Ifraah Beg, Greg Linkowski and Wyatt McAllister

ECE 486: Final Project Report

Fall 2015

TA: Yinai Fan

Lab Section: AB4 (Thursday 12:00pm)

Introduction

In this lab we designed a series of controllers of increasing complexity to effectively control the Reaction Wheel Pendulum (RWP). This device consists of a weight and motor mounted at the end of a pendulum affixed to the body of the device through the pivot point. We are able to actuate this system via the DC motor mounted to the end of the pendulum. The spin of the motor generates a torque on the pendulum which allows us to control the pendulum angle. Our control system aims first to stabilize the pendulum about the unstable equilibrium point where the pendulum stands vertically with the weight suspended in midair and also later about the stable equilibrium where the weight hangs downwards under the force of gravity. Our system contains two high quality optical encoders which allow us to measure the angles between the pivot point and the pendulum and between the pendulum and the rotor, φ_p and φ_r respectively. Ultimately, we will design a controller which swings the pendulum up to the unstable equilibrium by destabilizing the system about the stable equilibrium point and implementing switching control to stabilize our system about the unstable equilibrium once the system reaches the desired position.

Mathematical Model (_/10)
Derivation of differential equations from Lagrangian (_/5pts)

The following system definitions are derived in the lab manual.

$J = J_p + m_p l_p{}^2 + m_r l_r{}^2$ $\quad m = m_p + m_r$ $\quad ml = m_p l_p + m_r l_r$

The system diagram shown below allows us to find the height of interest in calculating the potential energy for our pendulum

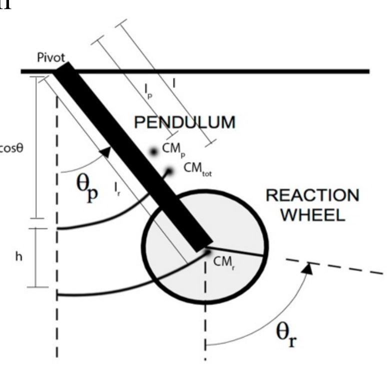


Figure 2: Schematic Diagram

Given the pendulum system, the kinetic energy is found to be one half the effective moment of inertia times the angular velocity.

$$KE_p = \frac{1}{2} J \omega_p{}^2 = \frac{1}{2}\left(J_p + m_p l_p{}^2 + m_r l_r\right)\dot{\theta}_p^2$$

The potential energy is then found to be the effective mass times g times the effective height for the potential energy calculation. This height is the vertical height between the center of mass of the rotor and the center of mass of the whole system. The potential energy can then be calculated as follows.

$$PE_p = mgh = mg\left(l_r - l\cos\theta_p\right) \qquad PE_p = mg\left(l_r - \frac{\left(m_p l_p + m_r l_r\right)}{m}\cos\theta_p\right)$$

$$PE_p = g\left(\left(m_p + m_r\right)l_r - \left(m_p l_p + m_r l_r\right)\cos\theta_p\right)$$

The kinetic and potential energy of the rotor are found in the same manner. Since the rotor's kinetic energy is independent of the kinetic energy of the pendulum, the moment of inertia of the rotor alone can be used. Since the rotor is symmetric about the origin point, its effective length is zero and it therefore has zero potential energy.

$$KE_r = \frac{1}{2}J_r\omega_r^2 = \frac{1}{2}J_r\dot{\theta}_r^2 \qquad PE_r = 0$$

The Lagrangian for the pendulum is the kinetic minus the potential energy.

$$PE_p = mgh = mg\left(l_r - l\cos\theta_p\right) \qquad PE_p = mg\left(l_r - \frac{\left(m_p l_p + m_r l_r\right)}{m}\cos\theta_p\right)$$

$$PE_p = g\left(\left(m_p + m_r\right)l_r - \left(m_p l_p + m_r l_r\right)\cos\theta_p\right) \qquad L_p = \frac{1}{2}\left(J_p + m_p l_p^2 + m_r l_r^2\right)\dot{\theta}_p^2 - g\left(\left(m_p + m_r\right)l_r - \left(m_p l_p + m_r l_r\right)\cos\theta_p\right)$$

The Lagrange equation for the pendulum is the time derivative of the partial derivative of the Lagrangian with respect to the special derivative of the parameter minus the partial derivative of the Lagrangian with respect to the parameter. This is equal to the torque, or k times the current i.

$$\frac{d}{dt}\left(\frac{\partial L_p}{\partial\dot{\theta}_p}\right) - \frac{\partial L_p}{\partial\theta_p} = \tau_k$$

$$\frac{d}{dt}\left(\left(J_p + m_p l_p^2 + m_r l_r^2\right)\dot{\theta}_p\right) + g\left(m_p l_p + m_r l_r\right)\sin\theta_p = -ki \qquad \left(J_p + m_p l_p^2 + m_r l_r^2\right)\ddot{\theta}_p + g\left(m_p l_p + m_r l_r\right)\sin\theta_p = -ki$$

$$\ddot{\theta}_p + mg\frac{\left(m_p l_p + m_r l_r\right)}{m\left(J_p + m_p l_p^2 + m_r l_r^2\right)}\sin\theta_p = \frac{-ki}{\left(J_p + m_p l_p^2 + m_r l_r^2\right)} \qquad \ddot{\theta}_p + \frac{mgl}{J}\sin\theta_p = \frac{-ki}{J}$$

The Lagrange equation for the rotor is found trivially in the same manner.

