The cart on a track

AAE 364L

This experiment consists of a cart with mass M on a one dimensional track, driven by a force from a servo motor. The position of the cart is denoted by $y(t)$, and the voltage to the servo motor is denoted by $v(t)$. A simplified equation of motion for this system is given by

$$m\ddot{y}(t) + c\dot{y}(t) = \gamma v(t). \tag{0.1}$$

Here m and c are the constants determined by

$$m = M + M_J \quad \text{and} \quad c = B_{eq} + B_{emf}. \tag{0.2}$$

The mass of the cart is M and M_J represents the effective mass added to the system due to the moment of inertia of the motor. The total damping $c = B_{eq} + B_{emf}$ is the sum of two damping terms, B_{eq} is the damping of the cart due to viscous friction, and B_{emf} is the damping of the cart due to the back electromagnetic force (EMF) of the motor. The constant γ is the gain from the voltage applied to the motor to the force on the cart. The constants M_J, B_{emf} and γ are determined by

$$\begin{aligned} M_J &= \frac{\eta_g K_g^2 J_m}{r_{mp}^2} \\ B_{emf} &= \frac{\eta_g K_g^2 \eta_m K_t K_m}{R_m r_{mp}^2} \\ \gamma &= \frac{\eta_g K_g \eta_m K_t}{R_m r_{mp}}. \end{aligned} \tag{0.3}$$

The values and definitions of these constants are given in the Appendix. Notice that the only unknown parameter is the damping B_{eq} due to the viscous friction between the cart and the track. Finally, in our experiment we assume that the mass of the cart contains the mass weight. So by consulting the table in the Appendix $M = M_c + M_w = 0.94kg$.

By taking the Laplace transform with zero initial conditions in (0.1), the open loop transfer function G from the input voltage v to the output y is given by

$$G(s) = \frac{Y(s)}{V(s)} = \frac{\gamma}{ms^2 + cs}. \tag{0.4}$$

Notice that this system has a pole at $s = 0$. So the open loop system is only marginally stable.

1 Part (i): The open loop model

Compute γ, m, and B_{emf}, using the parameters given by the table in the Appendix. In this part of the Lab, we will determine the viscous friction B_{eq} experimentally.

To compute B_{eq}, unplug the motor to eliminate the damping from the back EMF, then manually tap the cart so that the initial condition is $y(0) = 0$ and $\dot{y}(0) = \dot{y}_0$. This is equivalent to giving an impulse to the cart. Then record the output $y(t)$ from this experiment and save this plot in MATLAB as a Mat file. This output $y(t)$ is simply the impulse response. In this case, $B_{emf} = 0$ and the voltage $v(t) = 0$. So the equation of motion in (0.1) reduces to

$$m\ddot{y} + B_{eq}\dot{y} = 0 \qquad (y(0) = 0 \text{ and } \dot{y}(0) = \dot{y}_0). \tag{1.1}$$

The solution to this differential equation is given by

$$y(t) = \frac{\dot{y}_0 m}{B_{eq}}\,(1 - e^{-\frac{B_{eq}t}{m}})\,. \tag{1.2}$$

By observing the plot of this $y(t)$, estimate the ratio B_{eq}/m, and the steady state value $y(\infty) = \dot{y}_0 m/B_{eq}$. Using your estimate of B_{eq}/m and the steady state value, plot your theoretical $y(t)$ on the same graph as the experimental $y(t)$. Finally, it is noted that when the experiment starts the computer always sets the starting position of the cart at zero. A positive distance is to the right and negative distance to the left.

To solve for B_{eq}, first observe that (1.2) is in exponential form. By rearranging and taking logarithm of (1.2), we arrive at

$$-m\ln\left(1 - y(t)/\frac{\dot{y}_0 m}{B_{eq}}\right) = B_{eq}\,t\,.$$

If the ratio $\dot{y}_0 m/B_{eq}$ is known, then the previous equation is the linear equation $\hat{y} = a\hat{x}$ where

$$\hat{y} = -m\ln\left(1 - y(t)/\frac{\dot{y}_0 m}{B_{eq}}\right), \quad a = B_{eq}, \quad \text{and} \quad \hat{x} = t\,.$$

By observing the slope a, the parameter B_{eq} can be computed.

