FORCE ANALYSIS AND SYSTEM EQUATIONS

The system parameters are defined as follows:

(M) mass of the cart 0.5 kg

(m) mass of the pendulum 0.2 kg

(b) coefficient of friction for cart 0.1 N/m/sec

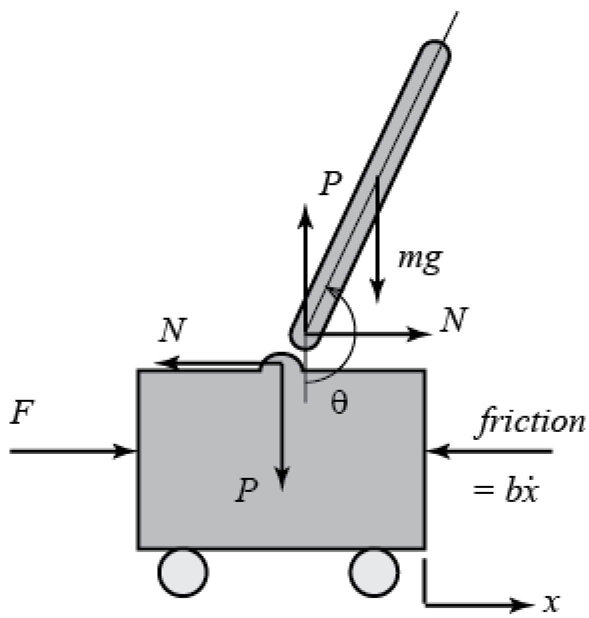
(l) length to pendulum center of mass 0.3 m

(I) mass moment of inertia of the pendulum 0.006 kg.m^2

The design requirements for this system are:

- Settling time for θ of less than 5 seconds
- Pendulum angle θ never more than 0.05 radians from the vertical

Below are the free-body diagrams of the two elements of the inverted pendulum system.



Summing the forces in the free-body diagram of the cart in the horizontal direction, the following equation of motion are obtained.

(1) $M\ddot{x} + b\dot{x} + N = F$

Note that the forces can be summed in the vertical direction for the cart, but no useful information would be gained.

Summing the forces in the free-body diagram of the pendulum in the horizontal direction, the following expression for the reaction force N will be obtained.

(2) $N = m\ddot{x} + ml\ddot{\theta}\cos\theta - ml\dot{\theta}^2\sin\theta$

If this equation is substituted into the first equation, one of the two governing equations for this system will be obtained.

(3) $(M+m)\ddot{x} + b\dot{x} + ml\ddot{\theta}\cos\theta - ml\dot{\theta}^2\sin\theta = F$

To get the second equation of motion for this system, sum the forces perpendicular to the pendulum. Solving the system along this axis greatly simplifies the mathematics. The following equation is obtained.

(4) $P\sin\theta + N\cos\theta - mg\sin\theta = ml\ddot{\theta} + m\ddot{x}\cos\theta$

To get rid of the P and N terms in the equation above, sum the moments about the centroid of the pendulum to get the following equation.

(5) $-Pl\sin\theta - Nl\cos\theta = I\ddot{\theta}$

Combining these last two expressions, the second governing equation is obtained.

(6) $(I + ml^2)\ddot{\theta} + mgl\sin\theta = -ml\ddot{x}\cos\theta$

Since the analysis and control design techniques we will be employing in this example apply only to linear systems, this set of equations needs to be linearized. Specifically, we will linearize the equations about the vertically upward equillibrium position, θ = π, and will assume that the system stays within a small neighborhood of this equillbrium. This assumption should be reasonably valid since under control we desire that the pendulum not deviate more than 20 degrees from the vertically upward position. Let ϕ represent the deviation of the pedulum's position from equilibrium, that is, θ = π + ϕ. Again presuming a small deviation (ϕ) from equilibrium, we can use the following small angle approximations of the nonlinear functions in our system equations:

(7) $\cos\theta = \cos(\pi + \phi) \approx -1$

(8) $\sin\theta = \sin(\pi + \phi) \approx -\phi$

(9) $\dot{\theta}^2 = \dot{\phi}^2 \approx 0$

After substituting the above approximations into nonlinear governing equations, we arrive at the two linearized equations of motion. Note u has been substituted for the input F.

(10) $(I + ml^2)\ddot{\phi} - mgl\phi = ml\ddot{x}$

(11) $(M + m)\ddot{x} + b\dot{x} - ml\ddot{\phi} = u$

Transfer Function

To obtain the transfer functions of the linearized system equations, we must first take the Laplace transform of the system equations assuming zero initial conditions. The resulting Laplace transforms are shown below.

(12) $(I + ml^2)\Phi(s)s^2 - mgl\Phi(s) = mlX(s)s^2$

(13) $(M + m)X(s)s^2 + bX(s)s - ml\Phi(s)s^2 = U(s)$

A transfer function represents the relationship between a single input and a single output at a time. To find our first transfer function for the output $\Phi(s)$ and an input of $U(s)$ we need to eliminate $X(s)$ from the above equations. Solve the first equation for $X(s)$.

(14) $$X(s) = \left[\frac{I + ml^2}{ml} - \frac{g}{s^2}\right]\Phi(s)$$

Then substitute the above into the second equation.

(15) $$(M + m)\left[\frac{I + ml^2}{ml} - \frac{g}{s^2}\right]\Phi(s)s^2 + b\left[\frac{I + ml^2}{ml} - \frac{g}{s^2}\right]\Phi(s)s - ml\Phi(s)s^2 = U(s)$$

Rearranging, the transfer function

(16) $$\frac{\Phi(s)}{U(s)} = \frac{\frac{ml}{q}s^2}{s^4 + \frac{b(I+ml^2)}{q}s^3 - \frac{(M+m)mgl}{q}s^2 - \frac{bmgl}{q}s}$$

where,

(17) $q = [(M+m)(I+ml^2) - (ml)^2]$

From the transfer function above it can be seen that there is both a pole and a zero at the origin. These can be cancelled and the transfer function becomes the following.

$$P_{pend}(s) = \frac{\Phi(s)}{U(s)} = \frac{\frac{ml}{q}s}{s^3 + \frac{b(I+ml^2)}{q}s^2 - \frac{(M+m)mgl}{q}s - \frac{bmgl}{q}} \qquad [\frac{rad}{N}]$$
(18)

Second, the transfer function with the cart position $X(s)$ as the output can be derived in a similar manner to arrive at the following.

$$P_{cart}(s) = \frac{X(s)}{U(s)} = \frac{\frac{(I+ml^2)s^2 - gml}{q}}{s^4 + \frac{b(I+ml^2)}{q}s^3 - \frac{(M+m)mgl}{q}s^2 - \frac{bmgl}{q}s} \qquad [\frac{m}{N}]$$
(19)

The open-loop transfer functions of the inverted pendulum system is as follows.

$$P_{pend}(s) = \frac{\Phi(s)}{U(s)} = \frac{\frac{ml}{q}s}{s^3 + \frac{b(I+ml^2)}{q}s^2 - \frac{(M+m)mgl}{q}s - \frac{bmgl}{q}} \qquad [\frac{rad}{N}]$$
(1)

$$P_{cart}(s) = \frac{X(s)}{U(s)} = \frac{\frac{(I+ml^2)s^2 - gml}{q}}{s^4 + \frac{b(I+ml^2)}{q}s^3 - \frac{(M+m)mgl}{q}s^2 - \frac{bmgl}{q}s} \qquad [\frac{m}{N}]$$
(2)

where

(3) $q = (M+m)(I+ml^2) - (ml)^2$

The above two transfer functions are valid only for small values of the angle ϕ, which is the angular displacement of the pendulum from the vertically upward position. Also, the absolute pendulum angle θ is equal to π + ϕ.