$$L_r = KE_r - PE_r = \frac{1}{2}J_r\dot{\theta}_r^2 \qquad \frac{d}{dt}\left(\frac{\partial L_r}{\partial\dot{\theta}_r}\right) - \frac{\partial L_r}{\partial\theta_r} = \tau_k$$

$$\frac{d}{dt}\left(J_r\dot{\theta}_r\right) = ki \qquad J_r\ddot{\theta}_r = ki$$

$$\ddot{\theta}_r = \frac{ki}{J}$$

Defining the following constants

$$ki = k_u \qquad a = \omega^2_{np} = \frac{mgl}{J} \qquad b_p = \frac{k_u}{J} \qquad b_r = \frac{k_u}{J_r}$$

Our equations of motion become

$$\ddot{\theta}_p + a\sin\theta_p = -b_p \qquad \ddot{\theta}_r = b_r$$

Linearization into state space form (_/5pts)

The reaction wheel pendulum has the following equations of motion, disregarding frictional losses.

$$\ddot{\theta}_p + a\sin\theta_p = -b_p u \qquad \ddot{\theta}_r = -b_r u$$

We can rewrite the equations in state variable form as given by

$$\dot{x} = Ax + Bu \qquad x = \begin{bmatrix} x_1 & x_2 & x_3 & x_4 \end{bmatrix}^T = \begin{bmatrix} \theta_p & \dot{\theta}_p & \theta_r & \dot{\theta}_r \end{bmatrix}^T \qquad \dot{x} = \begin{bmatrix} x_2 \\ -b_p u - a\sin\theta_p \\ x_4 \\ -b_r u \end{bmatrix} = \begin{bmatrix} f_1(x,u) \\ f_2(x,u) \\ f_3(x,u) \\ f_4(x,u) \end{bmatrix}$$

Linearizing about the equilibrium position $\theta_p = \Pi$. We have that $\theta_p' = x_2 = 0$. The state space matrices can be determined via the Lagrangian method as follows.

$$A = \begin{bmatrix} \frac{df_1}{dx_1} & \frac{df_1}{dx_2} & \frac{df_1}{dx_3} & \frac{df_1}{dx_4} \\ \frac{df_2}{dx_1} & \frac{df_2}{dx_2} & \frac{df_2}{dx_3} & \frac{df_2}{dx_4} \\ \frac{df_3}{dx_1} & \frac{df_3}{dx_2} & \frac{df_3}{dx_3} & \frac{df_3}{dx_4} \\ \frac{df_4}{dx_1} & \frac{df_4}{dx_2} & \frac{df_4}{dx_3} & \frac{df_4}{dx4} \end{bmatrix} = \begin{bmatrix} \frac{d}{dx_1}(x_2) & \frac{d}{dx_2}(x_2) & \frac{d}{dx_3}(x_2) & \frac{d}{dx_4}(x_2) \\ \frac{d}{dx_1}(-b_p u - a\sin\theta_p) & \frac{d}{dx_2}(-b_p u - a\sin\theta_p) & \frac{d}{dx_3}(-b_p u - a\sin\theta_p) & \frac{d}{dx_4}(-b_p u - a\sin\theta_p) \\ \frac{d}{dx_1}(x_4) & \frac{d}{dx_2}(x_4) & \frac{d}{dx_3}(x_4) & \frac{d}{dx_4}(x_4) \\ \frac{d}{dx_1}(-b_r u) & \frac{d}{dx_2}(-b_r u) & \frac{d}{dx_3}(-b_r u) & \frac{d}{dx4}(-b_r u) \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ a & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \end{bmatrix}$$

$$B = \begin{bmatrix} \frac{df_1}{du} \\ \frac{df_2}{du} \\ \frac{df_3}{du} \\ \frac{df_4}{du} \end{bmatrix} = \begin{bmatrix} \frac{d}{du}(x_2) \\ \frac{d}{du}(-b_p u - a\sin\theta_p) \\ \frac{d}{du}(x_4) \\ \frac{d}{du}(-b_r u) \end{bmatrix} = \begin{bmatrix} 0 \\ -b_p \\ 0 \\ b_r \end{bmatrix} \qquad \begin{bmatrix} \delta\dot{\theta}_p \\ \delta\ddot{\theta}_p \\ \delta\dot{\theta}_r \\ \delta\ddot{\theta}_r \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ a & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \end{bmatrix} \cdot \begin{bmatrix} \delta\theta_p \\ \delta\dot{\theta}_p \\ \delta\theta_r \\ \delta\dot{\theta}_r \end{bmatrix} + \begin{bmatrix} 0 \\ -b_p \\ 0 \\ b_r \end{bmatrix} \cdot u$$

Full State Feedback Control with Friction Compensation (_/50)

1. Explain (in words) the development of the PD control (two-state and three-state feedback controllers in Chapter 4) with friction compensation (system identification in Chapter 2). How did you arrive at the values of your friction compensator? Was friction compensation beneficial to your state feedback controller or not, and why? (1-2 paragraphs of text) 10pts

The Friction Compensation values were calculated for velocity control by considering friction as a linear function of. The values of friction are derived using open loop system identification by keeping the pendulum at a fixed position and running the motor at constant reference speed for different velocities and in both directions. This allows us to record the control effort and gives us a linear fit of friction with the angular velocity.

The data collected during our experiment is summarized in the table below.

