Two-Axis Inverted Pendulum System

Final Project Report

Dec. 11, 2022

Erich Deetlefs

Contents

1. Introduction

Over the years, inverted pendulums have been thoroughly studied and solved by various authors through different stabilizing techniques, including reaction wheels or a moving cart. This report will consider the mathematical models of an inverted pendulum controlled via reaction. This system is very similar to those seen in spacecraft, where the torque from the reaction wheels is employed to stabilize the pendulum on multiple axes.

Reaction wheels, often known as flywheel systems, are one type of actuator that may be used to control the body in the applied axis. Three or more reaction wheels can be used to stabilize the satellite, which acts as a pendulum in the system. This stabilization of an inverted pendulum with reaction wheels demonstrates the application of Newton's third law of motion, which states "that for every action (force) in nature, there is an equal and opposite reaction. If object A exerts a force on object B, object B also exerts an equal and opposite force on object A. In other words, forces result from interactions." [1] This makes reaction wheels a critical choice in the design of control for satellites and various space-related devices. This report aims to understand how such devices control and interact with such systems.

For the mathematical models of the inverted pendulum system, the derivation of the state space model was first created from equations of motion that were made by linearizing the found Lagrange equations. These models could generate all the system's dynamics, including an analogous circuit, linear graph, and a bond graph. Once created, the system was simulated in MATLAB and Simulink, where various case studies were tested.

To further explore the concept of an inverted pendulum system, a PID controller was added to show how such a system would be stabilized based on input from the reaction wheels. This PID controller creates a closed-loop feedback system stabilizing the pendulum to 0°. As an extra step, the entire system was also studied with outside forces in the form of impulse step functions, which acted as outside disturbance the pendulum system would undergo.

2. Project Description

This project aims to model the movement and oscillation of an inverted pendulum in both the x and y axes with a reaction wheel connected to it. Traditionally, these systems are controlled via a moving base; however, in this project's scope, the system's control would originate from the inertia created by the reaction wheels. This project also allows for real-world applications, primarily aerospace-related applications, specifically in satellites. Using reaction wheels in satellites allows the system to be indefinitely controlled as long as it has a power source, without the need for propellants. Inverted pendulums, as previously stated, are classic dynamics projects which aim to balance a rod vertically on a single point of contact with 2 DOF [2]. This project also allows for the control of a two-axis inverted pendulum through the use of Simulink and MATLAB. Many previous authors have controlled such a system with different models, including PID (Propotional-Intergral-Dertivative) controllers and LQR (Linear-Quadratic Regulator) [3]. Through the finding in this report, the aim is to understand the working principles of the inverted pendulum and reaction wheel system and the process of controlling such an unstable system.

3. Modeling

For the modelling of this section, the mathematical equations will be used to find the Lagrange of the system. The equations of motion will also be derived alongside the states-space model of the linearized system. When solving for the equations, one assumption must first be made, and it consists of neglecting the forces of friction in both the reactions wheel and pendulum. Doing this simplifies the overall mathematical model of the Inverted Pendulum Reaction Wheel system. We can also assume that both the X and Y axis will have the same mathematical models. Therefore, we only need to derive the model for a single axis with only an additional moment of inertia [4].

In order to derive the equations of motion, we must first find the kinetic (T) and potential (U) energies of the Inverted Pendulum system. These can be calculated with the following equations:

$$KE = T = \frac{1}{2}(m)v^2 + \frac{1}{2}(I)\omega^2 \tag{1}$$

$$PE = U = mgcos(\theta) \tag{2}$$

To fill these equations, we can denote the angle θ to be a vector system that contains both the angle of the pendulum and the angle of rotation of the reaction wheel, written as $\theta = (\theta_p, \theta_w)^T$. We can also create a constant inertia matrix derived from the parallel axis theorem [5]. This theorem gives the relation between a body's moment of inertia about its centre of mass and its moment of inertia about an arbitrary axis parallel to the centre of mass, stated by the following equation

$$I = I_0 + mb^2 \tag{3}$$

Where I is the moment of inertia about a parallel axis at a distance b from the centre of mass. Using this theorem, we can derive the inertia matrix as follows:

$$M = \begin{bmatrix} m_p l_p^2 + m_w l_w^2 + I_p + I_w & I_w \\ I_w & I_w \end{bmatrix} \tag{4}$$

We can then solve for the T and U:

$$T = \frac{1}{2}(\dot{\theta}^T)M\dot{\theta} \tag{5}$$

$$U = (m_p l_p + m_w l_w)gcos(\theta) \tag{6}$$

With these equations, we can then calculate the Lagrange of the system:

$$L(\theta, \dot{\theta}) = \frac{1}{2}(\dot{\theta}^T)M\dot{\theta} + (m_p l_p + m_w l_w)gcos(\theta) \tag{7}$$

