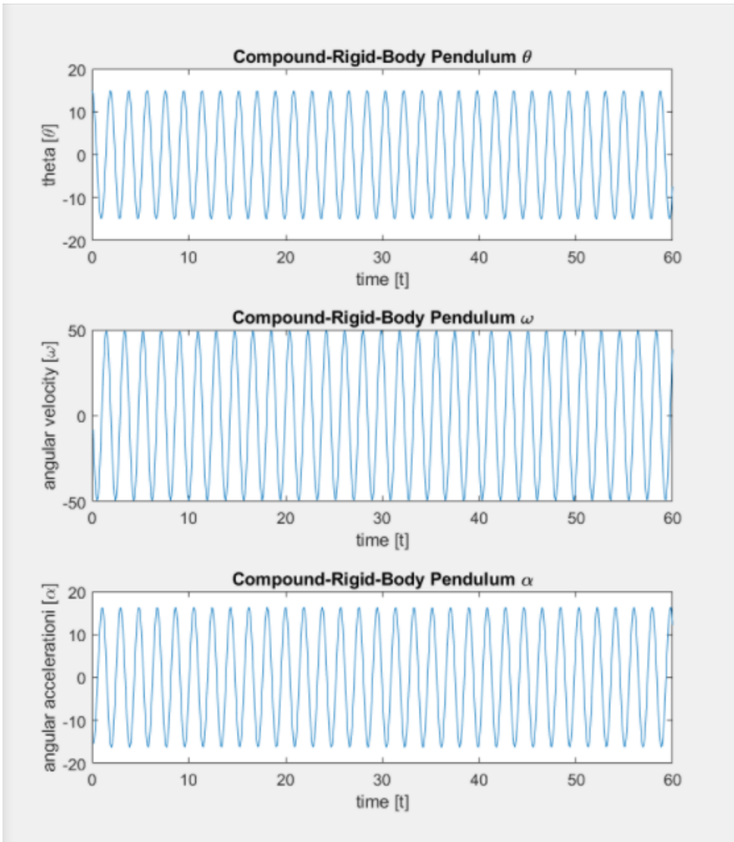

Figure 10: Compound-Rigid-Body Pendulum Plot

4.3 Linear Displacement, Velocity and Acceleration Analysis

The angular displacement, velocity, and acceleration calculations were then transformed to cartesian coordinates, making the data easier to compare with the sensor data gathered from the smartphone attached at the bottom of the pendulum.

```
startTime = 0;
endTime = 60;
dT = 0.1;

g = 9.81; %gravity
Q0 = 15; %Initial theta
L = 1 + 0.0762/2; %Length of pendulum + length of bob/2 (length to COM)
m_arm = 0.307;
m_bob = 0.210;

t=[startTime:dT:endTime]';

%4.3 Linear Displacement, Velocity, and Acceleration
% Simple Pendulum
pos_s = [L*sind(Q0*cos((g^(1/2).*t)/L^(1/2))), -L*cosd(Q0*cos((g^(1/2).*t)/L^(1/2)))];
vel_s = diff(pos_s)/dT;
accel_s = diff(diff(pos_s)/dT)/dT;

% Simple Pendulum Rigid Body
pos_rb = [L*sind(Q0*cos((6^(1/2)*g^(1/2)*t)/(2*L^(1/2)))), -L*cosd(Q0*cos((6^(1/2)*g^(1/2)*t)/(2*L^(1/2))))];
vel_rb = diff(pos_rb)/dT;
accel_rb = diff(diff(pos_rb)/dT)/dT;

% Compound Rigid-Body Pendulum
theta_crb = Q0*cos((6^(1/2)*g^(1/2)*t*(m_arm + 2*m_bob)^(1/2)*(m_arm + 3*m_bob)^(1/2))/(L^(1/2)*(2*m_arm + 6*m_bob)));
pos_crb = [L*sind(theta_crb), -L*cosd(theta_crb)];
vel_crb = diff(pos_crb)/dT;
accel_crb = diff(diff(pos_crb)/dT)/dT;
```

Figure 11: Kinematic Analysis, Cartesian Calculations

These calculations yielded the following cartesian plots.