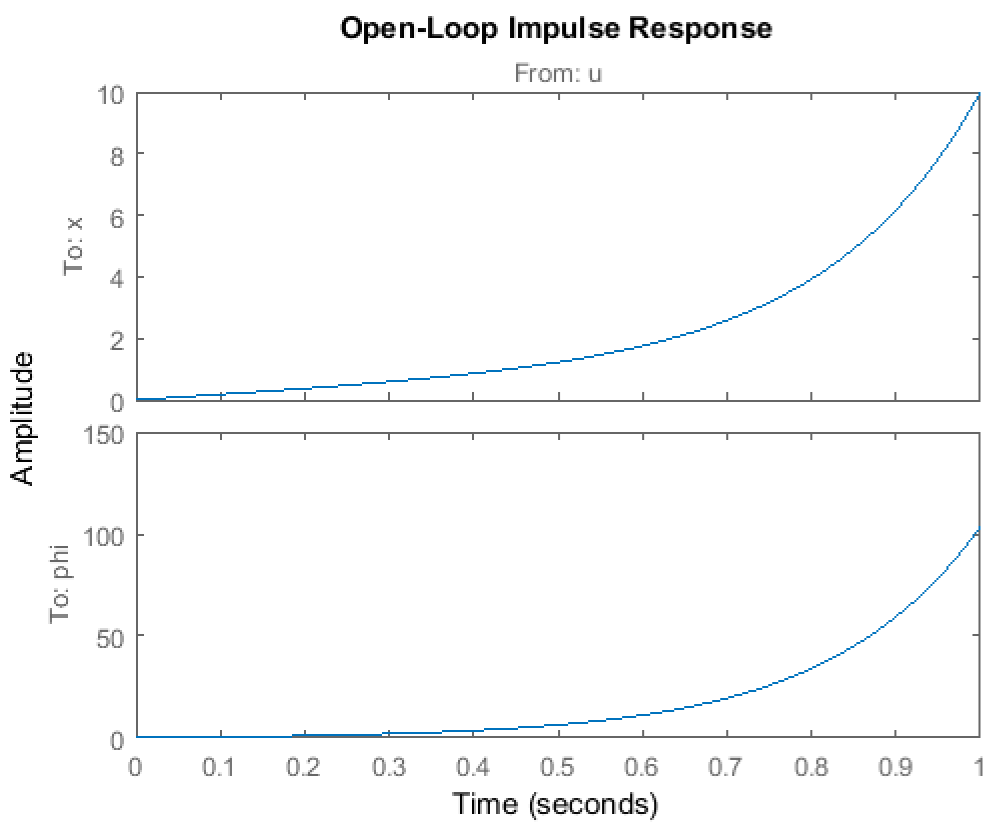
Considering the response of the pendulum to a 1-Nsec impulse applied to the cart, the design requirements for the pendulum are:

- Settling time for θ of less than 5 seconds
- Pendulum angle θ never more than 0.05 radians from the vertical

Additionally, the requirements for the response of the system to a 0.2-meter step command in cart position are:

- Settling time for x and θ of less than 5 seconds
- Rise time for x of less than 0.5 seconds
- Pendulum angle θ never more than 20 degrees (0.35 radians) from the vertical

OPEN-LOOP IMPULSE RESPONSE



From the plot, the system response is entirely unsatisfactory. In fact, it is not stable in open loop. Although the pendulum's position is shown to increase past 100 radians (15 revolutions), the model is only valid for small ϕ. It can be seen that the cart's position moves infinitely far to the right, though there is no requirement on cart position for an impulsive force input.

The poles of a system can also tell about its time response. Since this system has two outputs and one input, it is described by two transfer functions. In general, all transfer functions from each input to each output of a multi-input, multi-output (MIMO) system will have the same poles (but different zeros) unless there are pole-zero cancellations. We will specifically examine the poles and zeros of the system using the MATLAB function **zpkdata**. The parameter 'v' shown below returns the poles and zeros as column vectors rather than as cell arrays.

The zeros and poles of the system where the pendulum position is the output are found as shown below:

zeros = 0

poles = 5.5651 -5.6041 -0.1428

Likewise, the zeros and poles of the system where the cart position is the output are found as follows:

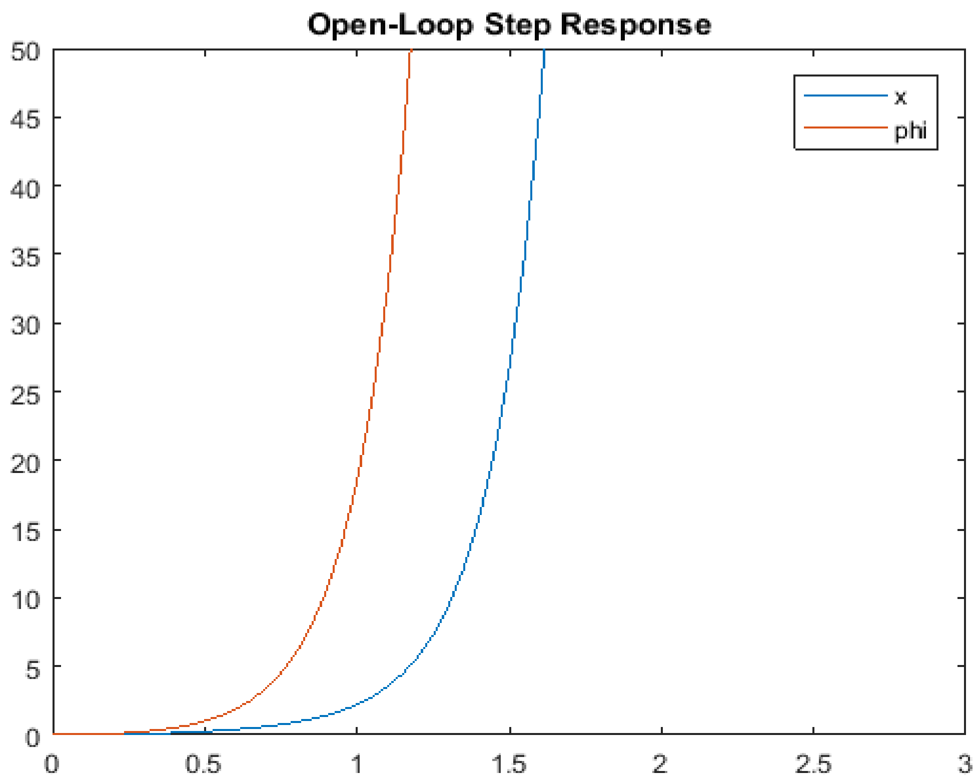
zeros = 4.9497 -4.9497

poles = 0 5.565 -5.6041 -0.1428

As predicted, the poles for both transfer functions are identical. The pole at 5.5651 indicates that the system is unstable since the pole has positive real part. In other words, the pole is in the right half of the complex s-plane. This agrees with what is observed above.

OPEN LOOP STEP RESPONSE

Since the system has a pole with positive real part its response to a step input will also grow unbounded.



Some important characteristics of the response are identified.

cart_info =

struct with fields:

SettlingTime: 9.9959

Min: 0

MinTime: 0

Max: 8.7918e+21

MaxTime: 10

pend_info =

struct with fields:

SettlingTime: 9.9959

Min: 0

MinTime: 0

Max: 1.0520e+23

MaxTime: 10

The above results confirm our expectation that the system's response to a step input is unstable.

It is apparent from the analysis above that some sort of control will need to be designed to improve the response of the system.

PID CONTROLLER DESIGN

We will design a PID controller for the inverted pendulum system. In the design process a single-input, single-output plant is assumed as described by the following transfer function. Otherwise stated, we will attempt to control the pendulum's angle without regard for the cart's position.