1.1 The Lab steps to experimentally determine B_{eq}

(i) Open MATLAB and change the directory to *Desktop\aae364L\labcart\section#*

(ii) Open up labcartbeq in Simulink and select `Quarc - Build`.

(iii) Unplug the black cable to the motor to eliminate the back EMF. Put the cart at the left end of the track.

(iv) In the Simulink window, select `Tools - External Mode Control Panel - Signal & Triggering`, and change the duration to 15,000.

(v) Double-click the cart scope and select Parameters. Un-check the *Limit to 5000 data points* check-box. Repeat for rod scope.

(vi) In the Simulink window, set the simulation time to 29 seconds.

(vii) Click `Quarc - Build`. Select `Simulation - Connect to Target`, and `Quarc - Start`.

(viii) Tap the cart gently to simulate $y(0) = 0$ and $\dot{y}(0)$. Once the cart stops hit the stop.

(ix) Save the cart position in MATLAB, that is, go to: `File - Save - Save as Mat file`. Take this data file with you.

(x) Plug the black cable back in.

1.2 In your lab report include the following under Part (i):

(a) Hand in your values for m, γ, B_{emf}, B_{eq} and c.

(b) Hand in your experimental plot of y and y from (1.2) on the same graph.

2 Part (ii): Model validation and saturation

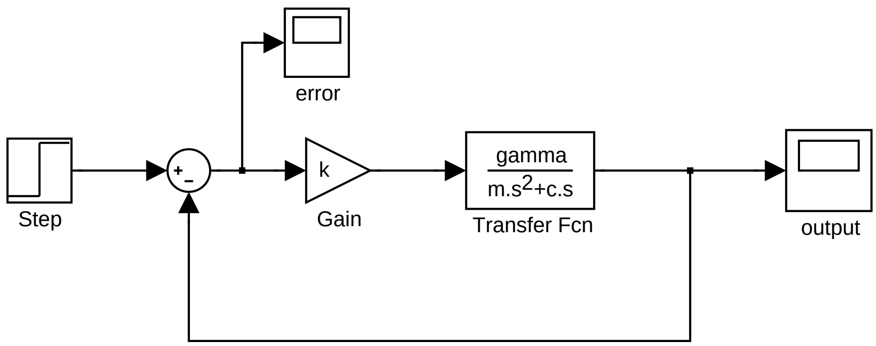

Figure 1: The simple closed loop model.

Recall that the open loop transfer function is given by

$$G(s) = \frac{Y(s)}{V(s)} = \frac{\gamma}{ms^2 + cs}.$$

Consider the closed loop system in Figure 1 with gain k. The error e is measured by the scope labeled error and output y is measured by the scope labeled output. The transfer function from the reference signal r to the position y is given by

$$\frac{Y(s)}{R(s)} = \frac{k\gamma}{ms^2 + cs + k\gamma}\,. \tag{2.1}$$

The transfer function from the reference signal r to the error e to is determined by

$$\frac{E(s)}{R(s)} = \frac{ms^2 + cs}{ms^2 + cs + k\gamma}\,. \tag{2.2}$$

Recall that the roots of the quadratic polynomial $s^2 + as + b$ are all in the open left half plane $\{s : \Re s < 0\}$ if and only if $a > 0$ and $b > 0$. Since m, c and γ are positive, both of these transfer functions in (2.1) and (2.2) are stable for all $k > 0$.

Now assume that the reference signal r is a step of magnitude r_0. In other words, assume that $r(t) = r_0$ is constant for all t. Then the steady state output $y_{ss}(t) = y(\infty) = r_0$ and steady state error $e_{ss}(t) = e(\infty)$ is zero. To be precise,

$$\begin{aligned} y(\infty) = \lim_{t\to\infty} y(t) &= r_0 \\ e(\infty) = \lim_{t\to\infty} e(t) &= 0\,. \end{aligned} \tag{2.3}$$

Run the experiment for the closed loop system with the reference signal $r(t) = r_0 = 0.4\,m$ and $k = 50\,V/m$. Record the output $y(t)$ of the experiment in MATLAB for at least three seconds. Take this Mat file with you. Later using the Simulink diagram in Figure 1, simulate the system with open loop transfer function $G(s)$ you found with $r_0 = 0.4\,m$ and $k = 50\,V/m$.

On the same graph plot the output y from the experiment and the output from Simulink for three seconds. Hand in this plot. Notice the model and experiment do not match. The difference is due to a saturation on the motor. Our model did not include the saturation on the voltage. To present a more accurate model, we must include this saturation in our model.