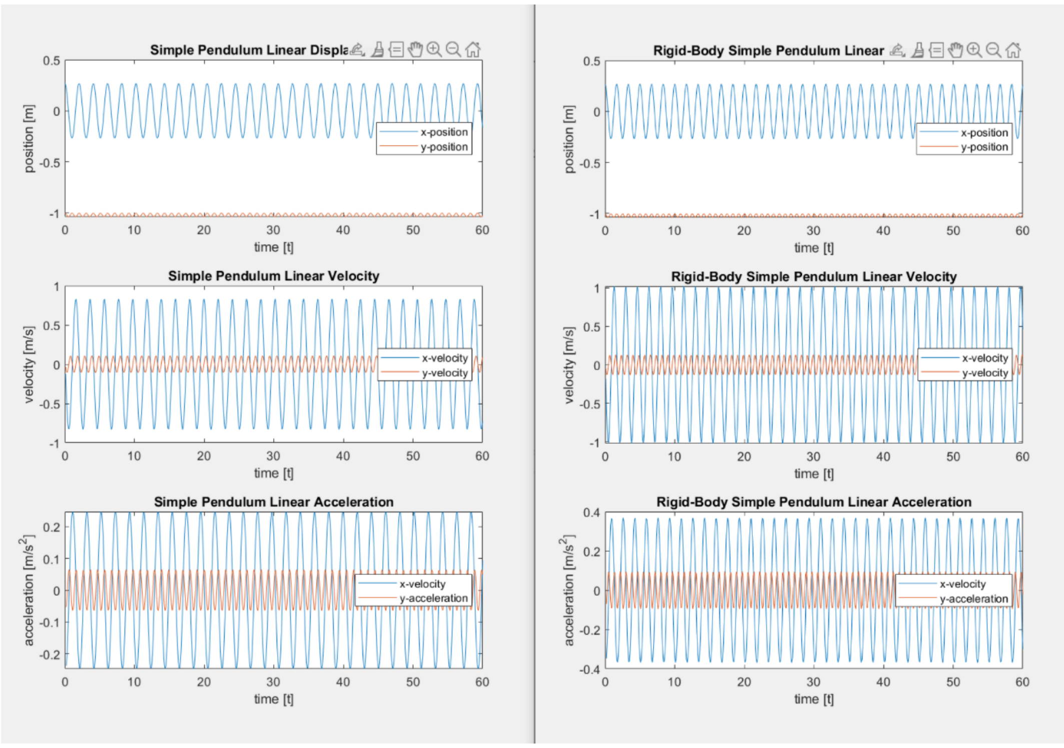

Figure 12: Simple and Rigid-Body-Cartesian Plot

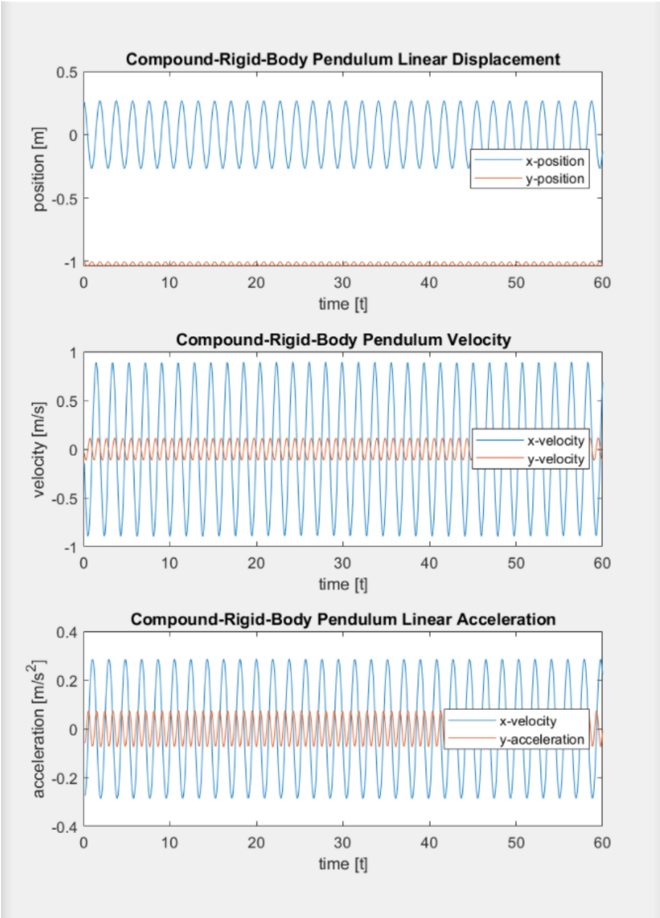

Figure 13: Compound-Rigid-Body Cartesian Plot

5.0 Experimentation

The experiment required a pendulum, a stand and finally a method of capturing data. We used the suggested application *Phyphox* with a cellphone.

The first experiment, finding the period, was conducted as follows:

1. Attach a pendulum to the stand with the phone securely attached and ready to measure the period.
2. Raise pendulum to an initial amplitude of 15° and release.
3. Allow the period to become constant and record the period.

The pendulum's period was determined to be 1.92s with a frequency of 0.52Hz.

The second experiment, Measure Angular Rotation was as followed:

1. Attach a pendulum to the stand with the phone securely attached and ready to measure velocities and linear acceleration, also have a camera ready to capture the trajectory of the pendulum with the angles as a backdrop.
2. Release the pendulum above 15° and wait for the pendulum to stabilize
3. Start measuring and recording once it reaches an initial amplitude of 15°
4. Stop measuring and recording after pendulum as reached an amplitude of 10° (Takes about 60 seconds)
5. From the video recording, record the timestamps for every time the pendulum reaches an amplitude of ±5°

From all the data acquired the rest of the project can be completed. Refer to Appendix B for graphs from the experiment.

6.0 Dynamic Analysis with Friction

Compound rigid body pendulum's oscillate under the force of friction, which opposes the angular movement. The previous models of the simple pendulum, rigid body pendulum, and compound rigid body pendulum, the effects of friction on the frequency of oscillation are neglected. The force of friction in a pendulum of what causes the frequency of oscillation to slowly converge to zero and time progresses.

6.1 Revisited Pendulum Model

Taking our previous knowledge of our compound rigid body pendulum, which closely represents our experimental pendulum, the effect of the opposing frictional force is factored in. This force is always in the opposing direction of motion, which is tangential to the fixed point in our simulation, allowing the effects to be shown in our calculation of torque as a negative constant.

$$= \ -\left(\frac{L}{2}m_{arm} + Lm_{bob}\right)gsin\theta \ - bL\frac{d\theta}{dt}$$

The moment of inertia of the compound rigid body around pivot point A will remain the same as prior to factoring in the effects of friction. Here we continue to treat each body as separate entities to calculate their individual moments. The relation for the sum of torque is then set equal to the moment of interior of the pendulum, representing our compound rigid body pendulum with the effects of friction.

$$= I_P\alpha = I_P\frac{d^2\theta}{dt^2}$$

$$-\left(\frac{L}{2}m_{arm} + Lm_{bob}\right)gsin\theta + bL\frac{d\theta}{dt} = \left(\frac{1}{3}m_{arm}L^2 + m_{bob}L^2\right)\frac{d^2\theta}{dt^2}$$

$$0 = \left(\frac{1}{3}m_{arm}L^2 + m_{bob}L^2\right)\frac{d^2\theta}{dt^2} - bL\frac{d\theta}{dt} + \left(\frac{1}{2}m_{arm} + m_{bob}\right)gL\theta$$

6.2 Pendulum with Friction Differential Analysis

In our kinematic analysis we take our derivation of the sum of torques throughout the pendulum, and factor in our frictional force, which opposes the given direction of motion.