For the equations of motion, we can use the derived equation below. Since we have two sets of coordinates, θ_p and θ_w, and only a single actuator torque that works on the reaction wheel. By setting the values of $\tau_1 = 0$ and $\tau_2 = \tau$. We can calculate the equations of motion

$$\frac{d}{dt}\left(\frac{\delta L}{\delta \dot{q_i}}\right) - \frac{\delta L}{\delta q_i} = \tau_i \tag{8}$$

$$\frac{d}{dt}\left(\frac{\delta L}{\delta \dot{\theta_p}}\right) - \frac{\delta L}{\delta \theta_p} = 0 \tag{9}$$

$$\frac{d}{dt}\left(\frac{\delta L}{\delta \dot{\theta_w}}\right) - \frac{\delta L}{\delta \theta_w} = \tau \tag{10}$$

We can input the previously found equation for the Lagrange function to solve these equations. Placing this into the above equation gives us the following

$$\left(m_p l_p^2 + m_w l_w^2 + I_p + I_w\right)\ddot{\theta_p} + I_p\ddot{\theta_w} = \left(m_p l_p + m_w l_w\right)gsin(\theta_p) \tag{11}$$

$$I_p\ddot{\theta_p} + I_p\ddot{\theta_w} = \tau \tag{12}$$

We can then combine these two equations, solving for $\ddot{\theta_p}$ and $\ddot{\theta_w}$ respectively and finding the following two equations:

$$\ddot{\theta_p} = \frac{I_w\left(m_p l_p + m_w l_w\right)gsin(\theta_p)}{(m_p l_p^2 + m_w l_w^2 + I_p + I_w)I_w - I_w^2} - \frac{I_w\tau}{(m_p l_p^2 + m_w l_w^2 + I_p + I_w)I_w - I_w^2} \tag{13}$$

$$\ddot{\theta_w} = \frac{I_p\left(m_p l_p + m_w l_w\right)gsin(\theta_p)}{(m_p l_p^2 + m_w l_w^2 + I_p + I_w)I_w - I_w^2} + \frac{(m_p l_p^2 + m_w l_w^2 + I_p + I_w)\tau}{(m_p l_p^2 + m_w l_w^2 + I_p + I_w)I_w - I_w^2} \tag{14}$$

With these two equations of motion, we can create a State-Space model. We must first linearize them about the equilibrium point, which we do by setting $\theta_p = 0$ and substituting $\tau = K_m u$, where K_m is the motor torque constant and u is the control system. With these, we get the following state-space model

$$\dot{x} = \begin{bmatrix} 0 & 1 & 0 \\ \frac{I_p\left(m_p l_p + m_w l_w\right)g}{l_p^2 I_w m_p + l_w^2 I_w m_p + I_w I_p} & 0 & 0 \\ -\frac{I_p\left(m_p l_p + m_w l_w\right)g}{l_p^2 I_w m_p + l_w^2 I_w m_p + I_w I_p} & 0 & 0 \end{bmatrix} \begin{bmatrix} \theta_p \\ \dot{\theta_p} \\ \dot{\theta_w} \end{bmatrix} + \begin{bmatrix} 0 \\ \frac{-I_p(K_m)}{l_p^2 I_w m_p + l_w^2 I_w m_p + I_w I_p} \\ \frac{(m_p l_p^2 + m_w l_w^2 + I_p + I_w)K_m}{l_p^2 I_w m_p + l_w^2 I_w m_p + I_w I_p} \end{bmatrix}[u] \tag{15}$$

$$y = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \theta_p \\ \dot{\theta_p} \\ \dot{\theta_w} \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \end{bmatrix}[u] \tag{16}$$

Since the state space model was first found, we can use MATLAB to solve for the transfer function.

$$\frac{\theta_p}{U(S)} = \frac{(I_p s(m_p l_p^2 + m_w l_w^2 + I_w))}{((I_p^2 - I_p(m_p l_p^2 + m_w l_w^2 + I_p + I_w))(m_p l_p^2 s^2 - g m_p l_p + m_w l_w^2 s^2 - g m_w l_w + I_w s^2)} \tag{17}$$

$$\frac{\theta_w}{U(S)} = \frac{-\left(m_p l_p^2 + m_w l_w^2 + I_p + I_w\right)}{\left(s\left(I_p^2 - I_p\left(m_p l_p^2 + m_w l_w^2 + I_p + I_w\right)\right)\right)} - \frac{(I_p g(I_p m_p + I_w m_w))}{(s(I_p^2 - I_p(m_p l_p^2 + m_w l_w^2 + I_p + I_w))(m_p l_p^2 s^2 - g m_p l_p + m_w l_w^2 s^2 - g m_w l_w + I_w s^2)} \tag{18}$$

4. Simulations

The simulations for a 2-axis inverted pendulum were carried out using MATLAB and Simulink. The state space model of the system was first built using the various arithmetic functions in Simulink. The overall model of this is seen in Figure 4.1.