A saturation $f(u)$ is a nonlinear function which limits the input to a maximum and minimum value. A saturation with a maximum of β and minimum of α is the nonlinear function defined by

$$\begin{aligned} f(u) &= \beta \quad && \text{if } u > \beta \\ &= u && \text{if } \alpha \leq u \leq \beta \\ &= \alpha && \text{if } u < \alpha. \end{aligned}$$

It is emphasized that the saturation $f(u)$ is linear in the region $\alpha \leq u \leq \beta$. Finally, it is noted that in many applications $\beta > 0$ and $\alpha = -\beta$.

The reason that the model and actual system differ is that the actual system has a saturation on the voltage. This voltage saturation is added in to protect the motor. The actual system has a saturation of 6 volts on the motor, that is, the a nonlinear function of the form

$$\begin{aligned} f(v) &= 6 \quad && \text{if } v > 6 \text{ volts} \\ &= v && \text{if } |v| \leq 6 \text{ volts} \\ &= -6 && \text{if } v < -6 \text{ volts}. \end{aligned} \tag{2.4}$$

Now add saturation to your Simulink model; see Figure 2. As before, $r(t) = r_0 = 0.4\,m$ and $k = 50\,V/m$. Now using the Simulink diagram in Figure 2, simulate the system with open loop transfer function $G(s)$ you found with $r_0 = 0.4\,m$ and $k = 50\,V/m$. On the same graph plot the experimental output y with the Simulink simulation in Figure 2 of y for three seconds. You may even want to adjust your B_{eq} to obtain a better model. In other words, you can adjust the viscous friction B_{eq} in your Simulink model to find a better fit of the experimental data. This new estimate of B_{eq} may be more accurate for your future simulations.

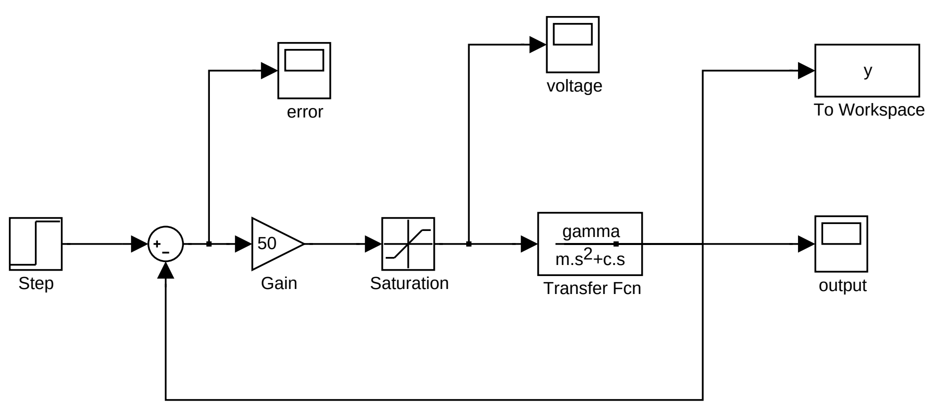

Figure 2: The closed loop model with saturation.

2.1 The Lab steps to model validation.

(i) In the same directory, open up model labcartsat. In the Simulink model window, click `Quarc - Build`.

(ii) Put the cart at the left end of the track.

(iii) In the Simulink window, select `Tools - External Mode Control Panel - Signal & Triggering`, and change the duration to 15,000.

(iv) Double-click the cart scope and select Parameters. Un-check the *Limit to 5000 data points* check-box. Repeat for voltage scope.

(v) In the Simulink window, set the simulation time to 10 seconds.

(vi) Click `Quarc - Build`. Select `Simulation - Connect to Target`, and `Quarc - Start`.

(vii) Once the cart stops hit the stop button.

(viii) Save the cart position y and voltage v in MATLAB, that is, go to `File - Save - Save as Mat file`. Take these Mat files with you.

2.2 In your lab report include the following under Part (ii):

(a) In Simulink compute the input voltage v using Figure 1 (without saturation), and compute the input voltage v using Figure 2 (with saturation). Hand in the experimental plot of v and these two Simulink plots of v on the same graph for three seconds. Are these plots different, the same?