This new term that was derived and introduced into our equation for motion of the pendulum is then manipulated using the same symbolic derivation in MATLAB as all three pendulums described in section 2. The MATLAB code block for determining the friction factor is shown below in Figure 14.

```
clear all;%Clears all variables that may be stored in the 'workspace'
close all;%Closes all plots that may be open
clc;%Clears command window

syms Q(t) Q0 m_arm m_bob L g b %NEW TERMS----
DQ = diff(Q);
ode = (((1/3)*m_arm*L^2) + (m_bob*L^2))*diff(Q,t,2) + b*L*diff(Q,t,1) +(((1/2)*m_arm) + (m_bob))*g*L*Q;
cond1 = Q(0) == Q0;
cond2 = DQ(0) == 0;
conds = [cond1 cond2];
QSol(t) = dsolve(ode, conds);
QSol = simplify(QSol)
```

Figure 14: MATLAB Symbolic Derivation Code for Compound Rigid Body Pendulum with Friction Term

6.3 Pendulum Setup and Experimentation

The experimental setup and procedure used to determine the friction coefficient was the same as the previous process used to determine the kinematic parameters in section 4.0. No changes were made to the setup or experimental process for the second iteration of data collection. Please refer to section 3.0 for the experimental setup of the rigid body pendulum and to section 5.0 for the experimental procedure used.

6.4 Linear regression

In order to determine the coefficient of friction, we need to perform a simplified version of a linear regression with the coefficient b. This is done by varying the constant until they match our theoretical amplitude with the experimental one. For the given data of the pendulum, b is varied until we reach an angular position of 10 degrees at our 60 second mark. After the linear regression, we determined the b constant to be a value of **0.004**.

6.5 Kinematic Analysis

The angular is then realised for the position of the represented pendulum, with a new term ‘b’ introduced. This b term represents the frictional effects on the pendulum, and is varied until it matches the results from our experimental date. Using the section_6_differential_analysis file, with the experimental data inputted we get the following graph (Figure 15) where the theoretical representation of the pendulum oscillation is represented by the blue wave function, and the theoretical data points are represented by the pink circles.

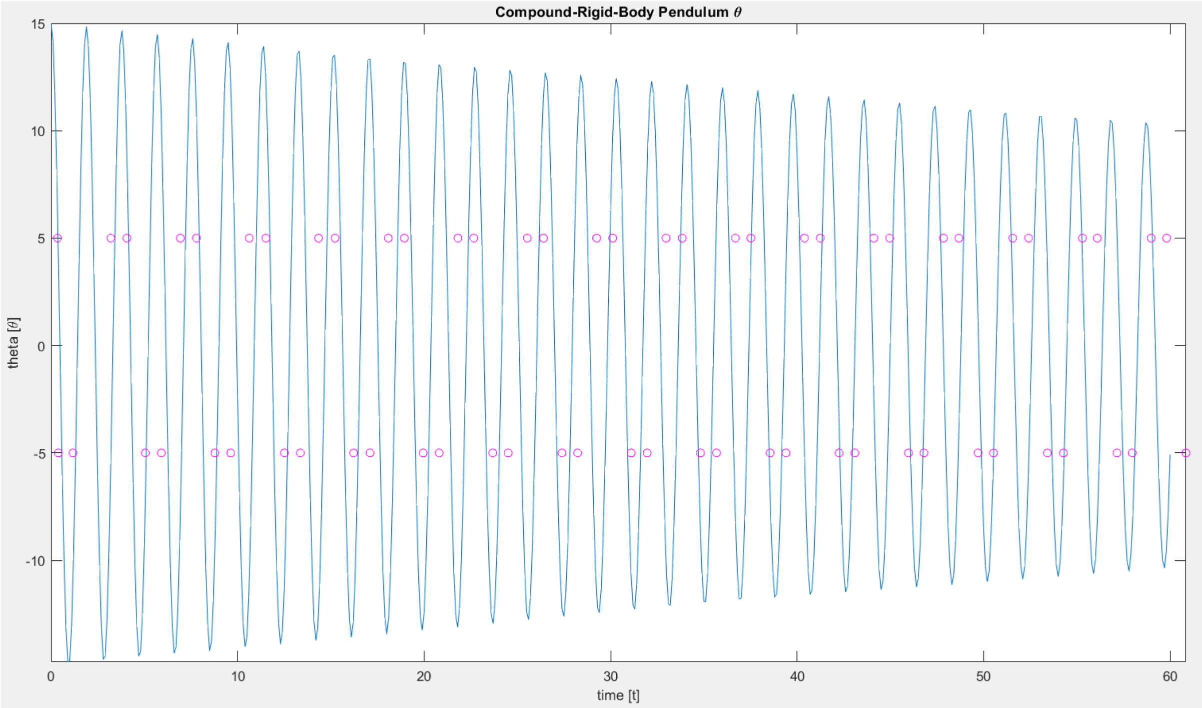

Figure 15: MATLAB experimental data, and theoretical friction overlay angular plot

Once the b value is varied for the coefficient of the frictional force with the angular position wave, the data is plotted for the linear displacement, velocity and acceleration of the bob. Use the same kinematic analysis as in **section 4.3**. Plots shown in *figure 16*, show the adjustment to velocity, displacement and acceleration due to the force of opposing friction.