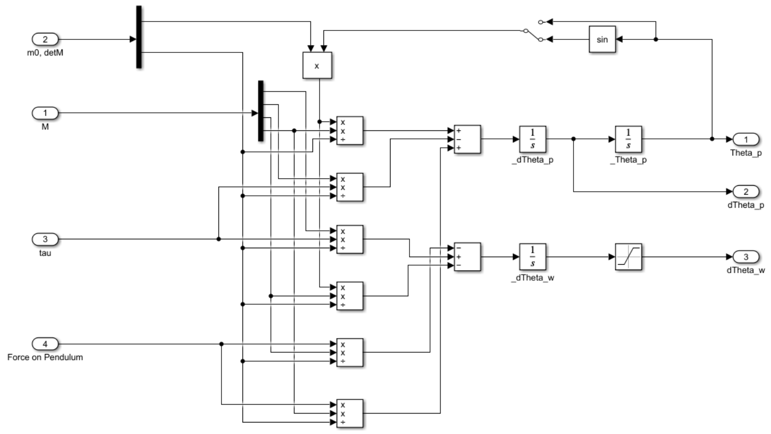

Figure 4.1 – Equations of Motion in Simulink

Since this entire system must be duplicated for the second axis the mathematical model can be organized into a subsystem. With this subsystem now in place the rest of the Two-Axis Reaction Wheel Inverted Pendulum System can be built. In Figure 4.2 we can see the all the input parameters now being sent to the previously created subsystem.

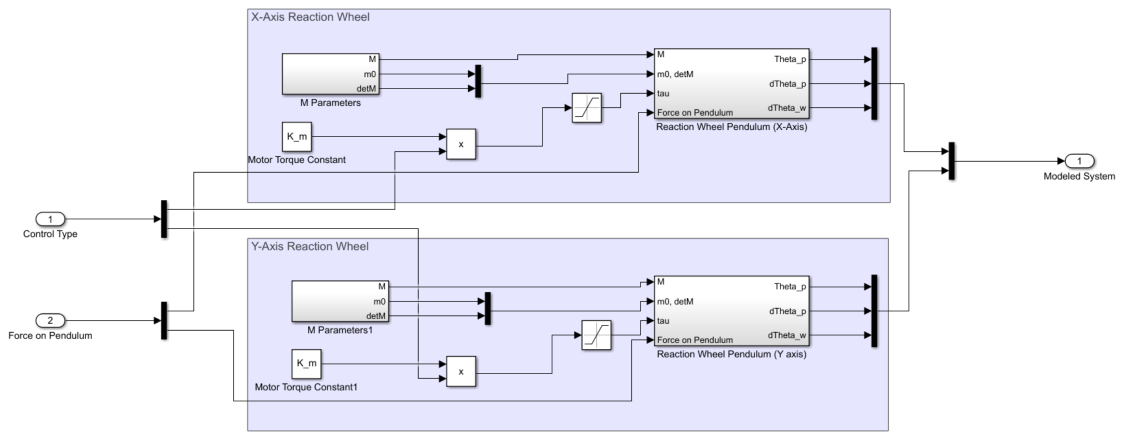

Figure 4.2 – The State Space model representation in Simulink

Since this is a two-axis system both sections need to be duplicate for their modeling of the X and Y axis. The above model is all that is needed to model an uncontrolled system however since we want this system to be controlled with PID we can create another subsystem that contains the finished modelling of the inverted pendulum system. In figure 4.3 the finished system is seen and organized into several different parts.

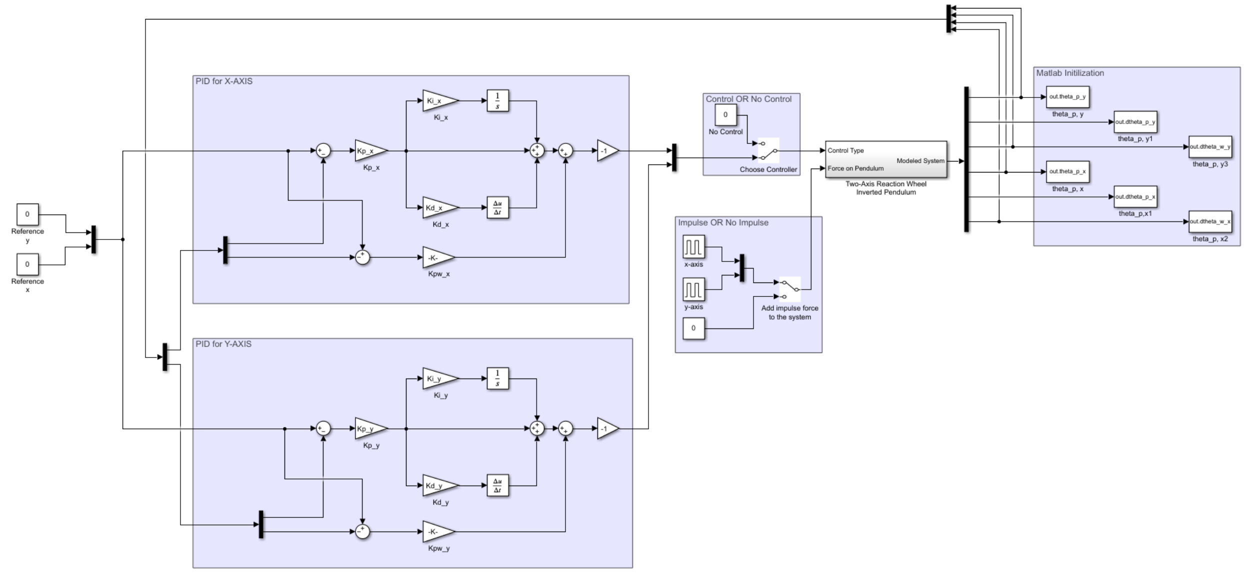

Figure 4.3 – Full Simulink model including the PID control system

To start on the left side, we find two PID blocks one for each axis, with their respective parameters found in Table 4.2. A manual switch was added into the system to allow for PID to be manually turned of and the simulation of the system to be examined separate from it. To further the complexity of the model an impulse system was also added to act as an outside force pushing on the pendulum. This function can also be turned off. Since we also want to visualize the output of this system the model sends the data back to MATLAB where it can be plotted. The code for which is found in Appendix A.