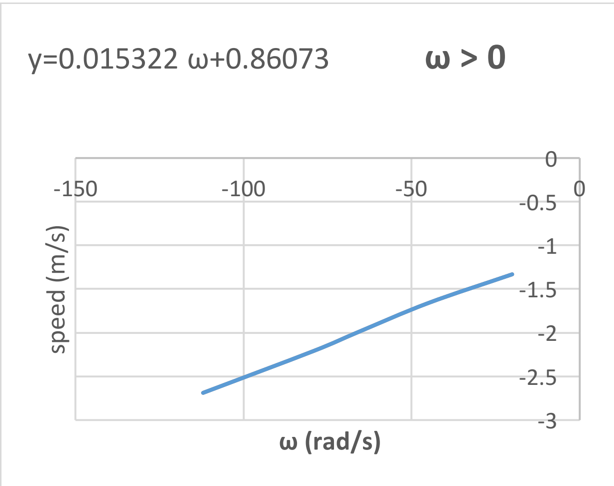
Table 1: Friction Compensation

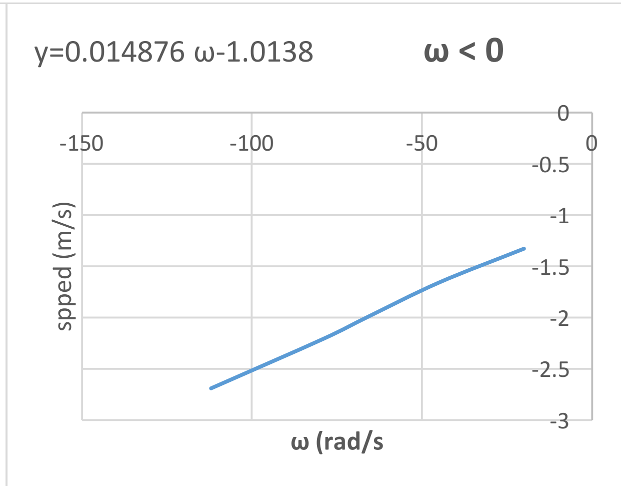
(rad/s)	20	40	60	88	112
Control Effort Positive Speed	1.778	1.4596	1.7769	2.2172	2.5752

(rad/s)	-20	-45	-66	-79	-112
Control Effort Negative Speed	-1.3306	-1.6611	-1.9970	-2.2100	-2.6925

After performing a curve fit on the data as shown below we arrived at the following equations.

$$\left\{\begin{array}{ll} f = 0.015322\omega + 0.086073 & \omega > 0 \\ f = 0.014976\omega - 1.0138 & \omega < 0 \end{array}\right\}$$





This yielded the following values for the frictional constants.

$\mathbf{b^+}$	$\mathbf{c^+}$	$\mathbf{b^-}$	$\mathbf{c^-}$
0.015322	0.086073	0.014976	-1.0138

Friction composition enhances our feedback controller by allowing the controller to maintain a steady state equilibrium without as great of an effort.

In order to design a proportional controller with friction compensation, we started with the following system parameters.

$$t_r = 0.2s \qquad u = 100 \cdot 1(t)\frac{rad}{s} \qquad b_r = 198\frac{rad}{s}$$

$$\frac{\omega(s)}{U(s)} = \frac{b_r}{s} \qquad u(t) = -k(\omega_r - \omega)$$

Writing the system equation in the Laplace domain and then transforming back allows us to swiftly compute the time response

$$\omega(s) = -k\left(\frac{\omega_r}{s} - \omega(s)\right)\frac{b_r}{s} \qquad \omega(s) = -kb_r\omega_r\frac{1}{s(s-kb_r)} = \left(\frac{r}{s} - \frac{r}{s-kb_r}\right)$$

$$r = kb_r \qquad \omega(t) = r\left(1 - e^{-kb_r t}\right)$$

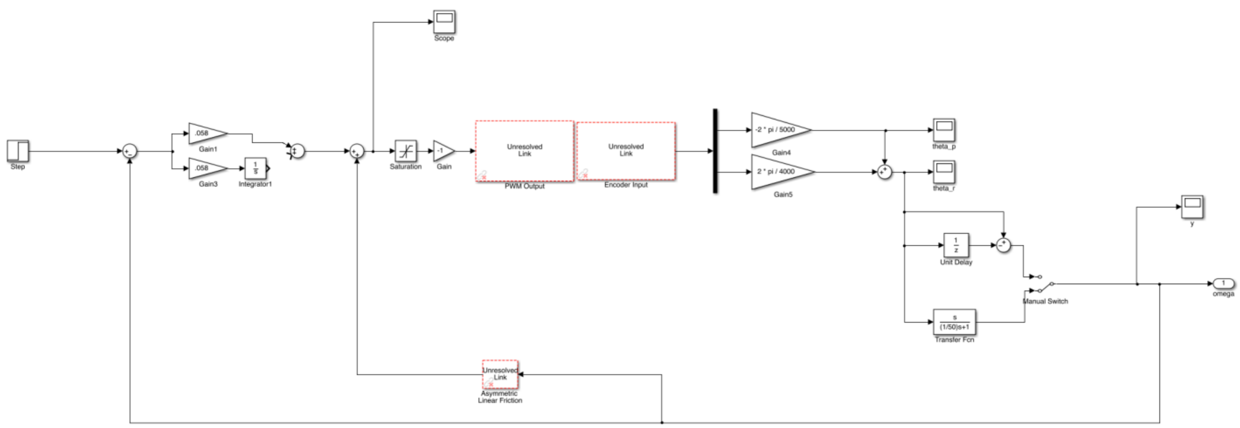
Using our knowledge of the rise time (t_r), calculation of the gain for our proportional controller is now trivial

$$r\left(1 - e^{-kb_r t_r}\right) = 0.9r \qquad e^{-kb_r 0.2} = 0.1 \qquad -kb_r 0.2 = \ln(0.1)$$

$$k = \frac{-\ln(0.1)}{0.2b_r} = \frac{-\ln(0.1)}{0.2(198)} = 0.0581$$

On implementation in Windows Target, we observe that the swing of the pendulum around the equilibrium position is much lower with application of friction compensation. This is because the friction compensation allows the controller to pull back with less torque, resulting in smaller swing.