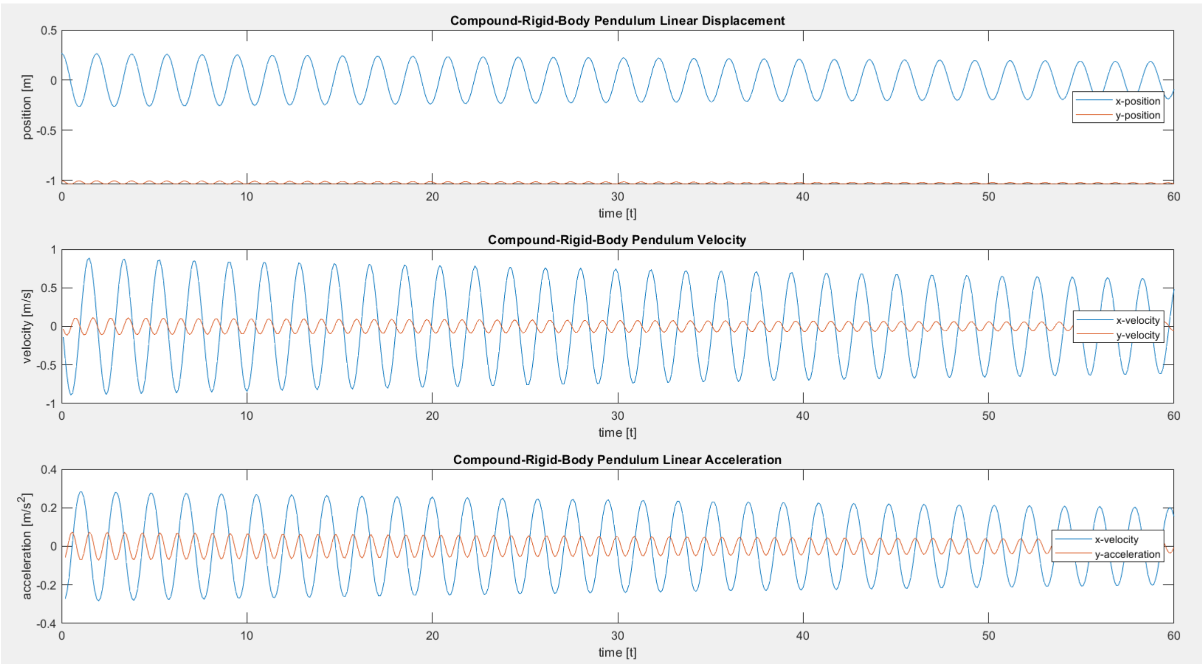

Figure 16: MATLAB experimental data, and theoretical friction overlay linear plot

7.0 Animation

An animation was generated, using the solved differential equation. A transformation matrix was created to convert the computed angle θ to cartesian coordinates. This matrix was multiplied into 8 vertices which composed the hammer-shaped pendulum. The frame rate was set to match the specified dT of 0.1s, so the animation accurately represents the frequency of the pendulum model. Figure 17 shows a capture of the pendulum animation, and Figure 18 is the corresponding code module which created the animation.

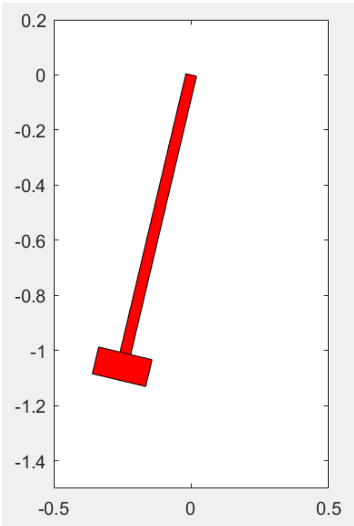

Figure 17: Pendulum Animation

```
startTime = 0;
endTime = 60;
dT = 0.1;
t=[startTime:dT:endTime]';

g = 9.81; %gravity
Q0 = 15; %Initial theta
L = 1 + 0.0762/2; %Length of pendulum + length of bob/2 (length to COM)
m_arm = 0.307;
m_bob = 0.210;
w = 0.02;
w_b = 0.1;
h_b = 0.1;
% Compound Rigid-Body Pendulum
theta_crb = Q0*cos((6^(1/2)*g^(1/2)*t*(m_arm + 2*m_bob)^(1/2)*(m_arm + 3*m_bob)^(1/2))/(L^(1/2)*(2*m_arm + 6*m_bob)));
pos_crb = [L*sind(theta_crb), -L*cosd(theta_crb)];
vel_crb = diff(pos_crb)/dT;
accel_crb = diff(diff(pos_crb))/dT;
%Defining vertex positions
V1i = [w;0];
V2i = [-w;0];
V3i = [-w;-L];
V4i = [w;-L];
V5i = [w_b; -L];
V6i = [-w_b; -L];
V7i = [w_b; -L-h_b];
V8i = [-w_b;-L-h_b];
%Animating
for i=1:1:600
    [i, theta_crb(i)]
    R = [cosd(theta_crb(i)) -sind(theta_crb(i)); sind(theta_crb(i)) cosd(theta_crb(i))];
    V1 = R*V1i; V2 = R*V2i; V3 = R*V3i; V4 = R*V4i; V5 = R*V5i; V6 = R*V6i; V7 = R*V7i; V8 = R*V8i;
    P = [V1,V2,V3, V5, V7, V8, V6, V4,V1];
    Pendulum=fill(P(1,:),P(2,:),'r');
    axis equal; axis([-.5 .5 -1.5 0.2]);
    pause(0.1)
    set(Pendulum,'visible','off')
end
```

Figure 18: Pendulum Animation Code

8.0 Conservation of Energy Analysis

To further confirm the results of the torque analysis completed on the pendulums models, the systems were then solved using the conservation of energy approach. Figures 19 through 21 show the hand calculations for the work and energy analysis of the 3 pendulums models derived in section 1.0. For every model, the maximum velocity was much higher than the velocities calculated during section 4.0.