Overall, the system can switch between 3 different cases, as described below. The model also simulates the control of the system which is done by two reaction wheels that work independently of one another. This control is made possible by using PID. The initial x and y axis angle at which the pendulum was placed were -10° and 14° respectively.

Parameter	Value
l_w	0.330 [m]
l_p	0.185 [m]
m_w	0.588[kg]
m_p	0.033[kg]
I_w	2.662E-4 [kg m^2]
I_p	3.765E-4 [kg m^2]
g	9.81 [m/s^2]
K_m	5.5E-3 [Nm/$\sqrt{w}$]

Table 4.1 – Initial set parameters obtained from thesis paper [4].

Case 1: No PID Control:

No control was used in the initial simulation. This was done to demonstrate how the model will oscillate around the stable downright equilibrium while starting with an offset from the unstable upright equilibrium.

Of course, this is not possible with a physical model since we cannot have an offset of more than 180°. For the case of this model, friction was not modeled, as to simplify the process. Doing so would allow the pendulum to swing at a constant amplitude as show in figure 4.4. For the results of this simulation the outcome was expected. The model as seen in figure 4.4 and 4.5 shows the pendulum swinging about the initially set values. If the pendulum was placed at an offset on only one of the axes, it would be expected to oscillate on only a single axis as seen in figure 4.6 and 4.7. The modeling for this case is as expected and lines up with the mathematical models previously stated. To note due to the absence of frictions working on the system it would not converge towards a stable equilibrium unless added.

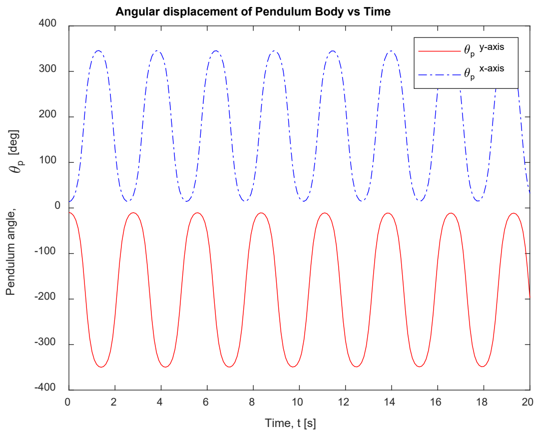

Figure 4.4 – Oscillation of Inverted Pendulum on two-axis offset (NO Control)

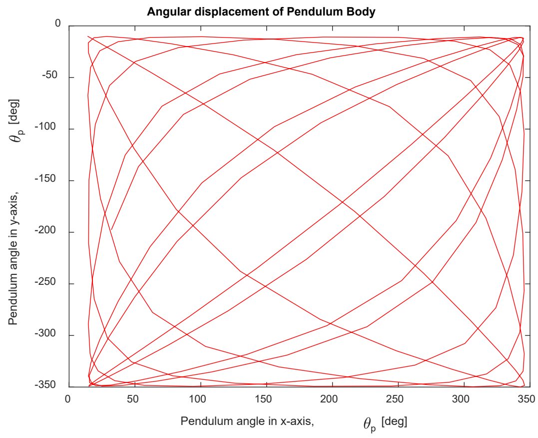

Figure 4.5 – Oscillation of Inverted Pendulum on two-axis offset (Top-Down View, NO Control)

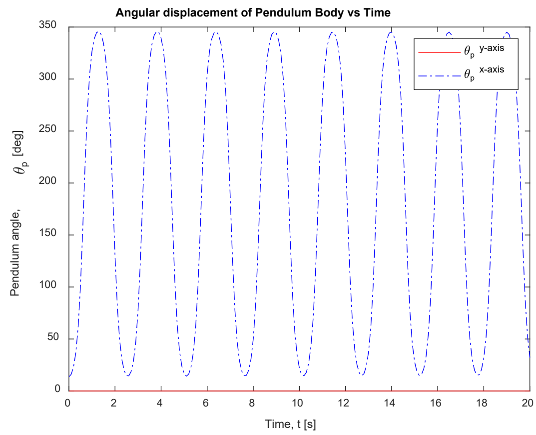

Figure 4.6 – Oscillation of Inverted Pendulum on single axis Offset (NO Control)