(1) $$P_{pend}(s) = \frac{\Phi(s)}{U(s)} = \frac{\frac{ml}{q}s}{s^3 + \frac{b(I+ml^2)}{q}s^2 - \frac{(M+m)mgl}{q}s - \frac{bmgl}{q}} \qquad [\frac{rad}{N}]$$

where,

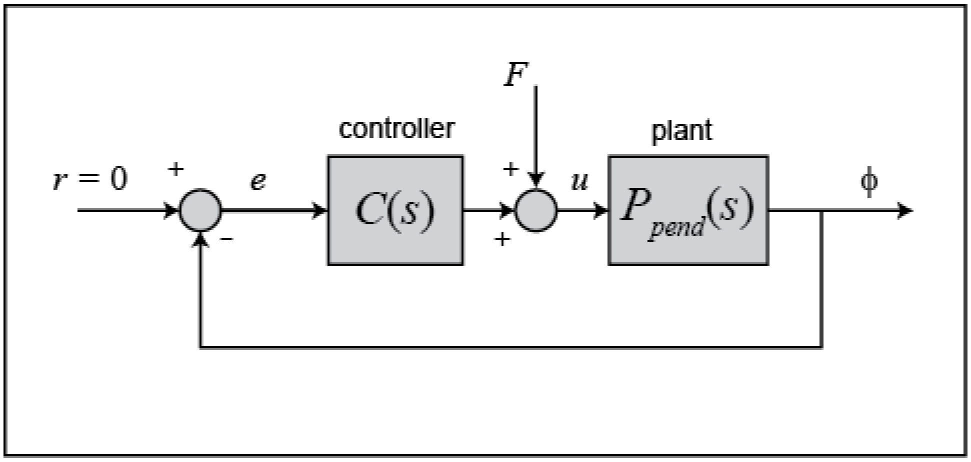
(2) $$q = (M+m)(I+ml^2) - (ml)^2$$

More specifically, the controller will attempt to maintain the pendulum vertically upward when the cart is subjected to a 1-Nsec impulse. Under these conditions, the design criteria are:

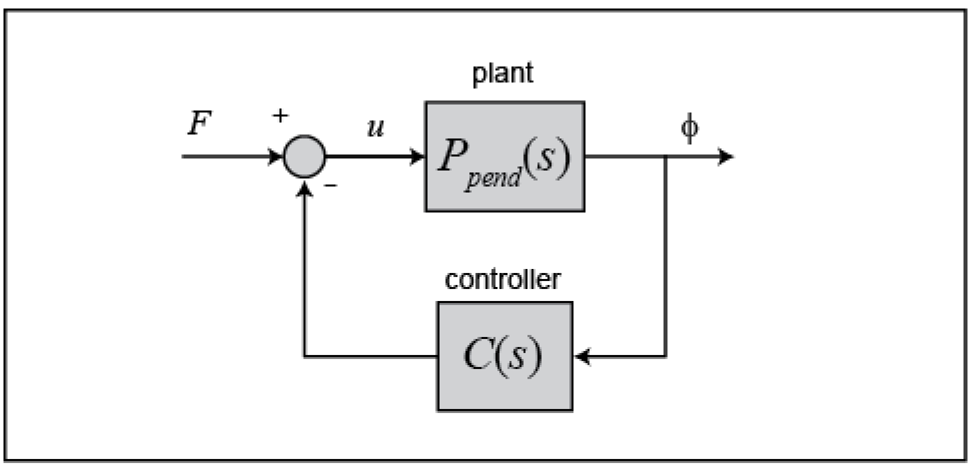
- Settling time of less than 5 seconds
- Pendulum should not move more than 0.05 radians away from the vertical

SYSTEM STRUCTURE

The structure of the controller for this problem is a little different than the standard control problems you may be used to. Since we are attempting to control the pendulum's position, which should return to the vertical after the initial disturbance, the reference signal we are tracking should be zero. This type of situation is often referred to as a Regulator problem. The external force applied to the cart can be considered as an impulsive disturbance. The schematic for this problem is depicted below.



The system is re-arranged as follows.



The resulting transfer function $T(s)$ for the closed-loop system from an input of force F to an output of pendulum angle ϕ is then determined to be the following.

(3) $$T(s) = \frac{\Phi(s)}{F(s)} = \frac{P_{pend}(s)}{1 + C(s)P_{pend}(s)}$$

Before we begin designing our PID controller, we first need to define our plant within MATLAB.

.

```
M = 0.5;

m = 0.2;

b = 0.1;

I = 0.006;

g = 9.8;

l = 0.3;

q = (M+m)*(I+m*l^2)-(m*l)^2;

s = tf('s');
P_pend = (m*l*s/q)/(s^3 + (b*(I + m*l^2))*s^2/q - ((M + m)*m*g*l)*s/q - b*m*g*l/q);
```

Next we will define a PID controller.

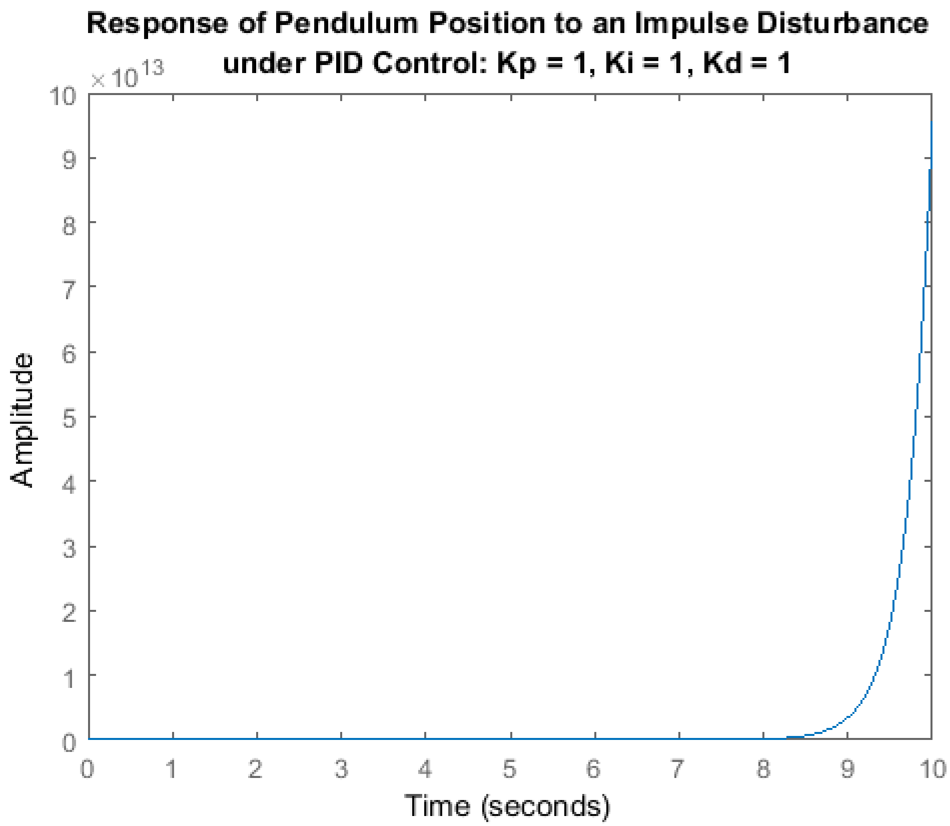
PID CONTROL

This closed-loop transfer function can be modeled in MATLAB by copying the following code to the end of previous code. Specifically, we define our controller using the pid object within MATLAB. We then use the feedback command to generate the closed-loop transfer function $T(s)$ as depicted in the figure above where the disturbance force F is the input and the deviation of the pendulum angle from the vertical ϕ is the output.

```
Kp = 1;

Ki = 1;

Kd = 1;

C = pid(Kp,Ki,Kd);

T = feedback(P_pend,C);
```

Now we can begin to tune our controller. First let's examine the response of the closed-loop system to an impulse disturbance for this initial set of control gains. The following code is added to the existing code. The response plot shown below is generated.