(b) In Simulink compute the output y using Figure 1 (without saturation), and compute the output y using Figure 2 (with saturation). Hand in the experimental plot of y and these two Simulink plots of y on the same graph for three seconds. Are these plots different, the same? What is the effect of saturation on y?

(c) Adjust parameter in your Simulink model so that the simulation is close to the experimental result. Hand in your new estimate of the viscous damping B_{eq}.

3 Part (iii): The effect of Coulomb friction

The force due to Coulomb friction is the nonlinear function of velocity defined by

$$F_c(\dot{y}) = f_c \,\mathrm{sign}(\dot{y}). \tag{3.1}$$

Here f_c is a constant positive scalar. To be precise,

$$\begin{aligned} F_c(\dot{y}) &= \quad f_c \qquad \text{if } \dot{y} > 0 \\ &= -f_c \qquad \text{if } \dot{y} < 0\,. \end{aligned} \tag{3.2}$$

The value of $F_c(0)$ is not specified and can be any force between $-f_c$ and f_c. The equation of motion including the effect of Coulomb friction is given by

$$m\ddot{y} + c\dot{y} + F_c(\dot{y}) = \gamma v(t)\,. \tag{3.3}$$

If the initial conditions $y(0) = 0$ and $\dot{y}(0) = 0$ are both zeros and $v(t) = v_0$ is a constant, then the cart will not move until the voltage v_0 is strong enough to overcome the Coulomb friction force f_c. In other words, if $0 \le \gamma v_0 \le f_c$, then the Coulomb friction force $F_c(\dot{y}) = F_c(0) = \gamma v_0$. In this case, $m\ddot{y} + c\dot{y} + \gamma v_0 = \gamma v_0$. In other words, $m\ddot{y} + c\dot{y} = 0$ with all the initial condition zero, and thus, $y(t) = 0$. On the other hand, once $\gamma v_0 > f_c$, then $m\ddot{y} + c\dot{y} = \gamma v_0 - f_c \neq 0$, and hence, $y(t)$ is nonzero. To measure the Coulomb friction one simply increases the voltage until the cart starts to move, that is, find the smallest constant v_0 such that the cart starts to move, or equivalently, $y(t)$ is nonzero. Then the Coulomb friction $f_c = \gamma v_0$.

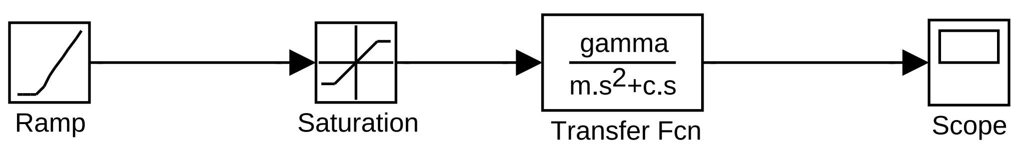

Figure 3: The model to measure Coulomb friction.

To measure the Coulomb friction in our system, let the voltage v be the ramp input given by $v = Kt$ where $K = 0.1$ V/s. Then run the experiment and graph the position of the cart in MATLAB; see Figure 3. (We have included saturation in this Simulink diagram. However, the voltage is too small for the saturation to have any effect.) Find the time t_0 when the cart start moving. The corresponding voltage v which makes the cart move is $v(t_0) = 0.1t_0$. The Coulomb friction constant f_c is computed by

$$f_c = \gamma v(t_0) = 0.1\,\gamma t_0. \tag{3.4}$$

The closed loop system with Coulomb friction. To see the effect of Coulomb friction consider the closed loop system in Figure 1. Set $k = 10$ V/m and $r(t) = 0.4$ m for all t. Now run the experiment and record the error $e(t)$. If there is no Coulomb friction, then the steady state error $e_{ss}(t) = e(\infty)$ is zero; see (2.3). However, due to Coulomb friction, the steady state error in the experiment is nonzero. The complete feedback system with saturation and Coulomb friction is given by the Simulink diagram in Figure 4. Finally, it is noted that with $k = 10$ V/m, the saturation block is in the linear region, that is, for $k = 10$ V/m, saturation does not play a role in our closed loop system