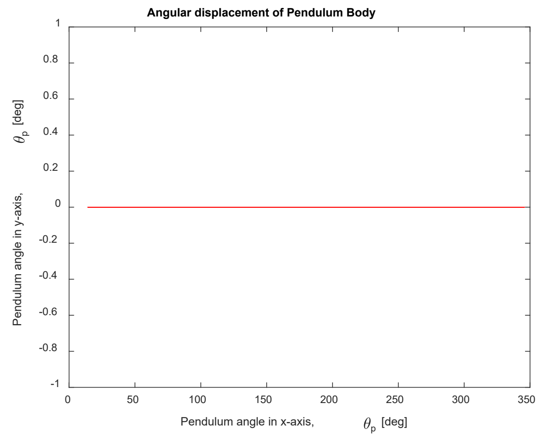

Figure 4.7 – Oscillation of Inverted Pendulum on single axis offset (Top-Down View, NO Control)

Case 2: PID Control on the angle of the Pendulum:

In this case the pendulum is controlled with a PID controller which is described as the following formula:

$$u(t) = K_p(e(t) + \frac{1}{T_i}\int_0^\infty e(\tau)d\tau + T_d\frac{de(t)}{dt}$$

Where the K_p, T_i, and T_d are the proportional, integral, and derivative gains respectively. The proportional gain penalizes the current error, e(t), while the integral and derivative gains penalize the past error and the possible future error. These constants can be calculated, or they can be solved using the Ziegler-Nichols method. With the Ziegler-Nichols method the value of T_i is set to infinity while the value to T_d is set to 0. K_p is the slowly increased until the system output becomes stable with constant oscillations, T_k. This K_p value is then considered as the critical gain K_{pk}. With this the PID gains can be calculated as follows:

$$K_p = 0.6K_{pk}$$

$$T_i = 0.5T_k$$

$$T_d = 0.75T_k$$

Using these formulas, the values for the PID gains were found to be

Controller Gain	Value
Proportional, K_p	1200
Integral, T_i	0.5
Derivative, T_d	0.75

Table 4.2 – Gain values for PID control

Using the values in Table 4.2 for the PID control, we can then control the system in case 1. In figure 4.7 we can see how the PID controller stabilizes the systems on each of the axes. Figure 4.8 shows the state trajectory in the on the X and Y axis plane and gives a top-down view of how the pendulum will move in a 3D space.

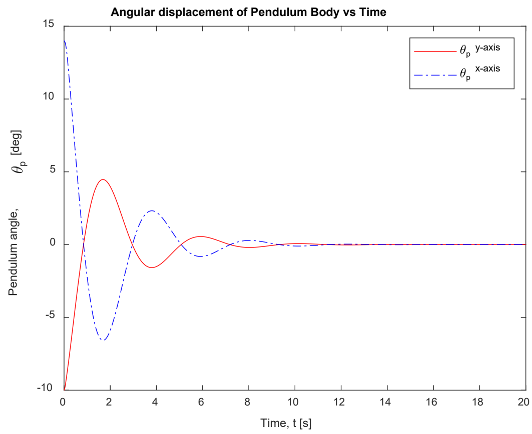

Figure 4.7 – Oscillation of Inverted Pendulum on two-axis offset (Control)

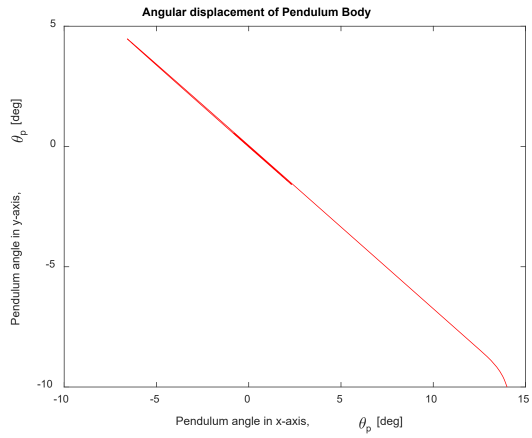

Figure 4.8 – Oscillation of Inverted Pendulum on two-axis offset (Top-Down View, Control)

Since we also include the motor constant (K_m) in our mathematical model we are able to plot the angular velocity of the reactions wheels as well as the angular velocity of the pendulum respective to the wheels. Figures 4.9 shows how the reactions wheels ramp up speed to try to stabilize the system and Figure 4.10 shows the response of the movement of the pendulum. In reality these results would be much noisier due to various vibrations that the system would feel when moving.

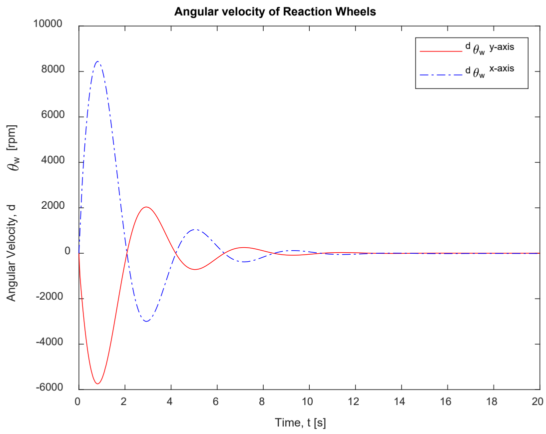

Figure 4.9 – Angular Velocity of the Reaction wheels (Control)