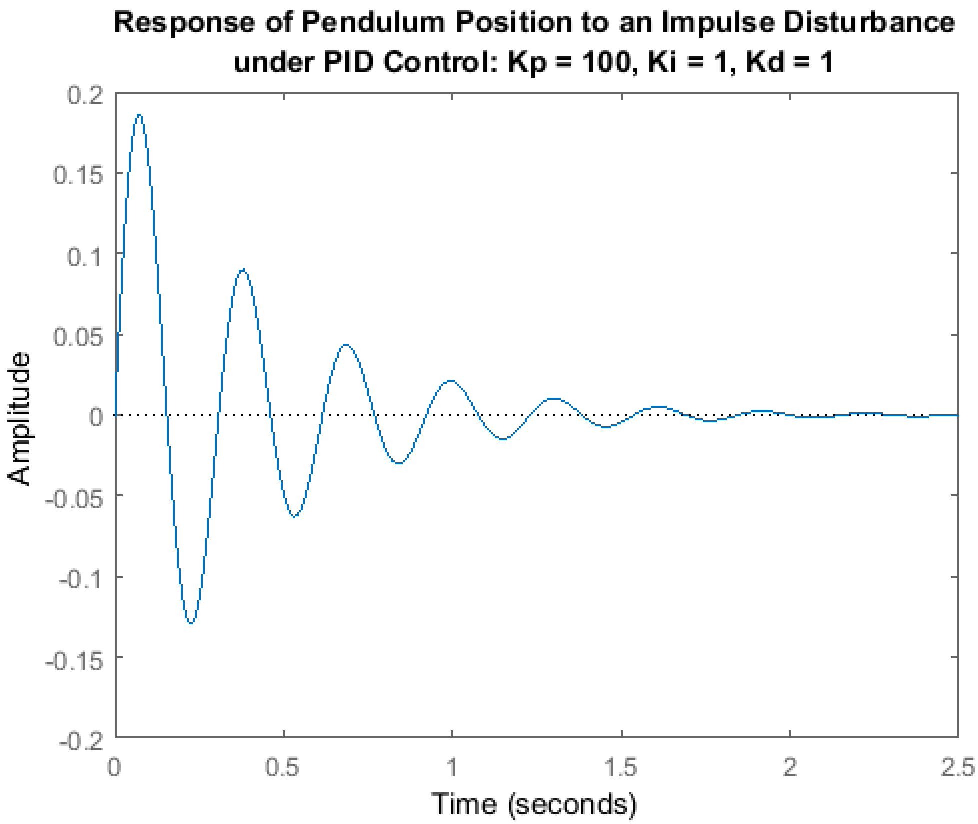
```
t=0:0.01:10;

impulse(T,t)

title({'Response of Pendulum Position to an Impulse Disturbance';'under PID Control: Kp = 1, Ki = 1, Kd = 1'});
```



This response is still not stable. The response is modified by increasing the proportional gain. Increase the K_p variable to see what effect it has on the response. If code is modified to the following where K_p = 100 and run in the command window, The response plot shown below is obtained.

```
Kp = 100;

Ki = 1;

Kd = 1;

C = pid(Kp,Ki,Kd);

T = feedback(P_pend,C);

t=0:0.01:10;
```

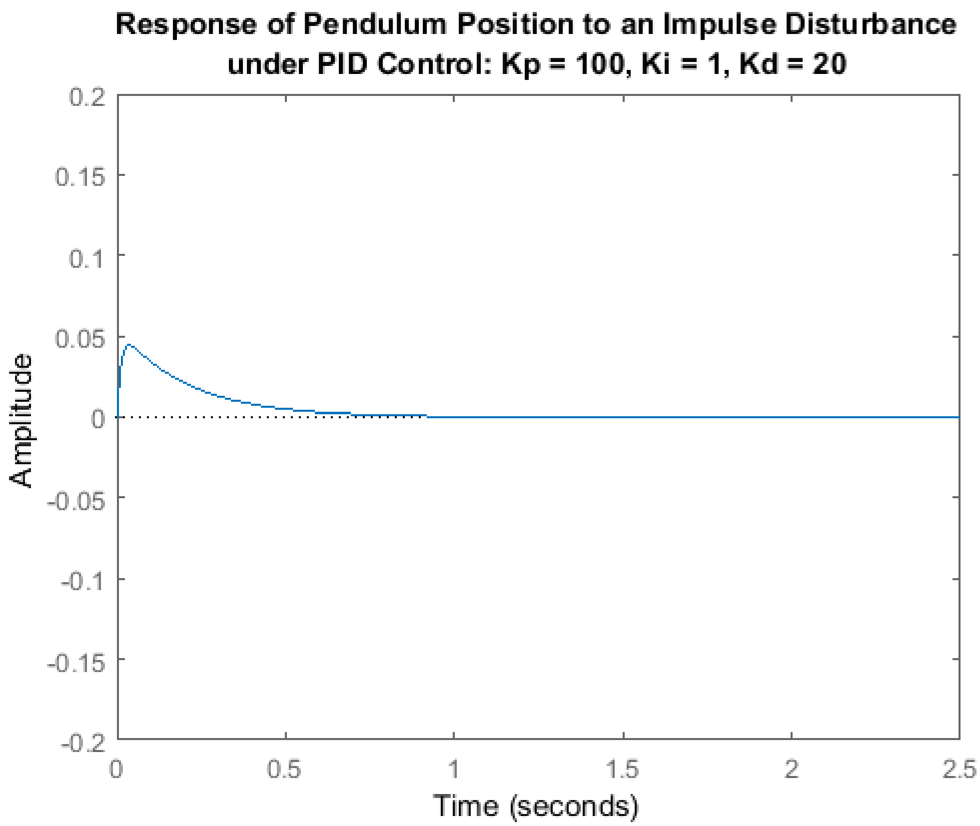
```
impulse(T,t)

axis([0, 2.5, -0.2, 0.2]);

title({'Response of Pendulum Position to an Impulse Disturbance';'under PID Control: Kp = 100, Ki = 1, Kd = 1'});
```



The important characteristics of the response are identified. Specifically, the settling time of the response is determined to be 1.64 seconds, which is less than the requirement of 5 seconds. Since the steady-state error approaches zero in a sufficiently fast manner, no additional integral action is needed. The peak response, however, is larger than the requirement of 0.05 radians. The overshoot often can be reduced by increasing the amount of derivative control. After some trial and error it is found that a derivative gain of 20 provides a satisfactory response. Modifying the code as follows and re-running should produce the response plot shown below

```
Kp = 100;

Ki = 1;

Kd = 20;

C = pid(Kp,Ki,Kd);
```

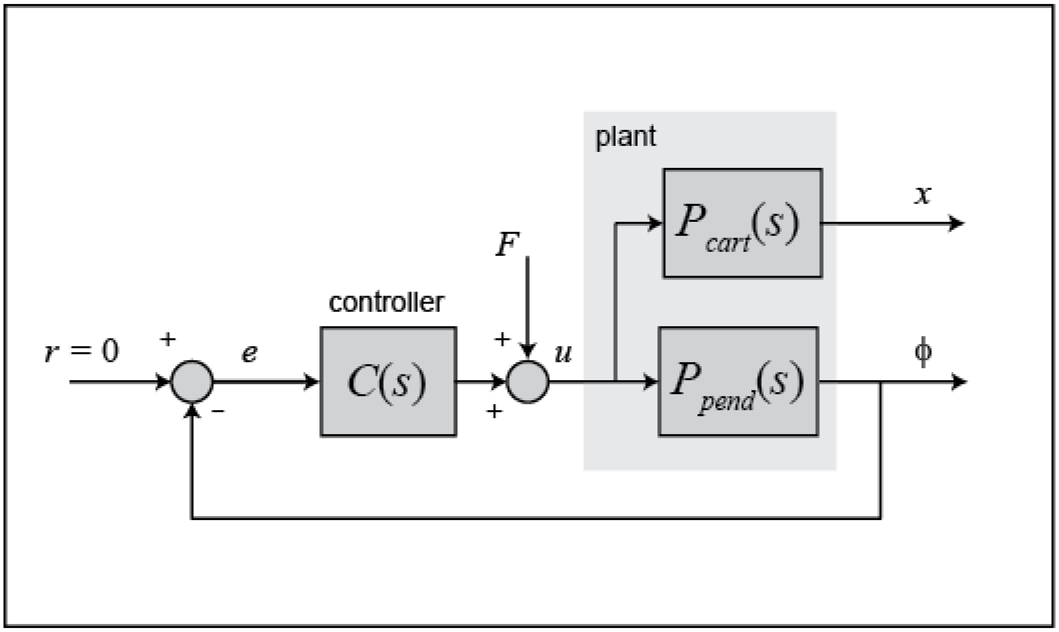
```
T = feedback(P_pend,C);

t=0:0.01:10;

impulse(T,t)

axis([0, 2.5, -0.2, 0.2]);

title({'Response of Pendulum Position to an Impulse Disturbance';'under PID Control: Kp
= 100, Ki = 1, Kd = 20'});
```



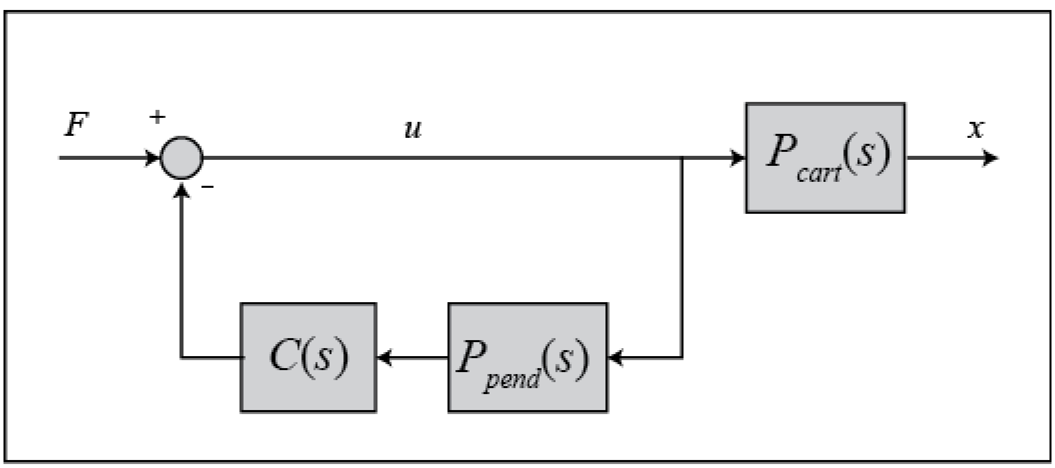
As can be seen, the overshoot has been reduced so that the pendulum does not move more than 0.05 radians away from the vertical. Since all of the given design requirements have been met, no further iteration is needed.