The block diagram for our implementation is shown below.



Proof closed-loop system is stable in the inverted position (_/20)

2. Provide a mathematical *proof* that the linearized, frictionless closed-loop system is stable in the inverted position. That is, prove that theta_ p(t) goes to pi as time t goes to infinity for the 3-state feedback controller you designed.
First, show that the only place where $\dot{x}$ is zero is at your linearized equilibrium position. Then, show that for the linearized system $\dot{x} = Ax + Bu$, the equilibrium state x=[0, 0, whatever, 0] is stable. (1-2 pages of equations, pole plots, block diagrams, etc.) 20pts
***Hint*: First, substitute your control in for *u*. You should arrive at a differential equation with x as the only variable. Set $\dot{x}$ to zero and solve for x to find the equilibrium points. Now, note that we don't care what theta_ r is, and that theta_ r doesn't affect any of the other states. So, to check that [0 0 whatever 0] a stable equilibrium, create a smaller 3x3 A matrix and 3x1 B matrix where your states are delta_ theta_ p, theta_ p_ dot, and theta_ r_ dot. If this smaller system (including your 3-state feedback) is stable, then delta_ theta_ p, theta_ p_ dot, and theta_ r_ dot all converge to 0, and [0 0 whatever 0] is a stable equilibrium. You *must* use Microsoft Equation Editor, LaTeX, or a comparable program to generate equations.**

When we designed the three state feedback controller, our goal was to control the velocity and leave the position alone. This stabilizes the velocity of the rotor but does not affect the position. Thus, we can ignore θ'_r since we are not controlling the position of the rotor. We leave the system as a four by four matrix for convenience and set the gain for θ'_r to zero.

We choose to stabilize the system about the equilibrium point $\theta_p = \Pi$ and so we define $\delta\theta_p$ as the deviation from this equilibrium point as follows.

$$\delta\theta_p = \theta_p - \pi$$
$$\delta\theta_r = \theta_r$$

The system is then modeled as follows.

$$\begin{bmatrix} \delta\dot{\theta}_p \\ \delta\ddot{\theta}_p \\ \delta\dot{\theta}_r \\ \delta\ddot{\theta}_r \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ a & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \end{bmatrix} \cdot \begin{bmatrix} \delta\theta_p \\ \delta\dot{\theta}_p \\ \delta\theta_r \\ \delta\dot{\theta}_r \end{bmatrix} + \begin{bmatrix} 0 \\ -b_p \\ 0 \\ b_r \end{bmatrix} \cdot u$$

For the state space system given above, we determined the system parameters experimentally. They are given below.

b_p	b_r	a
1.0951	206.5608	70.1741

In order to prove stability, we take the input to the system as -Kx and find the eigenvalues of the system matrix A-BK. For our experimentally determined parameters the system matrix is calculated below.

$$\dot{x} = Ax + Bu = Ax - BKx = (A - BK)x$$

$$A - BK = \begin{bmatrix} 0 & 1 & 0 & 0 \\ a & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \end{bmatrix} - \begin{bmatrix} 0 \\ -b_p \\ 0 \\ b_r \end{bmatrix} \cdot \begin{bmatrix} K_1 & K_2 & K_3 & K_4 \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 70.1741 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \end{bmatrix} - \begin{bmatrix} 0 \\ -1.0951 \\ 0 \\ 206.5608 \end{bmatrix} \cdot \begin{bmatrix} -191.2664 & -20.3910 & 0 & -0.0330 \end{bmatrix}$$

$$A - BK = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 70.1741 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \end{bmatrix} - 10^4 \cdot \begin{bmatrix} 0 & 0.0001 & 0 & 0 \\ 0.0209 & 0.0022 & 0 & 0 \\ 0 & 0 & 0 & 0.0001 \\ -3.9508 & -0.4212 & 0 & -0.0007 \end{bmatrix} = 10^4 \cdot \begin{bmatrix} 0 & 0.0001 & 0 & 0 \\ -0.0139 & -0.0022 & 0 & 0 \\ 0 & 0 & 0 & 0.0001 \\ 3.9508 & 0.4212 & 0 & 0.0007 \end{bmatrix}$$

The eigenvalues were calculated via Mat Lab and are shown below. It can be observed that they are all in the left half plane, proving stability.

$$\lambda_{A-BK} = \begin{bmatrix} 0.0000 + 0.0000i \\ -4.8795 + 7.7015i \\ -4.8795 - 7.7015i \\ -5.7547 + 0.0000i \end{bmatrix}$$

We now know that since A-BK is stable and full rank, and since x'=(A-BK)x must go to zero at equilibrium, the only solution is given by x=0 as shown below.

$$x_{eq} = \begin{bmatrix} \delta\theta_p \\ \delta\dot{\theta}_p \\ \delta\theta_r \\ \delta\dot{\theta}_r \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}$$

We know know that since $\delta\theta_p$ is defined as θ_p-Π, the equilibrium point occurs at θ_p=Π as described below. As time goes infinity, at steady state, x'=0 and this implies that θ_p goes to pi and θ_r goes to zero as shown below. Hence the position at the upward equilibrium is stable.

$$\delta\theta_p = \theta_p - \pi = 0 \rightarrow \theta_p = \pi$$

$$\delta\theta_r = \theta_r = 0 \rightarrow \theta_r = 0$$

$$
\begin{bmatrix} \delta\theta_p \\ \delta\dot{\theta}_p \\ \delta\theta_r \\ \delta\dot{\theta}_r \end{bmatrix} = \begin{bmatrix} \pi \\ 0 \\ 0 \\ 0 \end{bmatrix}
$$

The final block diagram for our controller is shown below.