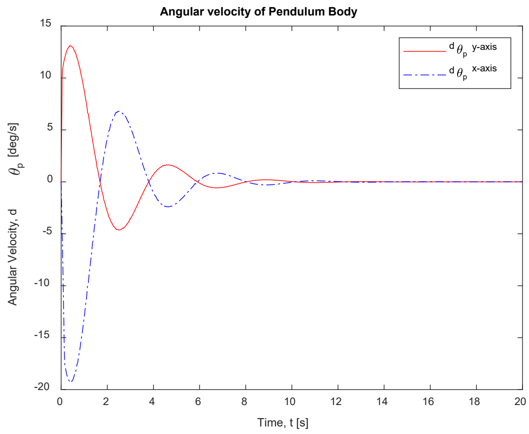

Figure 4.10 – Angular Velocity of the Pendulum Body (Control)

Case 3: PID Control with outside impulses:

For this case study a small amount of impulse force was applied to the pendulum body, which is used to show how the PID controller would react with disturbances. This impulse force acts as an outside force that pushes the pendulum, which in turn rapidly changes $\theta_{p,x}$ and $\theta_{p,y}$. Each axis receives a step function impulse with an amplitude of 0.7 for at 10 seconds. Since this is not a constant motion the width of the impulse remains small, only being 0.5% of the period. The single pulse is show in Figure 4.11.

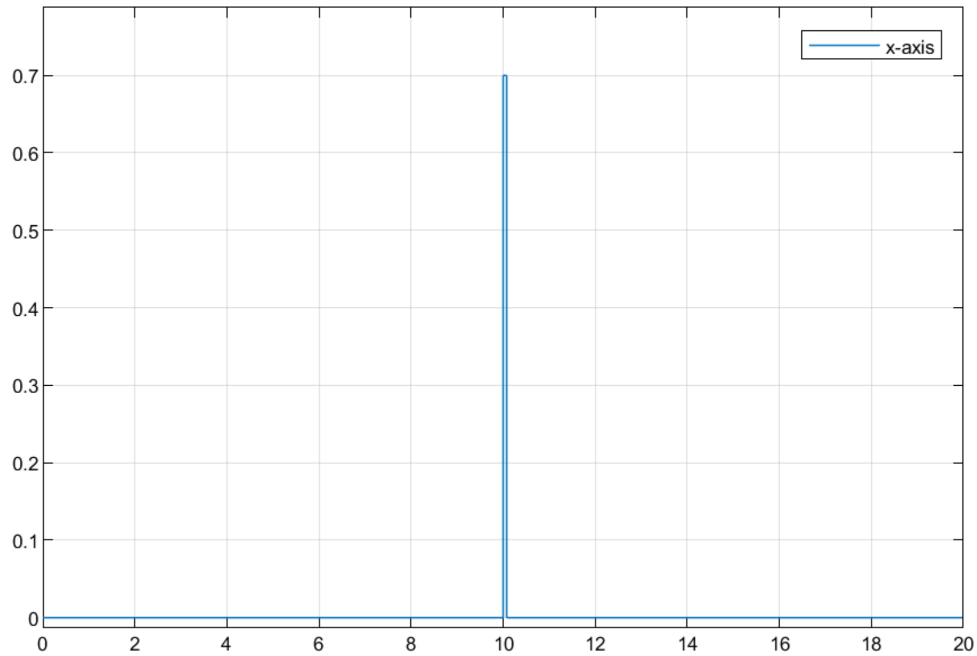

Figure 4.11 – Impulse Response for Force on Pendulum

The results for this are show in the following figures. Figure 4.12 shows how the previously stable PID controlled system is disturbed, before converging back to equilibrium. Figure 4.13 once again shows a top-down view that allows for a better visualization of the disturbance input.

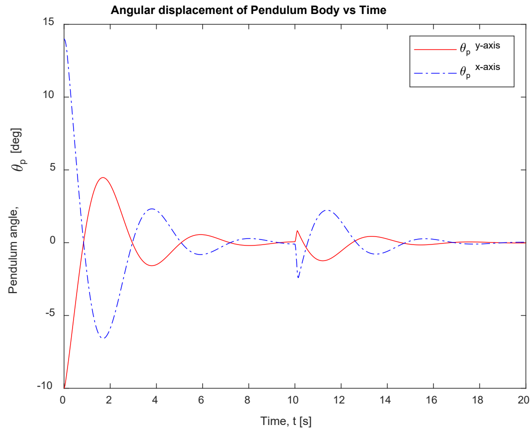

Figure 4.12 – Oscillation of Inverted Pendulum on two-axis offset (Control, Impulse)

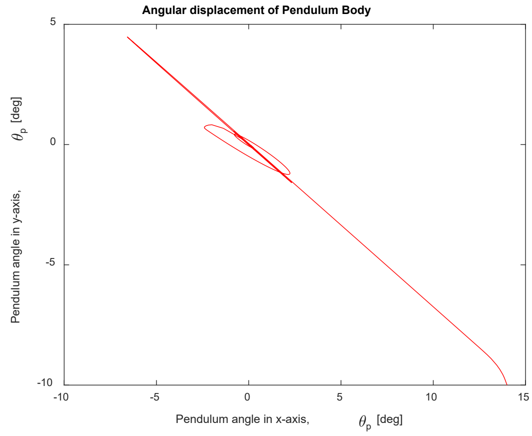

Figure 4.13 – Oscillation of Inverted Pendulum on two-axis offset (Top-Down View, Control, Impulse)

As in the previous case we are able to analyze the angular velocities from both the wheels and the pendulum. However, unlike the previous case when analyzing this motion, we can see how the impulse affects the system and how the system reacts to it when applied. Figures 4.14 and 4.15 show this motion of the system. If we were to make the size of the impulse larger the pendulum system would fall over as expected.