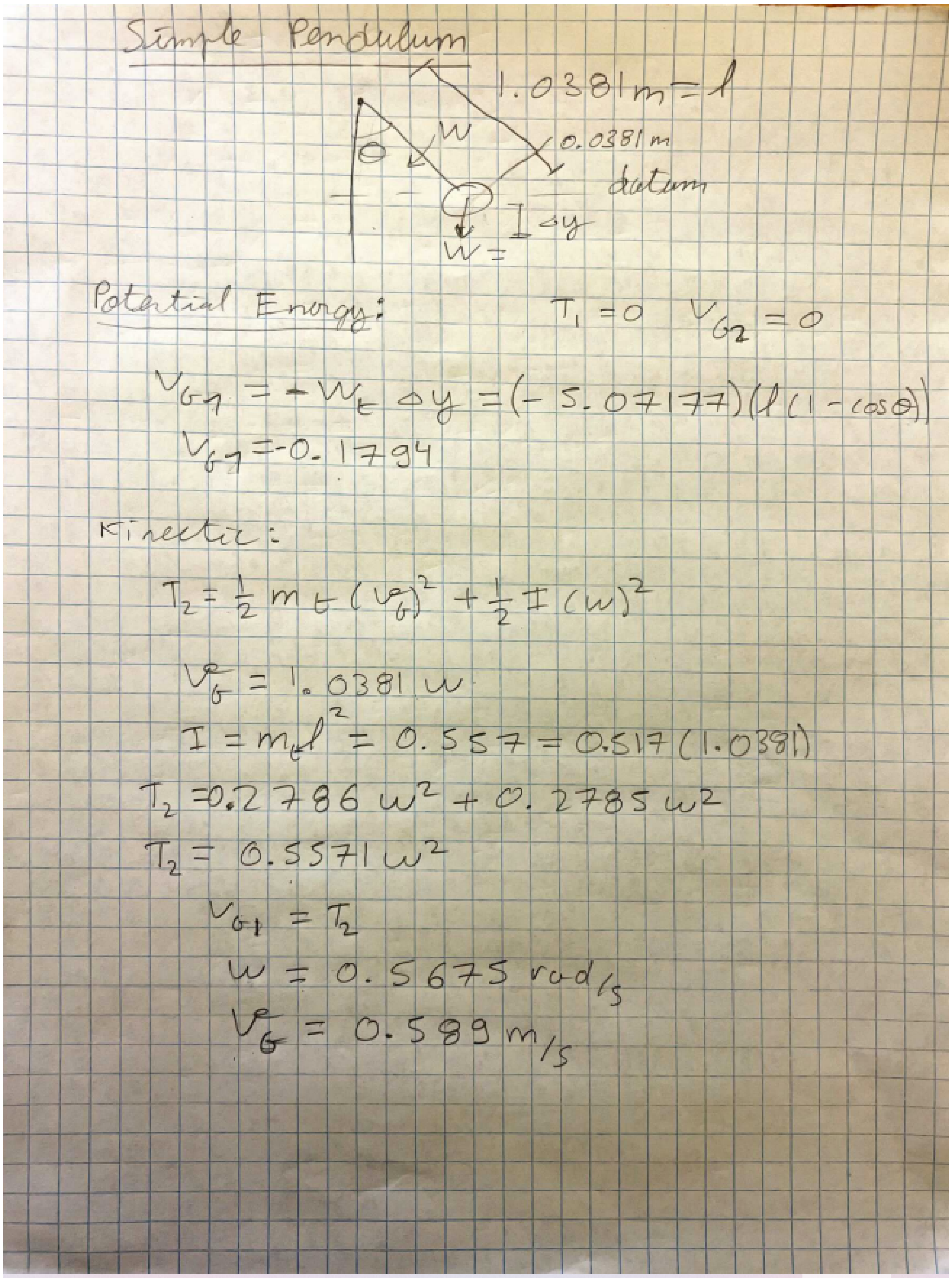

Figure 19: Simple Pendulum Bob Only Calculations

The velocity found using conservation of energy for a simple pendulum is v= 0.589 m/s, the calculations are shown in Figure 19 when comparing this to the graph found in Section 4, we can see a minimum difference in terms of speed. For section 4, we find from the graph that the maximum velocity is around v=0.59m/s. The two values are very close, hence the models are probably accurate representations of what a physical pendulum of this specification would have as a maximum value.

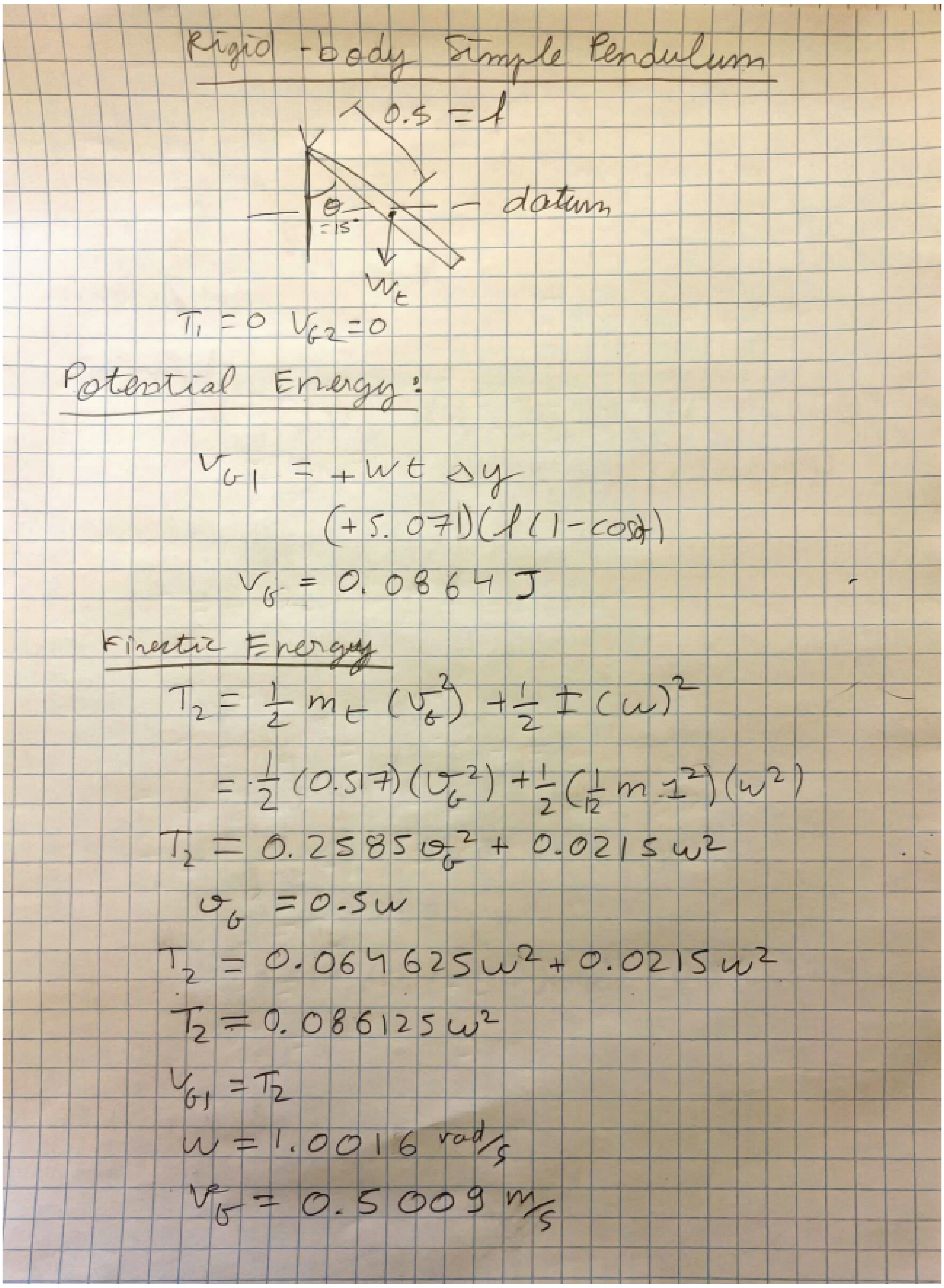

Figure 20: Simple Pendulum Rod Only Calculations

For the Rod only model, we once again have a maximum velocity much higher than what was found in section 4. From the calculations shown in Figure 20, we see that the max velocity found using the conservation of energy method is v=0.5009 m/s. However, this is for the max velocity at the COM (center of mass). We use this equation to find it for the velocity at the end of the rod:

Vend = v + W *r

r=0.5 m , w = 1.0016 rad/s , v = 0.5009 m/s => Vend = 1.0017 m/s

This gives us that Vend = 1.0017 m/s, when compared to the velocity in section 4, v = 1m/s, the values are almost the same as expected. As such, it is very probable that this model is accurate.

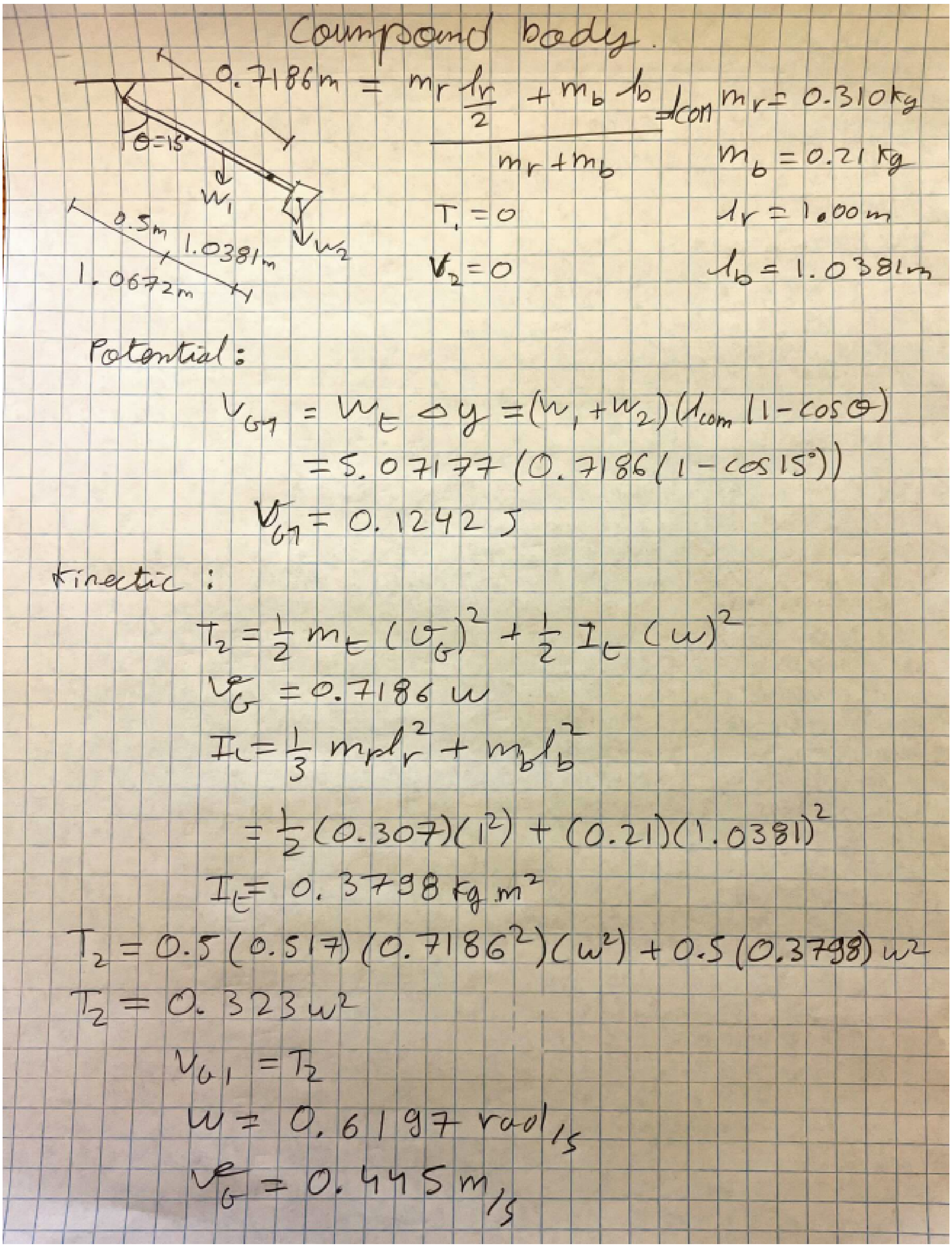

Figure 21: Compound Body Pendulum Calculations

Finally, the compound body pendulum max velocity found is v = 0.445m/s, the calculations can be seen in Figure 21. However, once again this is the velocity for the COM, we thus need to find the velocity at the bob to be able to compare with the matlab portion, we do so using this formula.

Vend = v + W *r

r=0.3195 m , w = 0.6197 rad/s , v = 0.445 m/s => Vend =0.643 m/s

When compared to the velocity found in Section 4, v = 0.9m/s, the difference is 0.257m/s, this is to be expected. This difference is probably from the model not taking into account certain parameters.

9.0 Conclusion

This paragraph will explore possible sources of errors that may have occurred during the experiment. For one, it is possible that the eye hook used to attach the pendulum to the stand was not completely in the middle of the rod, this would cause an imbalance in the pendulum that could affect the data gathered. Also, the stand might not have been level, thus causing the pendulum to have a bias towards one side when swinging. Another cause might be from the video recording that was captured during the experiment, due to the aspect ratio from the phone to the pendulum, it might seem that the pendulum is passing the 5 degrees mark sooner or later then it actually is. This would skew the time stamp found by analyzing the video and thus add a source of errors when superposing the data points to the friction graph in Section 6.5. While swinging it is possible that the pendulum was not swinging in a straight arc, possibly caused by the type of attachment point between the stand and the pendulum, this would also influence the measured data. Since we had to do 2 experiments, one to find the period, one where we recorded the video and measured the rest of the required data. The period from the experiment might not necessarily match the period from the experiment with the video. This would add an extra source of error thus making the friction graph be even more out of sync with the data points. Also, we used the delta = sin(delta) approximation, this over time adds a source of errors, this would also cause the friction graph to be out of sync with the data points. Finally, air resistance, we did not account for it in any of the models and assumed that it was zero, this is of course not the case and it introduces another source of errors.