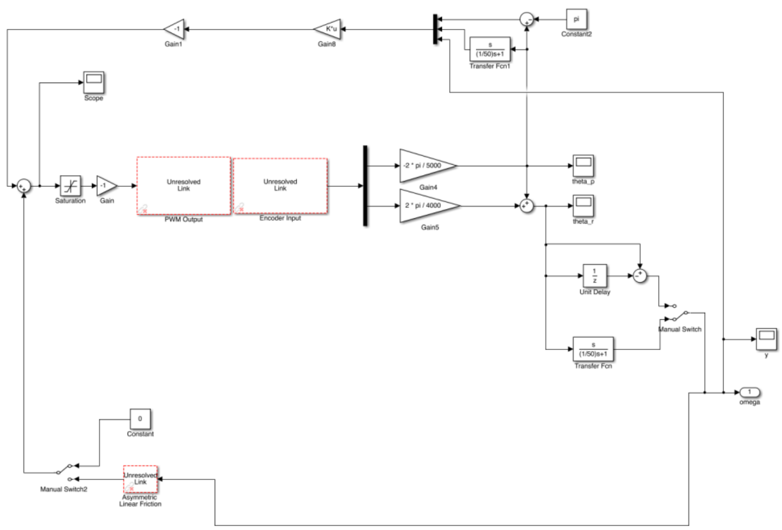


Table of Values (_/10)

3. From your Simulink simulations of the RWP, give the maximum IC deviation, pulse disturbance, and constant perturbation that the controller can stabilize. (1 table of values) 10pts

Based on the Simulink simulations of our state feedback design for the control of the RWP, the maximum IC deviations, pulse disturbance, and constant perturbation that the controller can stabilize are included in Table 1 below.

Table 2: Robustness Comparisons

	Two-State Feedback		Three-State Feedback	
	$\delta\theta_p$	$\delta\dot{\theta}_p$	$\delta\theta_p$	$\delta\dot{\theta}_p$
Max IC deviations	0.96 rad	0.98rad/s	0.96 rad	1.05rad/s
Max pulse	5		6	
Max disturbance	3		4.1	

Behavior of System (_/10)

4. From your Windows Target implementation, describe the behavior of the system caused by your controller. (1 paragraph) 10pts

After implementation in Windows Target, we observe that the pendulum successfully balances itself at the equilibrium position. The small disturbances are rejected by the controller. The three state feedback has an extra state that it must control. Therefore, it should be less robust than the two state feedback and this is confirmed by our experiment.

Full State Feedback Control with Decoupled Observer (_/50)

1. Explain the following: Why are observers used? Why can we decouple the 4-state observer design into two 2-state observers? What is the advantage of this versus a single 4-state observer? 10pts

An observer design is now used for the Reaction Wheel Pendulum (RWP) in order to replace the full state feedback controller and provide an accurate estimate of both the velocity and position states which describe our system.

State space Model of our system is of the form:

$$\dot{x} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ a & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \end{bmatrix} x + \begin{bmatrix} 0 \\ -b_p \\ 0 \\ b_r \end{bmatrix} u$$

From the above form we observe that A is of a block diagonal form. Hence the dynamics of the 4 observer system can be decoupled to a 2 observer system as shown below.

$$\begin{bmatrix} \dot{x}_{1,2} \\ \dot{x}_{3,4} \end{bmatrix} = \begin{bmatrix} M & 0 \\ 0 & N \end{bmatrix} \cdot \begin{bmatrix} x_{1,2} \\ x_{3,4} \end{bmatrix} = \begin{bmatrix} P \\ Q \end{bmatrix} u \qquad \left\{ \begin{array}{l} \dot{x}_{1,2} = Mx_{1,2} + Pu \\ \dot{x}_{3,4} = Nx_{3,4} + Qu \end{array} \right\}$$

$$\left\{ \begin{array}{ll} \dot{x}_{1,2} = \begin{bmatrix} 0 & 1 \\ a & 0 \end{bmatrix} x_{1,2} + \begin{bmatrix} 0 \\ -b_p \end{bmatrix} u & C_{1,2} = \begin{bmatrix} 1 & 0 \end{bmatrix} \\ \dot{x}_{3,4} = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} x_{3,4} + \begin{bmatrix} 0 \\ b_r \end{bmatrix} u & C_{3,4} = \begin{bmatrix} 1 & 0 \end{bmatrix} \end{array} \right\}$$

The eigenvalues of A are the union of the eigenvalues of M and N. The advantage of using this instead of a 4 state observer is that it allows us complete control over the controller and the observer poles separately.