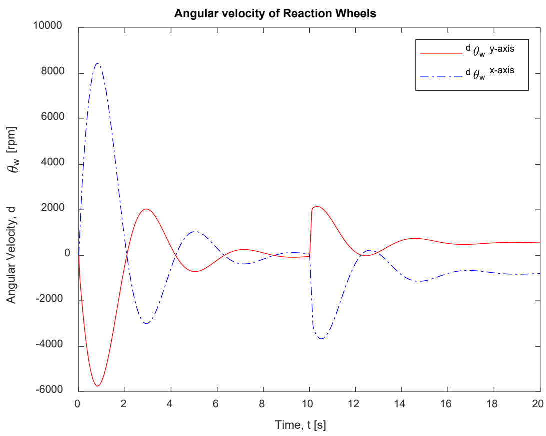

Figure 4.14 – Angular Velocity of the Reaction wheels (Control, Impulse)

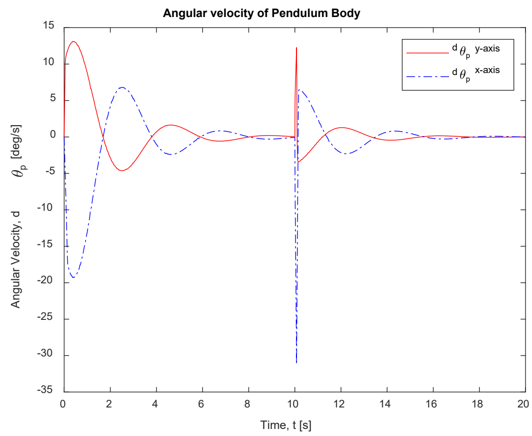

Figure 4.15 – Angular Velocity of the Pendulum Body (Control, Impulse)

Another interesting thing to note is that when further analyzing the plot data of the reaction wheels angular velocity, we can see that they begin to diverge from zero. This does not mean that the system is unstable, but that the reaction wheels stabilize at different non-zero values. In reality this would be a design flaw since the motor would wear down over time due to unwanted rotation. With further studying and a more complex control system however this problem can easily be fixed.

5. Conclusion

In this project report, we considered a unique dynamic system consisting of an inverted pendulum controlled by two reaction wheels. Throughout the report, we develop the mathematical models of such a system to further our knowledge of how such an unstable system works and how it can be controlled. The control systems of satellites are an important subsystem used daily. Understanding how the reaction wheels of such systems work, with respect to the inertial fixed frame, are important aspects of engineering.

To start this project, we first had to solve the mathematical model of the system. Since we assumed that both the X and Y axis have very similar models, we were able to solve for only a single axis. This was done by first finding the entire system's total potential and kinetic energies by lumping the reaction wheel and pendulum into a single system. To solve for the potential and kinetic energies, we had to find the moments of inertia using the Parallel Axis Theorem. This theorem gives the relation between a body's moment of inertia about its centre of mass and its moment of inertia about an arbitrary axis parallel to the centre of mass. With the system's total PE and KE, we were able to place them into the Lagrange formula, which would then be linearized, giving us the equations of motion. With these, we were able to solve the State Space matrix of the system.

Once all the mathematical models were developed and the state space model had been obtained, we could find an Analogous System, Linear Graph and Bond Graph, each of which confirmed our previous findings. Due to the nature of the Inverse Pendulum system, the Analogous system was very simple, and we could represent both parts as two different electrical circuits, one containing the contents of the reaction wheel and the other containing those of the pendulum body. Similarly, this system's linear and bond graphs were also very simplistic.

After all this, the simulation for the two-axis inverted pendulum system was built in Simulink and MATLAB. The initial Simulink model took the equations of motion and plotted them based on present parameters. Doing so created a displacement plot of the pendulum's angular displacement. We had assumed that friction was negligible, which simplified our model and meant that it would never reach an equilibrium stopping point. As an additional step to the project, we included using PID to control the system so that it would stay upright.

Overall, this project helped better our understanding of a Two-Axis Inverted Pendulum system and furthered our knowledge of how reaction wheels work to control them. While not in the scope of this course, implementing a PID control system helped verify our models. With the findings in this report, we would be able to create a practical model to further our tests and appreciation of such systems.

6. References

[1] "Newton's laws of motion - Glenn Research Center," NASA, 27-Oct-2022. [Online]. Available: https://www1.grc.nasa.gov/beginners-guide-to-aeronautics/newtons-laws-of-motion/. [Accessed: 11-Dec-2022].