RESPONSE OF CART'S POSITION

The block representing the response of the cart's position x was not included in the previous diagram because that variable is not being controlled. To see what is happening to the cart's position when the controller for the pendulum's angle ϕ is in place we need to consider the full system block diagram as shown in the following figure.



Rearranging, we get the following block diagram.



In the above, the block $C(s)$ is the controller designed for maintaining the pendulum vertical. The closed-loop transfer function $T_2(s)$ from an input force applied to the cart to an output of cart position is, therefore, given by the following.

$$T_2(s) = \frac{X(s)}{F(s)} = \frac{P_{cart}(s)}{1 + P_{pend}(s)C(s)} \tag{4}$$

The transfer function for $P_{cart}(s)$ is defined as follows.

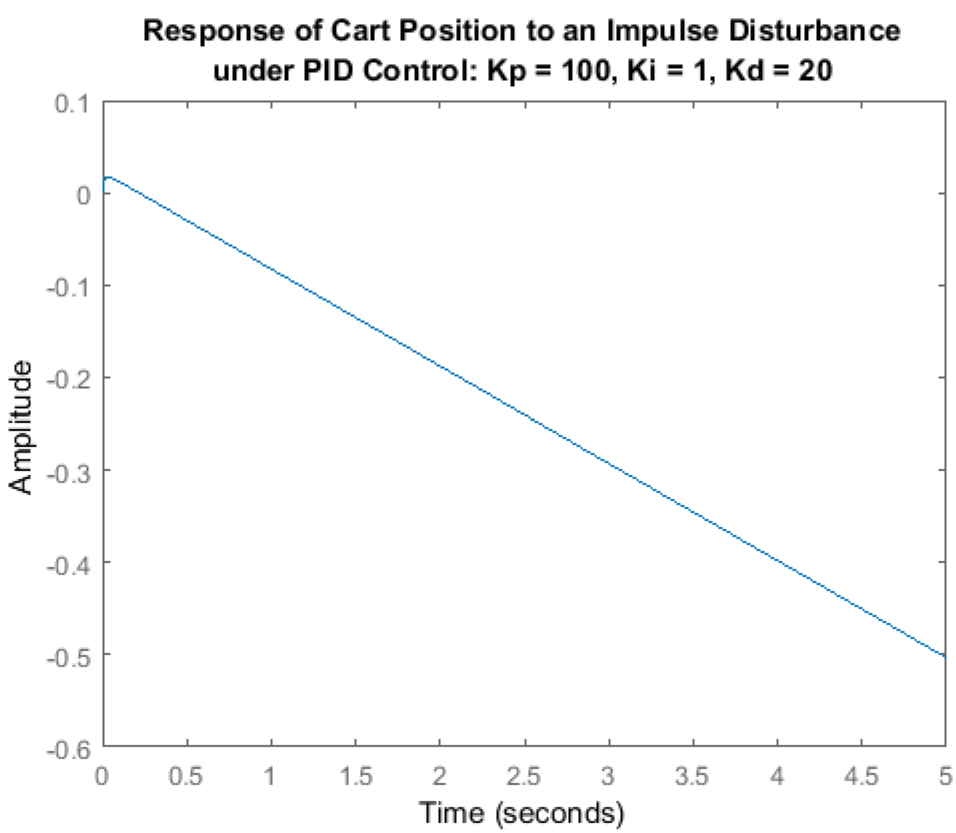
(5) $$P_{cart}(s) = \frac{X(s)}{U(s)} = \frac{\frac{(I+ml^2)s^2-gml}{q}}{s^4+\frac{b(I+ml^2)}{q}s^3-\frac{(M+m)mgl}{q}s^2-\frac{bmgl}{q}s} \qquad [\frac{m}{N}]$$

where,

(6) $$q = [(M+m)(I+ml^2)-(ml)^2]$$

Adding the following commands to previous code (presuming $P_{pend}(s)$ and $C(s)$ are still defined) will generate the response of the cart's position to the same impulsive disturbance we have been considering.

```
P_cart = (((I+m*l^2)/q)*s^2 - (m*g*l/q))/(s^4 + (b*(I + m*l^2))*s^3/q - ((M + m)*m*g*l)*s^2/q - b*m*g*l*s/q);

T2 = feedback(1,P_pend*C)*P_cart;

t = 0:0.01:5;

impulse(T2, t);

title({'Response of Cart Position to an Impulse Disturbance';'under PID Control: Kp = 100, Ki = 1, Kd = 20'});
```

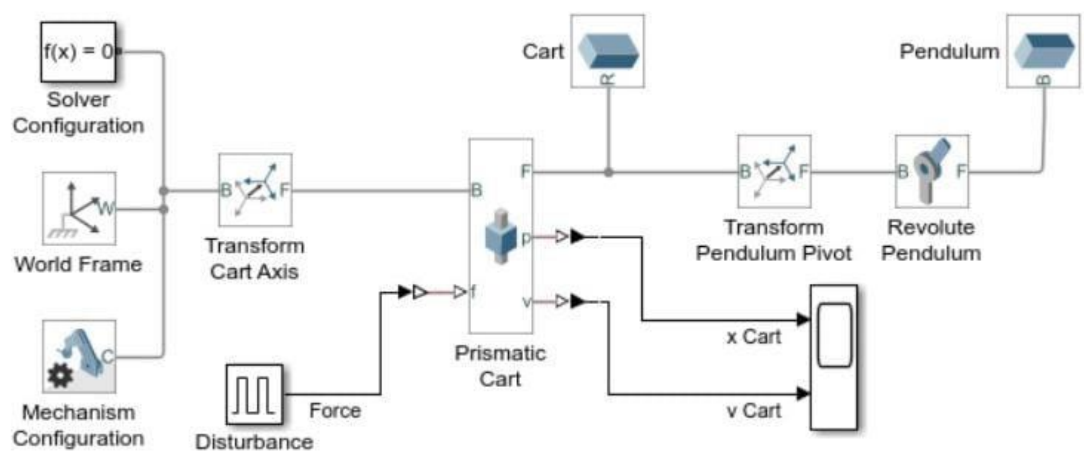


As can be seen, the cart moves in the negative direction with approximately constant velocity. Therefore, the PID controller stabilizes the angle of the pendulum,