Appendix A

The command window outputs of the three symbolic derivations of angular frequency for each pendulum model is provided in this section. The outputs shown are the expressions for the simplified harmonic closed-loop solutions for angular velocity as a function of time, of which angular frequency can be isolated. Note that the third capture of the command output for the compound rigid body pendulum model was copied into a different application to improve overall readability.

```
Command Window

QSol(t) =

(Q0*exp(-(t*(-L*g)^(1/2))/L)*(exp((2*t*(-L*g)^(1/2))/L) + 1))/2


QSol(t) =

Q0*cos((g^(1/2)*t)/L^(1/2))
```

Figure 22: MATLAB Command Window Output for Pendulum Bob Rotational Motion Equation

```
Command Window

QSol(t) =

(Q0*exp(-(6^(1/2)*t*(-L*g)^(1/2))/(2*L))*(exp((6^(1/2)*t*(-L*g)^(1/2))/L) + 1))/2


QSol(t) =

Q0*cos((6^(1/2)*g^(1/2)*t)/(2*L^(1/2)))
```

Figure 23: MATLAB Command Window Output for Simple Rigid Body Pendulum Rotational Motion Equation

```
QSol(t) =

(Q0*exp(-(6^(1/2)*t*(-L*g*(m_arm + 2*m_bob)*(m_arm + 3*m_bob))^(1/2))/(2*(L*m_arm +
3*L*m_bob)))*(exp((6^(1/2)*t*(-L*g*(m_arm + 2*m_bob)*(m_arm + 3*m_bob))^(1/2))/(L*m_arm +
3*L*m_bob)) + 1))/2


QSol(t) =

Q0*cos((6^(1/2)*g^(1/2)*t*(m_arm + 2*m_bob)^(1/2)*(m_arm + 3*m_bob)^(1/2))/(L^(1/2)*(2*m_arm
+ 6*m_bob)))
```

Figure 24: MATLAB Command Window Output for Compound Rigid Body Pendulum Rotational Motion Equation