Proof observer states converge to the real states (_/20)

2. You already proved that the 3-state feedback controller stabilizes the pendulum in the inverted configuration. Now prove that the observer states converge to the real states over time. That is, show that the error between the actual states and estimated states $e = x - \hat{x}$ goes to zero over time. 20pts
***Hint*: Differentiate the equation for the error. Next substitute the equations you have for x_dot, xhat_dot, and *u*. You should end up with a differential equation with the error e as the only variable. Next, prove that the poles are stable. Set e_dot equal to zero and show that e = [0,0,0,0] is the *only* stable equilibrium for this equation.**

A system is observable if the estimated state approaches actual state over time. In order to verify the stability of our system we must prove that this is the case. The error between estimated states and actual states is given by the following equation.

$$e = x - \hat{x}$$

On differentiating the error, we get

$$\dot{e} = \dot{x} - \dot{\hat{x}} = Ax + Bu - \left((A - LC)\hat{x} + Bu + Ly \right) \qquad \dot{e} = (A - LC)x - (A - LC)\hat{x} = (A - LC)e$$

The eigenvalues of A-LC are calculated as follows

$$A - LC = \begin{bmatrix} 0 & 1 & 0 & 0 \\ a & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \end{bmatrix} - \begin{bmatrix} L_{11} & L_{12} \\ L_{21} & L_{22} \\ L_{31} & L_{32} \\ L_{41} & L_{42} \end{bmatrix} \cdot \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 70.1741 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \end{bmatrix} - 10^4 \cdot \begin{bmatrix} 0.0250 & 0 \\ 1.5070 & 0 \\ 0 & 0.0380 \\ 0 & 3.6000 \end{bmatrix} \cdot \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}$$

$$A - LC = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 70.1741 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \end{bmatrix} - 10^4 \cdot \begin{bmatrix} 0.0250 & 0 & 0 & 0 \\ 1.5070 & 0 & 0 & 0 \\ 0 & 0 & 0.0380 & 0 \\ 0 & 0 & 3.6000 & 0 \end{bmatrix} = 10^4 \cdot \begin{bmatrix} -0.0250 & 0.0001 & 0 & 0 \\ -1.5070 & 0 & 0 & 0 \\ 0 & 0 & -0.0380 & 0.0001 \\ 0 & 0 & -3.6000 & 0 \end{bmatrix}$$

$$\lambda = \begin{bmatrix} -150.0035 \\ -99.9965 \\ -200.000 \\ -180.000 \end{bmatrix}$$

The eigenvalues are -150, -99, -200, and -180, which to the closed loop poles which we designed. As all the eigenvalues lie in LHP, the system is stable.

If we set e'=0 in the equation e'=(A-LC)e in order to find the steady state equilibrium, we know that since A-LC is stable and full rank the only solution is given by the vector below.

$$e_{eq} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}$$

Therefore, since the error goes to zero over time, the observer poles provide a perfect estimation of actual steady state value.

The block diagram of the full-state feedback control with observer added is shown below.

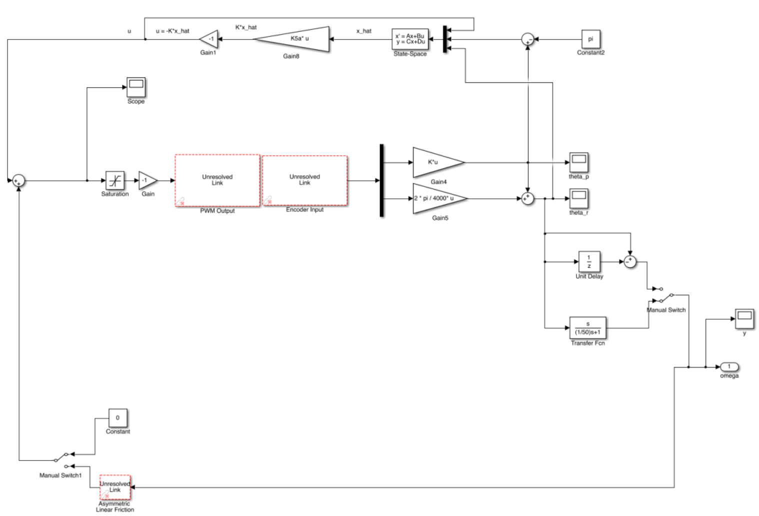


Table of Values (_/10)

3. From your Simulink simulations of the RWP, give the maximum IC deviation, pulse disturbance, and constant perturbation that the controller can stabilize. (1 table of values) 10pts

Based on the Simulink simulations of our observer design for the control of the RWP, the maximum IC deviations, pulse disturbance, and constant perturbation that the controller can stabilize are included in Table 2 below.

Table 2 Robustness Comparisons

	Two-State Feedback		Three-State Feedback		Observer	
	$\delta\theta_p$	$\delta\dot{\theta}_p$	$\delta\theta_p$	$\delta\dot{\theta}_p$	$\delta\theta_p$	$\delta\dot{\theta}_p$
Max IC deviations	0.96 rad	0.98rad/s	0.96 rad	1.05rad/s	0.99rad	0.82rad/s
Max pulse	5		6		5.5	
Max disturbance	3		4.1		4.5	

Behavior of System (_/10)

4. From your Windows Target implementation, describe the behavior of the system caused by your controller with the observer. (1 paragraph) 10pts

After implementing the observer in Windows Target, we verified that the simulation result agrees with the actual system behavior of the Reaction Wheel Pendulum (RWP). As predicted during simulation, the Reaction Wheel Pendulum (RWP) rejects the small disturbances and maintains its position. The poles were placed at -100, -180, -150 and -200. These poles were selected so that they are much faster than the controller poles. After the implementation of the observer design, the entire system became more sensitive and the error dies out quickly enough to reposition the pendulum arm almost perfectly about the top equilibrium.