To see why the steady state error is nonzero, notice that the input voltage is given by $v = ke$, where e is the error defined by $e(t) = r(t) - y(t)$ and the reference signal is $r(t)$; see Figure 1. The equation of motion becomes

$$m\ddot{y} + c\dot{y} + F_c(\dot{y}) = \gamma k(r(t) - y(t))\,. \tag{3.5}$$

Assume $r(t) = r_0$ is a constant for all t. In this case, $e = r_0 - y$, $\dot{e} = -\dot{y}$ and $\ddot{e} = -\ddot{y}$. Using this in (3.5), we arrive at

$$m\ddot{e} + c\dot{e} + \gamma ke = F_c(\dot{y}). \tag{3.6}$$

Since m, c and $k\gamma$ are all strictly positive, $m\ddot{e} + c\dot{e} + \gamma ke = 0$ is stable. Moreover, one can show that the solution e to the differential equation in (3.6) converges to a constant value $e(\infty)$ as t tends to infinity. In other words, in steady state $\dot{e} = \ddot{e} = 0$, and $\dot{y} = 0$. Hence the steady state error $e(\infty)$ is given by $\gamma ke(\infty) = F_c(0)$, that is, $e(\infty) = F_c(0)/k\gamma$. Since $F_c(0)$ can be anywhere in $[-f_c, f_c]$, we see that

$$|e(\infty)| \leq \frac{f_c}{k\gamma}.$$

To complete this section hand in your value of the Coulomb friction f_c that you computed and the error $e(\infty)$ in the experiment. Finally, using the Simulink model in Figure 4, simulate your model with your value of f_c and find the steady state error $e(\infty)$ in your simulation. Compare this value to the actual error in the experiment.

In the sign function block in Simulink you must uncheck the enable zero crossing detection box to make the simulation work properly.

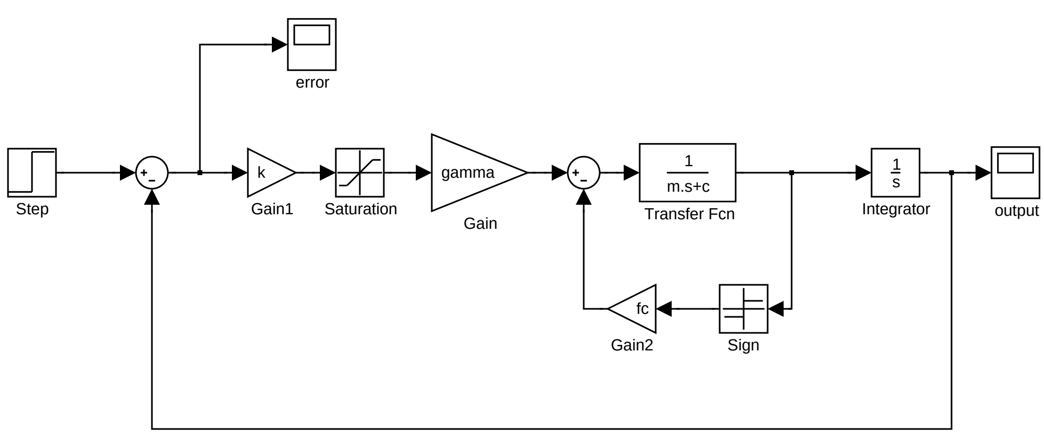

Figure 4: The feedback model with Coulomb friction and saturation.

3.1 The Lab steps to model Coulomb friction.

(i) In the same directory, open up labcartcol and select `Quarc - Build`.

(ii) Put the cart at the left end of the track.

(iii) In the Simulink window, select `Tools - External Mode Control Panel - Signal & Triggering`, and change the duration to 10,000.

(iv) Double-click the cart scope and select Parameters. Un-check the *Limit to 5000 data points* check-box.

(v) In the Simulink window, set the simulation time to 10 seconds.

(vi) Make sure that the ramp is set to 0.1.

(vii) Click `Quarc - Build`. Select `Simulation - Connect to Target`.

(viii) Select `Quarc - Start`. **Once the cart starts moving please hit the stop button. This system is unstable and the cart will slam into the wall.**

(ix) Save the cart position y in MATLAB as a Mat file. Take this data file with you.

- **For the second part of Coulomb friction**

(i) In the same directory, open up labcartcola in Simulink and select `Quarc - Build`.

(ii) Put the cart at the left end of the track.

(iii) In the Simulink window, select `Tools - External Mode Control Panel - Signal & Triggering`, and change the duration to 10,000.

(iv) Double-click the error scope and select Parameters. Un-check the *Limit to 5000 data points* check-box.