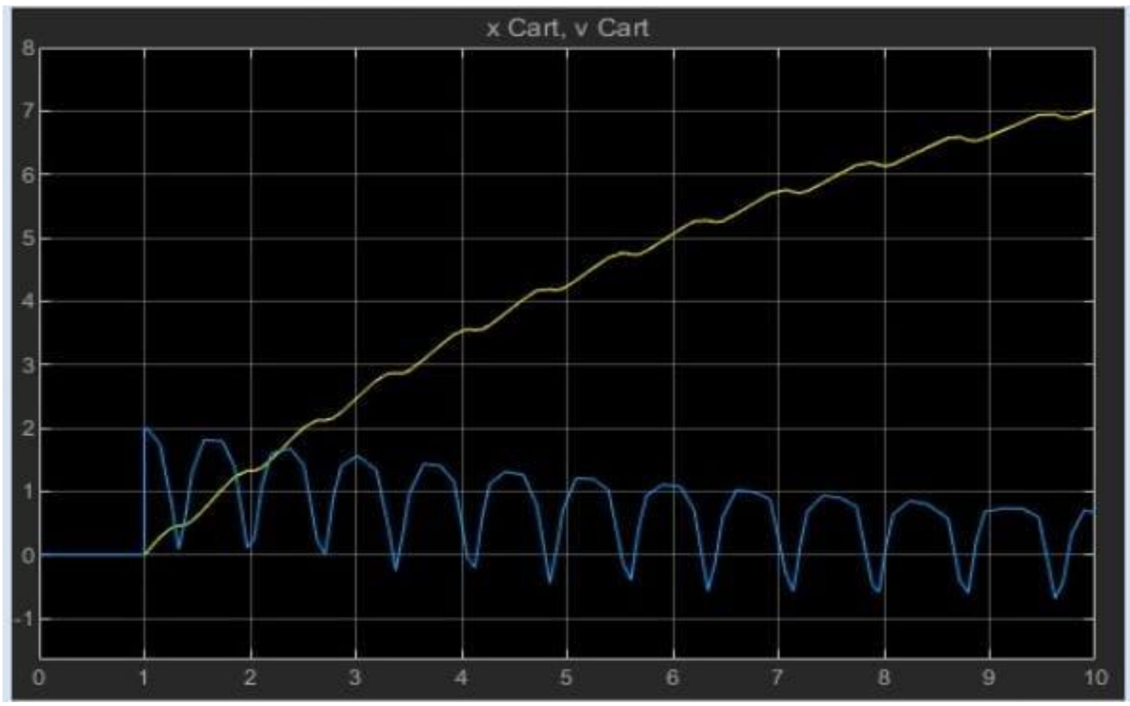
Now, we will build a Simscape model for the above design criterion and simulate it using Simulink.

PENDULUM SUBSYSTEM AND CONNECTING CART TO PENDULUM

The cart pendulum simulink model without any control mechanism and it's response is as shown.

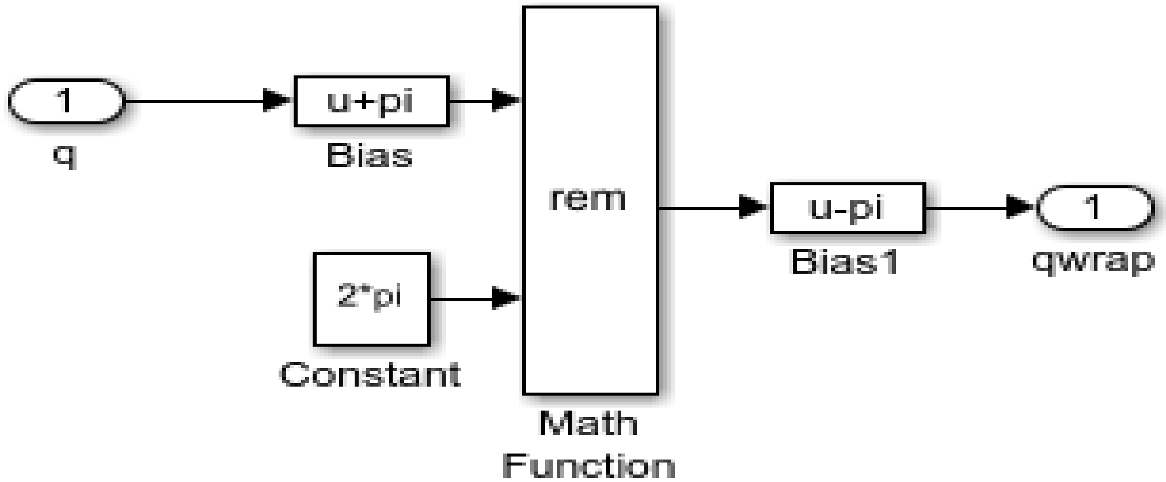


The following plot is generated, where one can see that the addition of the pendulum alters the cart behavior both in its distance traveled as well as its velocity.