Extra Credit (1/2 page each, 35 possible pts)

Explain your approach to the two optional sections. What techniques did you use? Also include your Windows Target model(s) in an appendix. Note: these sections do not count against your total number of pages

In the next part of the project, we designed a switching pendulum with two equilibrium points: the top equilibrium and the bottom equilibrium. We had previously designed a controller for balancing up using the three state feedback system. We now redesign this system for down control. Firstly, we calculate the Linearization of the system again about the point θ_p=0.

The derivation of the system equation is shown below.

$$A' = \begin{bmatrix} 0 & 1 & 0 & 0 \\ -a & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ -70.1741 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \end{bmatrix}$$

$$\dot{x} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ -70.1741 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \end{bmatrix} x + \begin{bmatrix} 0 \\ -b_p \\ 0 \\ b_r \end{bmatrix} u$$

We used the same approach described above to calculate new values of the gain.

Otherwise our control implementation was exactly the same.

$$K' = \begin{bmatrix} K'_1 & K'_2 & K'_3 \end{bmatrix} = \begin{bmatrix} -50.8274 & -8.1083 & 0.0321 \end{bmatrix}$$

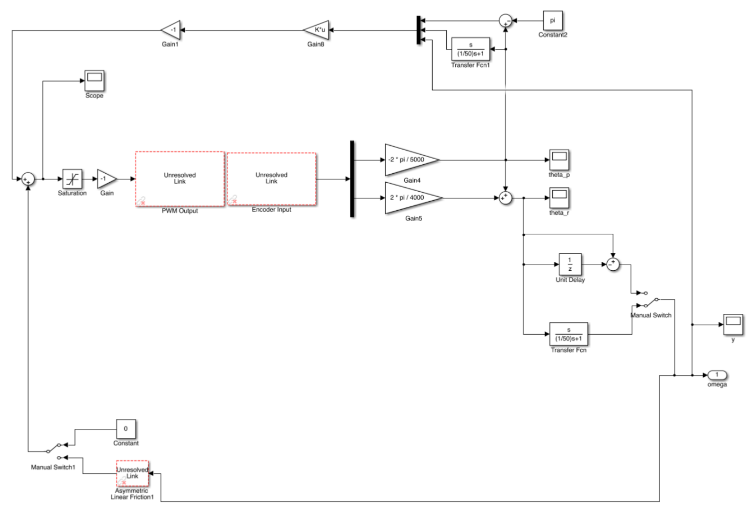
It can be shown as below that the Eigen values of this controller are once again in the LHP.

$$A - BK = \begin{bmatrix} 0 & 1 & 0 & 0 \\ -a & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \end{bmatrix} - \begin{bmatrix} 0 \\ -b_p \\ 0 \\ b_r \end{bmatrix} \cdot \begin{bmatrix} K_1 & K_2 & K_3 & K_4 \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ -70.1741 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \end{bmatrix} - \begin{bmatrix} 0 \\ -1.0951 \\ 0 \\ 206.5608 \end{bmatrix} \cdot \begin{bmatrix} -50.8274 & -8.1083 & 0 & 0.0321 \end{bmatrix}$$

$$A - BK = \begin{bmatrix} 0 & 1 & 0 & 0 \\ -70.1741 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \end{bmatrix} - 10^4 \cdot \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0.0056 & 0.0009 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ -1.0499 & -0.1675 & 0 & 0.0007 \end{bmatrix} = 10^4 \cdot \begin{bmatrix} 0 & 0.0001 & 0 & 0 \\ -0.0126 & -0.0009 & 0 & 0 \\ 0 & 0 & 0 & 0.0001 \\ 1.0499 & 0.1675 & 0 & -0.0007 \end{bmatrix}$$

$$\lambda = \begin{bmatrix} 0 \\ -4.2421 + 6.9449i \\ -4.2421 - 6.9449i \\ -7.0258 \end{bmatrix}$$

The block diagram of our final design is shown below.