Appendix B

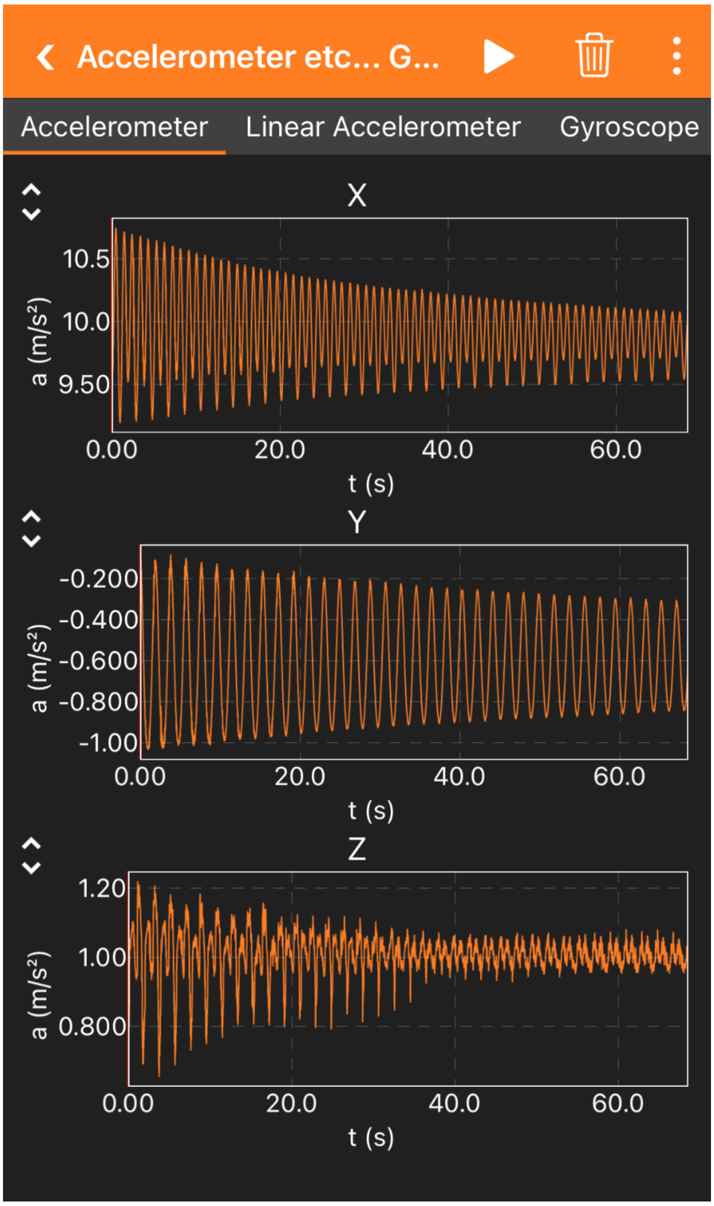

Figure 25 : Accelerometer Graphs

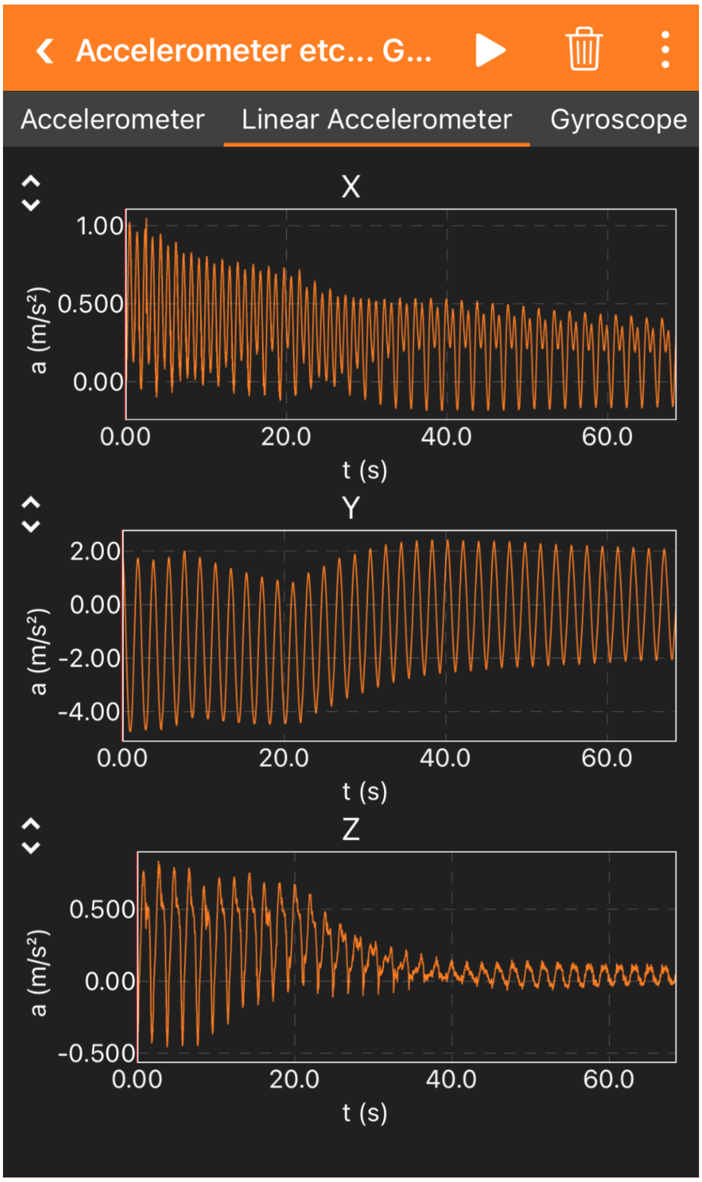

Figure 26: Graphs from the Linear Accelerometer

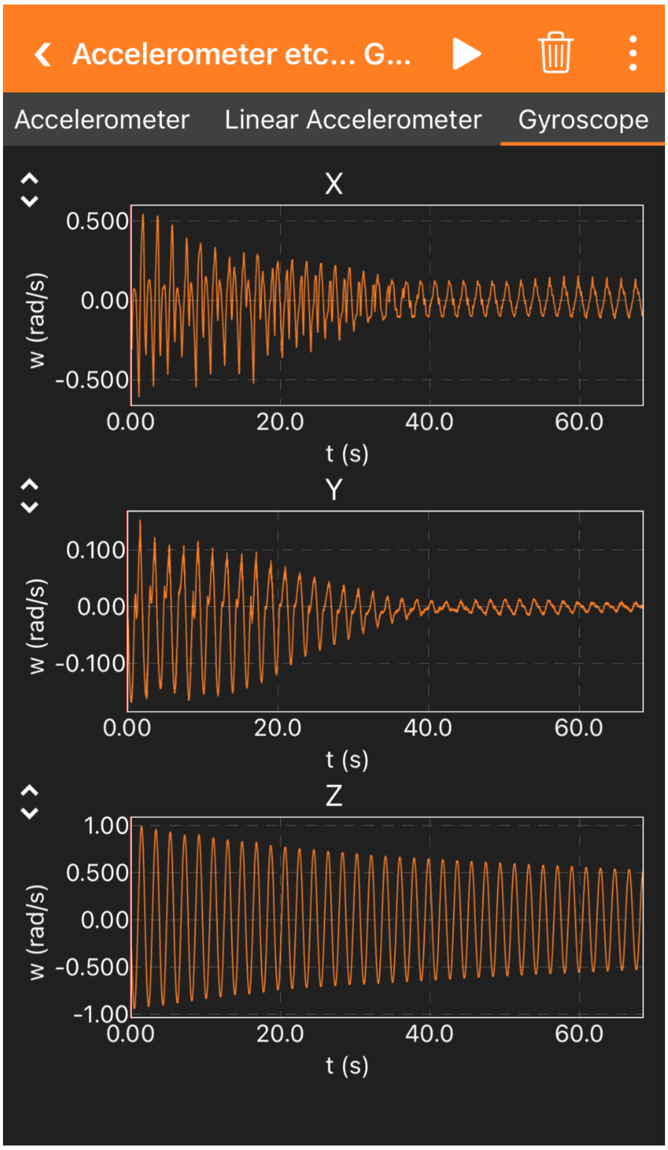

Figure 27: Graphs from the Gyroscope

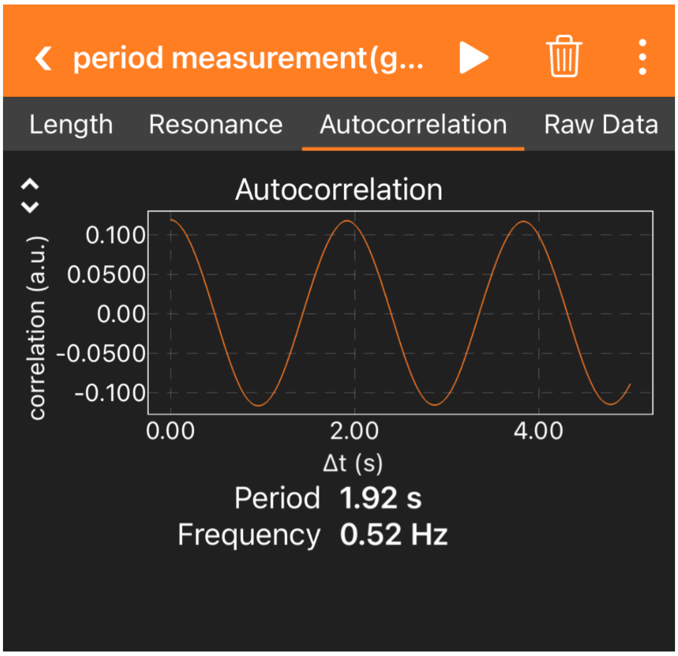

Figure 28: Graphs from the Pendulum Experiment for Finding the Periods