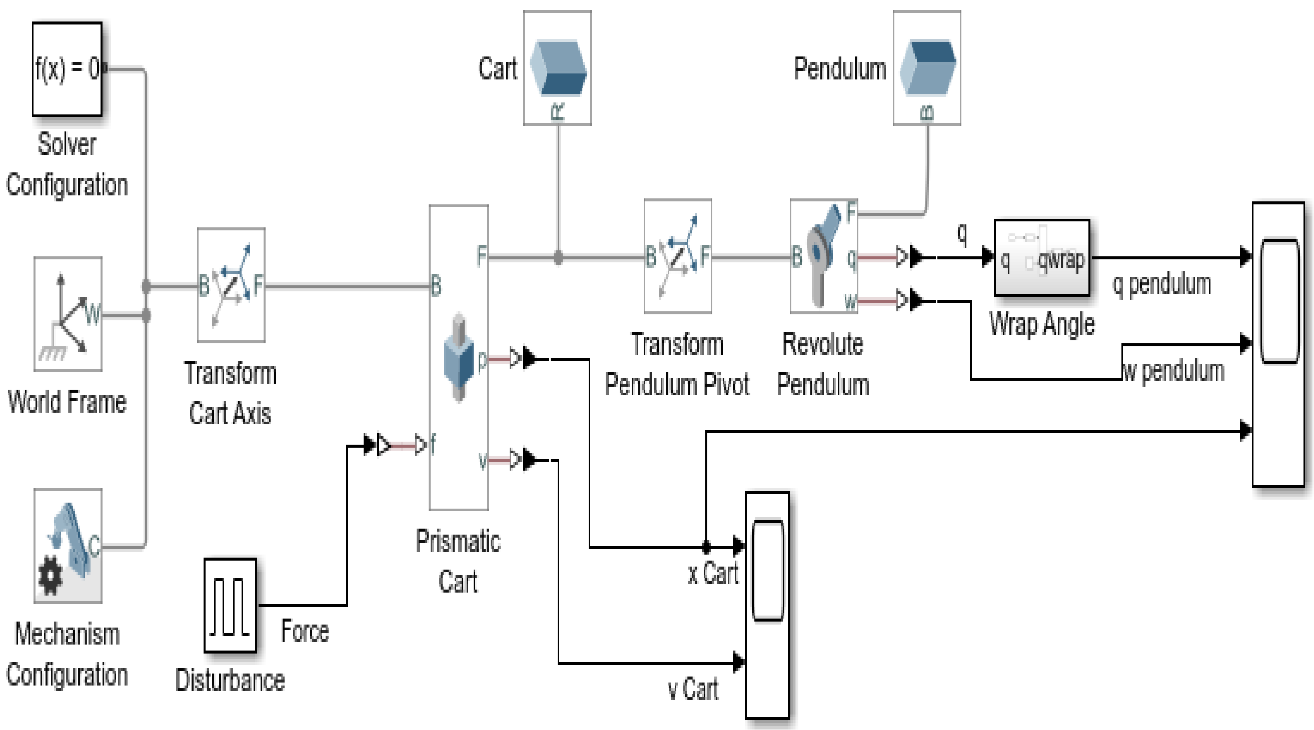


SELECTING OUTPUTS FOR CONTROLLER AND ANGLE CONVERSION

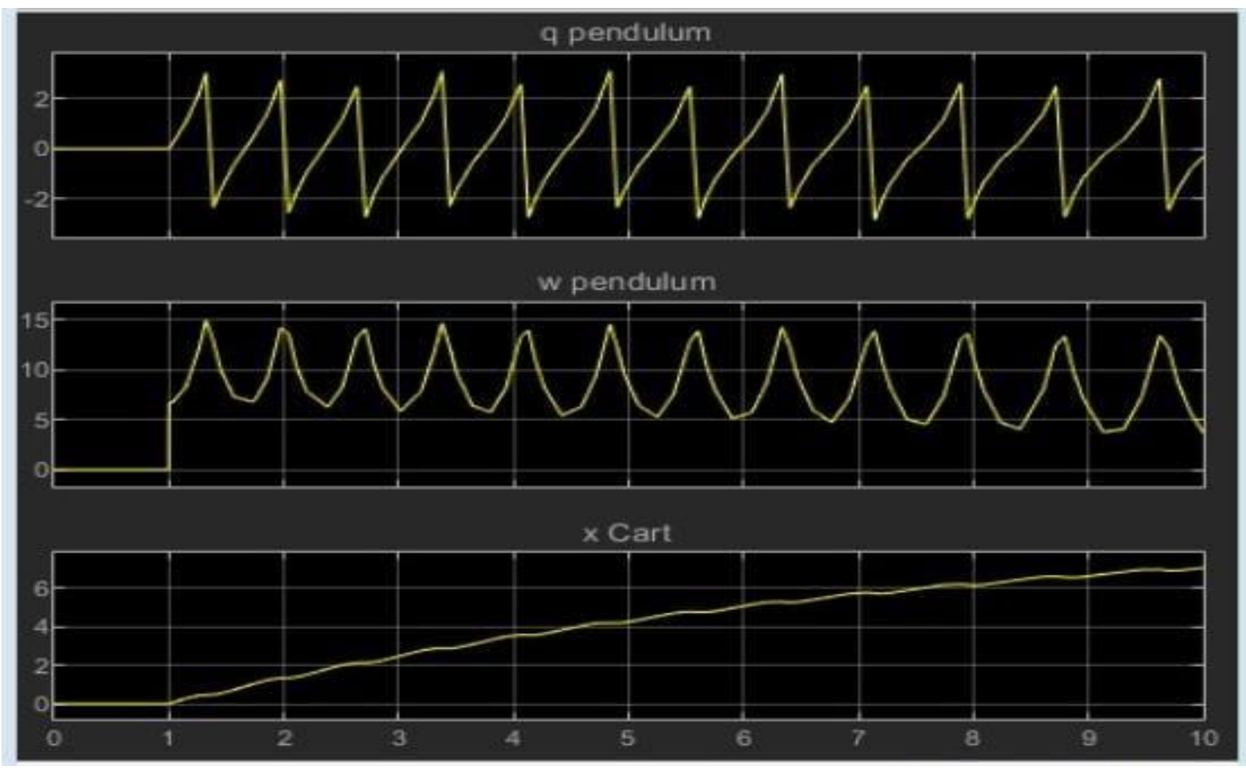
The resulting subsystem for wrapping the angle is shown.



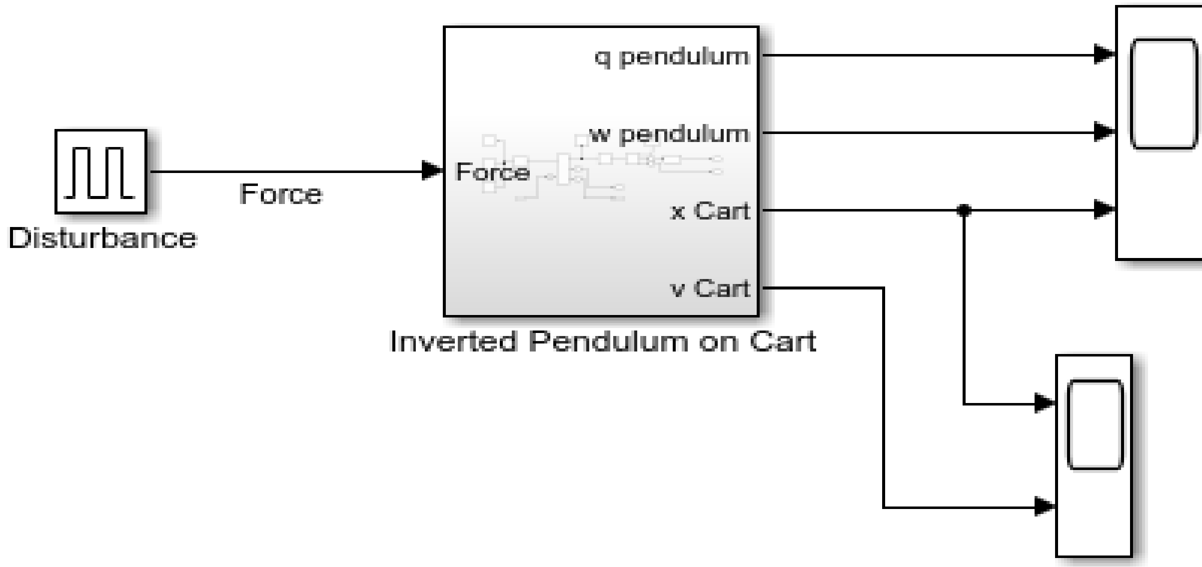
The resulting model should appear as follows.



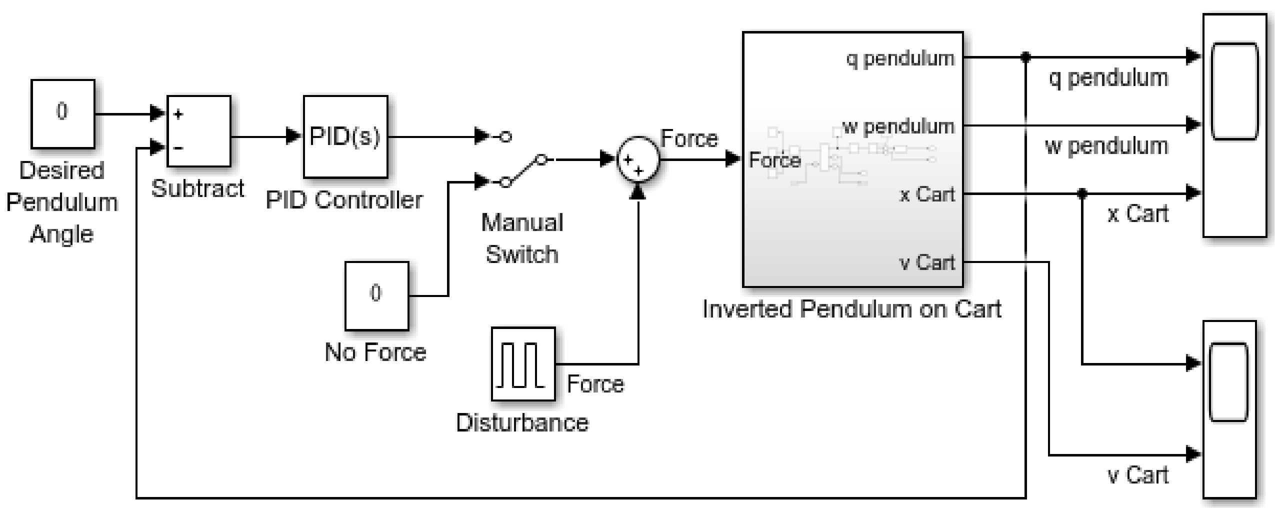
The following plots are generated. The motion of the cart is the same as before, but now we can see the motion of the pendulum. The associated animation shown below is also generated.