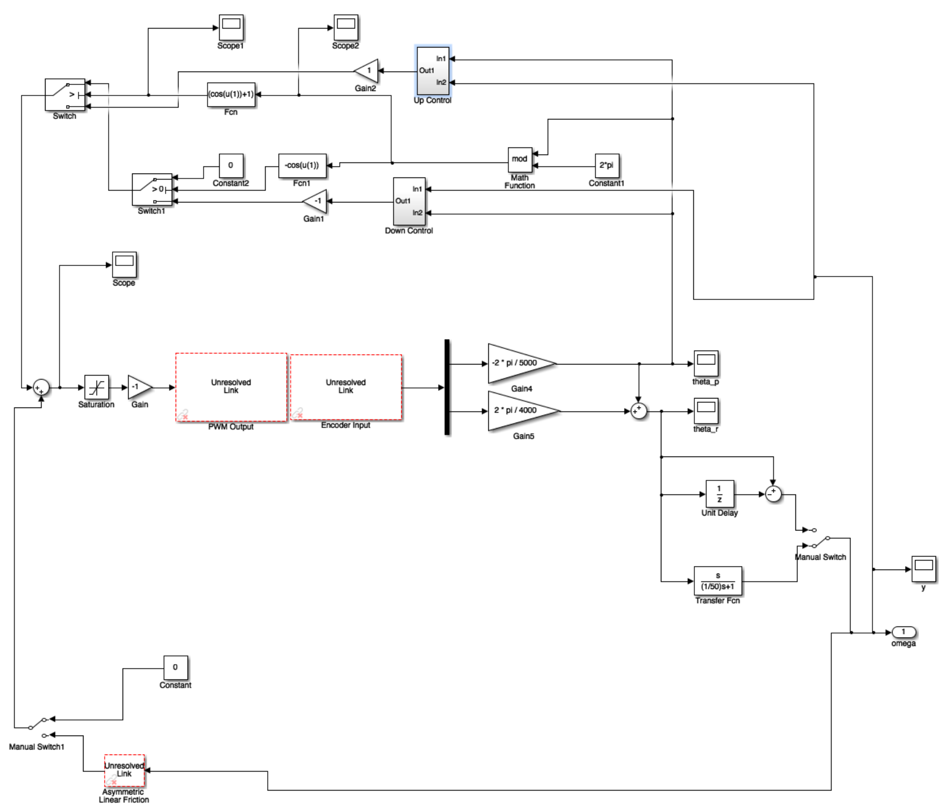


Implementing the three state feedback with the above values of K for the down controller along with the previously designed three state up controller, we are able to successfully implement a pendulum controller which intelligently switched to the appropriate control structure within the range of these respective equilibria.

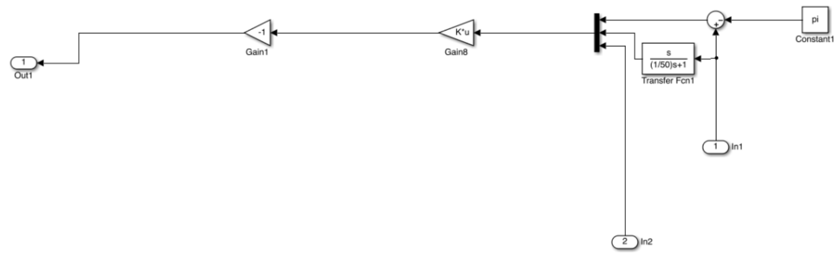
Our next task was to implement swing up control for the Reaction Wheel Pendulum (RWP). We decided to accomplish this by inverting the output from our stabilizing controller for the downwards equilibrium so that our controller destabilized the system about this equilibrium point. This allowed us to generate a swing up which could pull our controller into the upwards equilibrium. We then simply had a switch block which checked if the cosine of our angle plus one was within a certain threshold of one, that is, if our angle was within a certain threshold of Π, allowing us to transition into stabilizing control about the upwards equilibrium.

We ran into a major issue where our controller's behavior became indeterminate if the swing up took the controller over the unstable equilibrium at the top to fast for our controller to kick on and stabilize the system. We realized that two issues in our control structure were causing this problem. One was that our angle was not resetting to zero after the pendulum passed Π, so our swing up would not even work after a failure to stabilize. We solved this problem by adding a mod operator to our angle so that it always remained within 2Π. Our other problem was that the force of the swing through the upwards equilibrium was too great. We solved this problem by implementing a second switch which checked if the negative cosine of our angle was within threshold of zero, or our angle was close to pie over two. This allowed us to have a regime where the pendulum could slow down before transitioning to stabilization about the upwards equilibrium. Our design was robust to the satisfaction of our TA. Further improvement could be made by adding a variable inverse gain with feathering instead of the constant inverse gain in front of our down controller. This would allow the gain to be decreased closer to a higher angle and better control the force torque output of the pendulum such that it smoothly transitioned to the stabilizing regime.

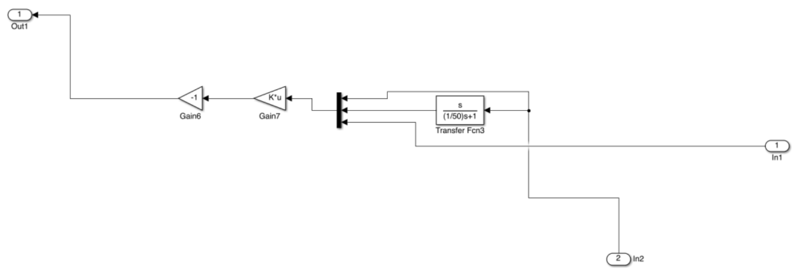
The overall block diagram for our intelligent swing up controller is shown below. It utilizes two controller blocks, the up controller and the down controller, whose diagrams are shown on the following page.



Up Controller



Down Controller



Conclusion:

Over the course of this lab, we designed six controllers, a proportional controller for friction compensation, a two state feedback controller, a three state feedback controller, a decoupled observer controller, a switching controller, and a swing up controller. The proportional controller allowed us to enhance all of our following designs by minimizing frictional losses. We found that in reality, the decoupled observer controller was the most sensitive, while the two state feedback controller was the most robust. However, only the three state feedback controller was able to stabilize the velocity of the rotor and the decoupled observer allowed full control of the system without perfect knowledge of the system states.

We decided to use our three state feedback controller to implement switching control and feedback control. After redesigning our controller for stabilization about the bottom equilibrium point, we were able to trivially implement switching control. For our swing up control, we simply inverted the output of our controller for the bottom equilibrium to introduce swing up instability, and used our switching control to stabilize the system about the top equilibrium point once it was at the desired position.

Over the course of this lab we went through a full control design process, including system identification, model validation, controller comparison, and design validation. We were able to use the tools we learned over the course of the semester to create an original design which implemented an intelligent controller which recognized current system properties and modified its own design based on the current system state. This seemingly simple design is actually the foundation of cutting edge research in the field of intelligent control. By completing this lab, we have taken the first step of many toward creating a new class of intelligent devices which interact with the world in new ways.