[2] D. Collins, "What are degrees of freedom (DOF) and how many are there?," Linear Motion Tips, 05-Oct-2021. [Online]. Available: https://www.linearmotiontips.com/motion-basics-what-are-degrees-of-freedom-dof-how-many-are-there/. [Accessed: 11-Dec-2022].

[3] Michel Levis Senior Application Engineer | Quanser Inc., M. Levis, and Senior Application Engineer | Quanser Inc., "Common and not so common pendulum configurations," *Quanser*, 11-Feb-2020. [Online]. Available: https://www.quanser.com/blog/common-and-not-so-common-pendulum-configurations/#:~:text=Similarly%20to%20the%20linear%20pendulum,the%20pendulum%20at%20the%20end. [Accessed: 11-Dec-2022].

[4] U. Tugcular, B. Daşer, and M. Ucar, "Control of an Inverted Pendulum with a Reaction Wheel," thesis, 2019.

[5] A. Türkmen, M. Y. Korkut, M. Erdem, Ö. Gönül and V. Sezer, "Design, implementation and control of dual axis self balancing inverted pendulum using reaction wheels," 2017 10th International Conference on Electrical and Electronics Engineering (ELECO), 2017, pp. 717-721.

[6] C. W. de Silva, *Mechatronics an integrated approach*. Boca Raton etc.: CRC Press, 2005.

[7] "Self Balancing Stick - Dual Axis Reaction Wheel Inverted Pendulum," *Free CAD Designs, Files & 3D Models | The GrabCAD Community Library*, 28-Feb-2016. [Online]. Available: https://grabcad.com/library/self-balancing-stick-dual-axis-reaction-wheel-inverted-pendulum-1. [Accessed: 11-Dec-2022].

7. Appendix A

```
%% Initialize system parameters

%*********************************
% Note: This code makes use of symbolic parameters in order to simplify it
% for the user. The report contains the full unsimplified calculations.
%********************************
clc;

l_w = 0.33;
l_p = 0.185;
m_p = 0.033;
I_p = (m_p*l_p^2)/3;
m_w = 0.088+0.5;
I_w = 0.088*0.055^2;

g0 = 9.81;
m0 = (m_p*l_p + m_w*l_w)*g0;
M_11 = m_p*l_p^2 + m_w*l_w^2 + I_p + I_w;
M_12 = I_w;
M_21 = I_w;
M_22 = I_w;

detM = M_11*M_22 - M_12*M_21; %For the TF

K_m = 5.5e-3;
T_max = 0.6;

%% PID Controller
Kp_x = 1200;%2000;
Ki_x = 0.5;%0.5;
Kd_x = 0.75;
Kpw_x = 0.2;

Kp_y = 1200;%2000;
Ki_y = 0.5;%0.5;
Kd_y = 0.75;
Kpw_y = 0.2;

%% State-Space Equations
A = [0 1 0; (M_22*m0)/detM 0 0; -(M_21*m0)/detM 0 0];
B = [0; -M_12/detM; M_11/detM];
C = [0 1 0; 0 0 1];
D = 0;

%% Initial Conditions
angleX = -10;
angleY = 14;

X_intial = (angleX)*(pi/180);
Y_intial = (angleY)*(pi/180);
x2_0 = 0;
x3_0 = 0;

%% Plot Simulations
```

```
figure('Name', 'Theta_p x');
plot(out.theta_p_x.signals.values(:,1)*180/pi, out.theta_p_y.signals.values(:,1)*180/pi,'r-');
title('Angular displacement of Pendulum Body');
xlabel('Pendulum angle in x-axis, \theta_p [deg]');
ylabel('Pendulum angle in y-axis, \theta_p [deg]');

figure('Name', 'dTheta_w');
plot(out.dtheta_w_y.time, out.dtheta_w_y.signals.values(:,1)*30/pi,'r-', out.dtheta_w_x.time,
out.dtheta_w_x.signals.values(:,1)*30/pi, 'b-.');
title('Angular velocity of Reaction Wheels');
xlabel('Time, t [s]');
ylabel('Angular Velocity, d\theta_w [rpm]');
legend('d\theta_w y-axis','d\theta_w x-axis');

figure('Name', 'dTheta_p');
plot(out.dtheta_p_y.time, out.dtheta_p_y.signals.values(:,1)*180/pi,'r-', out.dtheta_p_x.time,
out.dtheta_p_x.signals.values(:,1)*180/pi, 'b-.');
title('Angular velocity of Pendulum Body');
xlabel('Time, t [s]');
ylabel('Angular Velocity, d\theta_p [deg/s]');
legend('d\theta_p y-axis','d\theta_p x-axis');

figure('Name', 'Theta_p');
plot(out.theta_p_y.time, out.theta_p_y.signals.values(:,1)*180/pi,'r-', out.theta_p_x.time,
out.theta_p_x.signals.values(:,1)*180/pi, 'b-.');
title('Angular displacement of Pendulum Body vs Time');
xlabel('Time, t [s]');
ylabel('Pendulum angle, \theta_p [deg]');
legend('\theta_p y-axis','\theta_p x-axis');
```