<u>SUBSYTEM FOR PENDULUM AND CART</u>



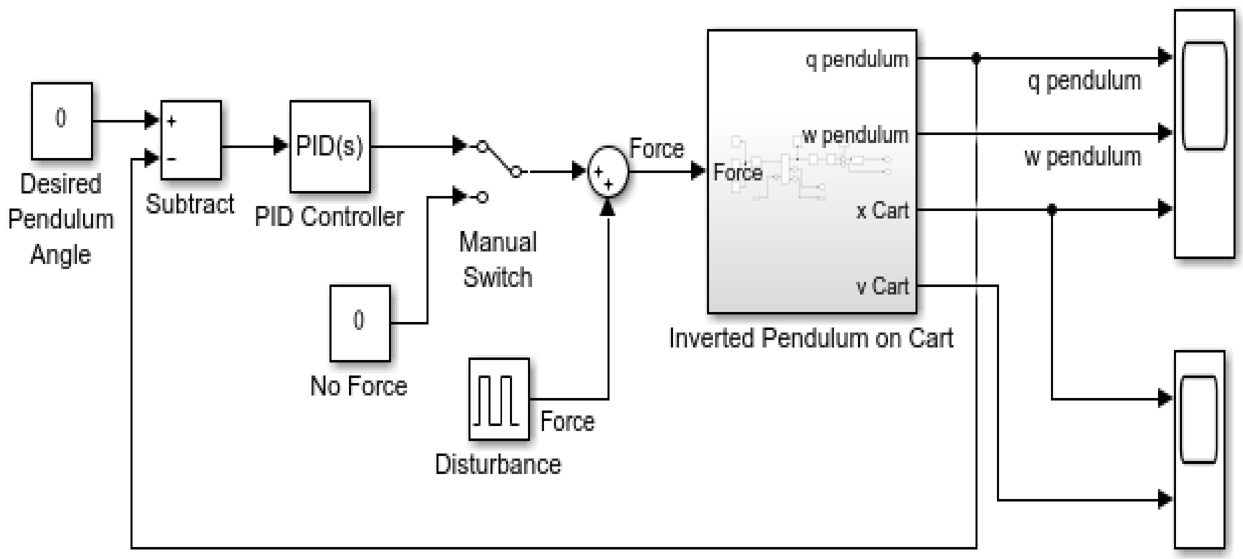
CLOSED-LOOP SETUP



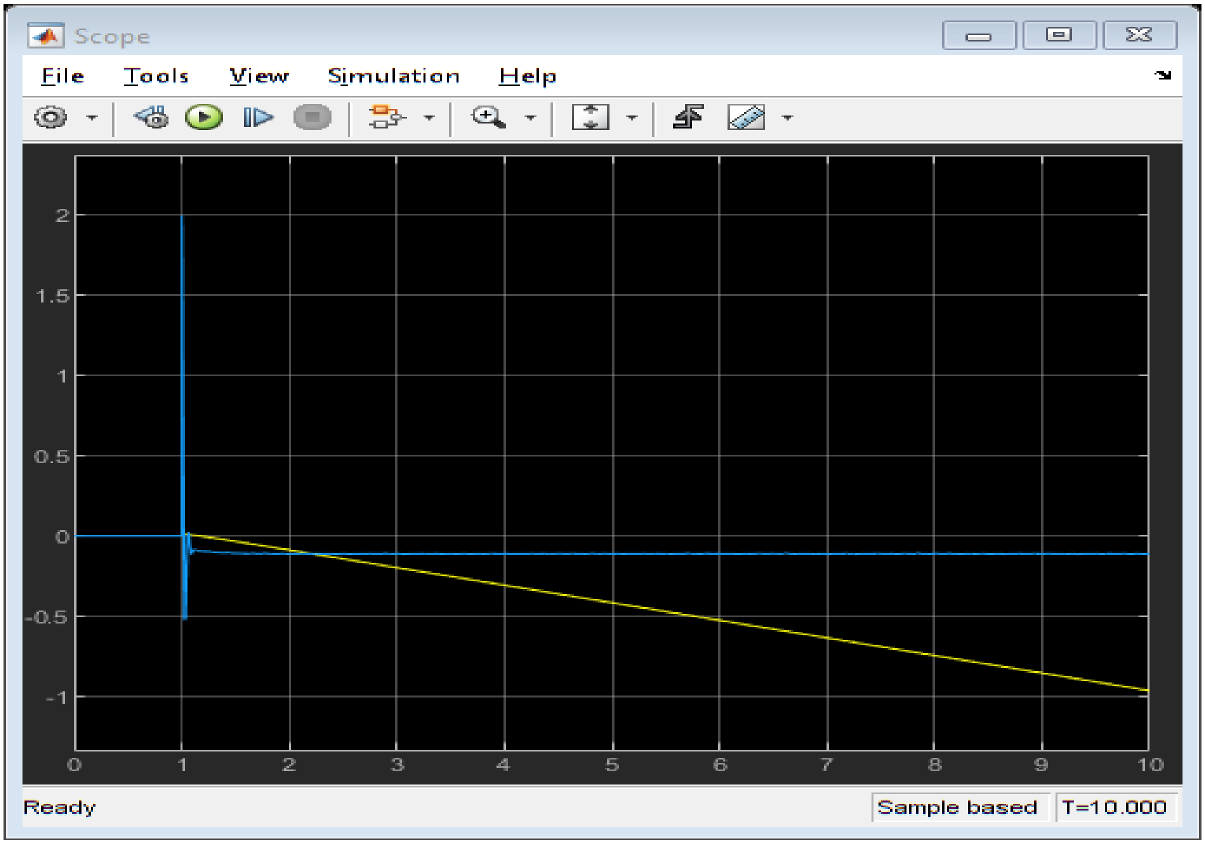
The output of the simulation is unchanged from prior results when in open-loop mode.

CONTROLLER IMPLEMENTATION

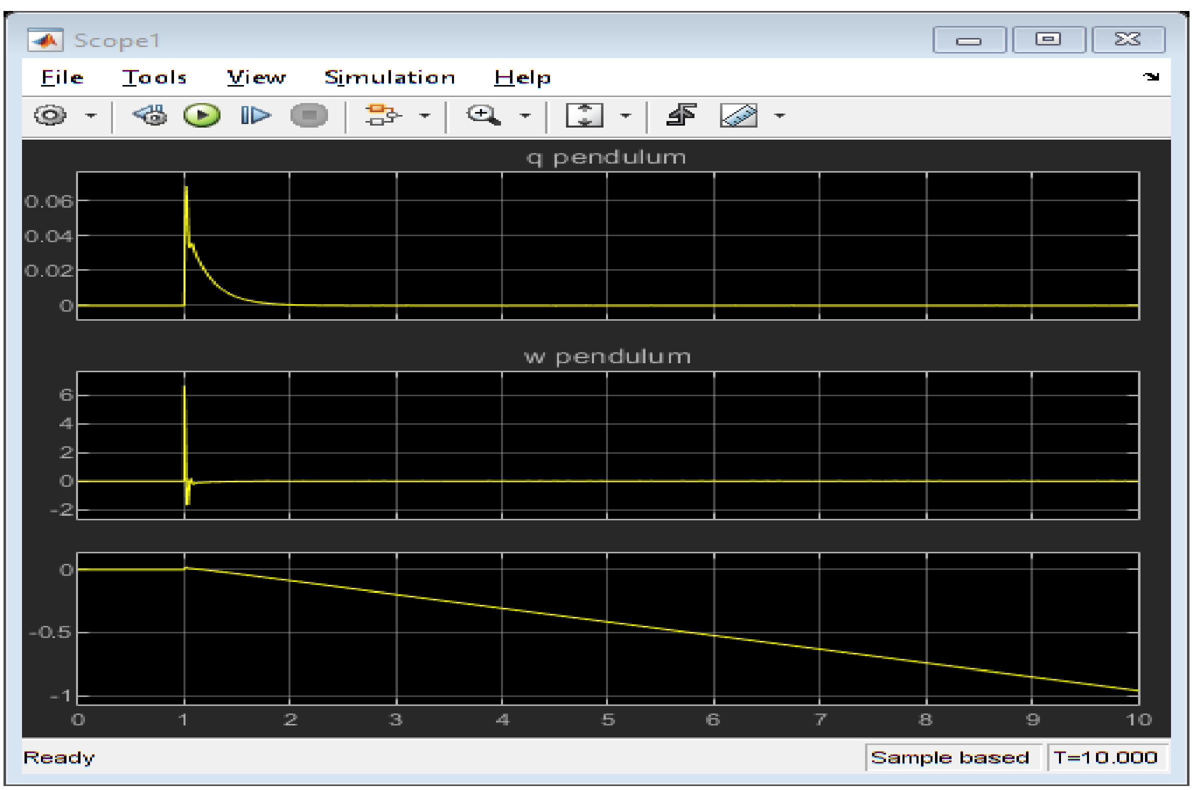
- Setting parameter **Proportional (P)** to "100"
- Setting parameter **Integral (I)** to "1"
- Setting parameter **Derivative (D)** to "20"



Running a simulation produces the following two plots show the controlled response of the system. After the initial impact the controller was able to quickly bring down the pendulum angle to zero and the pendulum velocity is also zero. The cart moves slowly and with a constant velocity in the negative X direction to keep the pendulum balanced.



Scope
File Tools View Simulation Help
2
1.5
1
0.5
0
-0.5
-1
0 1 2 3 4 5 6 7 8 9 10
Ready
Sample based
T=10.000



Scope1
File Tools View Simulation Help
q pendulum
0.06
0.04
0.02
0
w pendulum
6
4
2
0
-2
0
-0.5
-1
0 1 2 3 4 5 6 7 8 9 10
Ready
Sample based
T=10.000