(v) In the Simulink window, set the simulation time to 10 seconds.

(vi) Make sure that the step is set to 0.4 and gain $k = 10$.

(vii) Click `Quarc - Build`. Select `Simulation - Connect to Target`. Select `Quarc - Start` .

(viii) Once the cart stops, hit the stop button.

(ix) Save the error e in MATLAB. Take this Mat file with you.

3.2 In your lab report include the following under Part (iii):

(a) Hand in the experimental plot of the position y due to the ramp $r(t) = 0.1t$.

(b) Hand your value f_c for the Coulomb friction force.

(c) Hand in the upper bound of $|e(\infty)|$.

(d) In Simulink compute the error e using Figure 4 with saturation and your value of f_c. Hand in the experimental plot of e and this Simulink plot of e on the same graph. Are these plots approximately the same?

4 Part (iv): Integral controller and Coulomb friction

In this section, we will see how one can use an integral controller to eliminate the effect of Coulomb friction. Recall that a proportional plus integral controller (PI) is a controller of the form

$$v(t) = ke(t) + k_i \int_0^t e(\sigma)d\sigma \quad \text{where} \quad e(t) = r(t) - y(t).$$

Here k is the proportional gain and k_i is the integral gain. Recall that the error $e(t) = r(t) - y(t)$. The Simulink diagram including saturation and the effect of Coulomb friction is given in Figure 5. In this analysis we will assume that the system is not in saturation, and thus, we will ignore this saturation in our analysis. For the PI controller the voltage is determined by

$$v(t) = ke(t) + k_i \int_0^t e(\sigma)d\sigma = k(r(t) - y(t)) + k_i \int_0^t (r(\sigma) - y(\sigma))d\sigma\,. \tag{4.1}$$

Substituting this into (3.3), the equation of motion is given by

$$m\ddot{y} + c\dot{y} + F_c(\dot{y}) = \gamma v = \gamma k(r(t) - y(t)) + \gamma k_i \int_0^t (r(\sigma) - y(\sigma))d\sigma\,. \tag{4.2}$$

As before, assume that $r(t) = r_0$ is constant for all t. In this case, $e = r_0 - y$, $\dot{e} = -\dot{y}$ and $\ddot{e} = -\ddot{y}$. Using this in (4.2), we obtain

$$m\ddot{e} + c\dot{e} + \gamma ke + \gamma k_i \int_0^t e(\sigma)d\sigma = F_c(\dot{y}). \tag{4.3}$$

By differentiating this equation and using the fact that $F_c(\dot{y})$ is almost everywhere a constant, we arrive at

$$me^{(3)} + c\ddot{e} + \gamma k\dot{e} + \gamma k_i e = 0. \tag{4.4}$$

As expected, $e^{(3)}$ is the third derivative of e with respect to time. The characteristic polynomial for this equation is given

$$ms^3 + cs^2 + \gamma ks + \gamma k_i = 0. \tag{4.5}$$

The differential equation in (4.4) is stable if and only if all the roots of the polynomial $ms^3 + cs^2 + \gamma ks + \gamma k_i$ are in the open left half plane $\{s : \Re s < 0\}$. In other words,

$$0 = e(\infty) = \lim_{t\to\infty} e(t)$$

if and only if k and k_i are chosen such that all the roots of $ms^3 + cs^2 + \gamma ks + \gamma k_i$ are in $\{s : \Re s < 0\}$. Another way to see this is to take the Laplace transform of (4.4), that is,

$$\left(ms^3 + cs^2 + \gamma ks + \gamma k_i\right) E(s) = as^2 + bs + d. \tag{4.6}$$

Here a, b and d are constants depending on the initial conditions for the error e. Dividing this equation by $ms^3 + cs^2 + \gamma ks + \gamma k_i$ yields

$$E(s) = \frac{as^2 + bs + d}{ms^3 + cs^2 + \gamma ks + \gamma k_i}. \tag{4.7}$$

Notice that the poles of $E(s)$ are the roots of $ms^3 + cs^2 + \gamma ks + \gamma k_i$. Therefore $e(\infty) = 0$ if and only if all the roots of $ms^3 + cs^2 + \gamma ks + \gamma k_i$ are in the open left half plane $\{s : \Re s < 0\}$.

In the sign function block in Simulink you must uncheck the enable zero crossing detection box to make the simulation work properly.

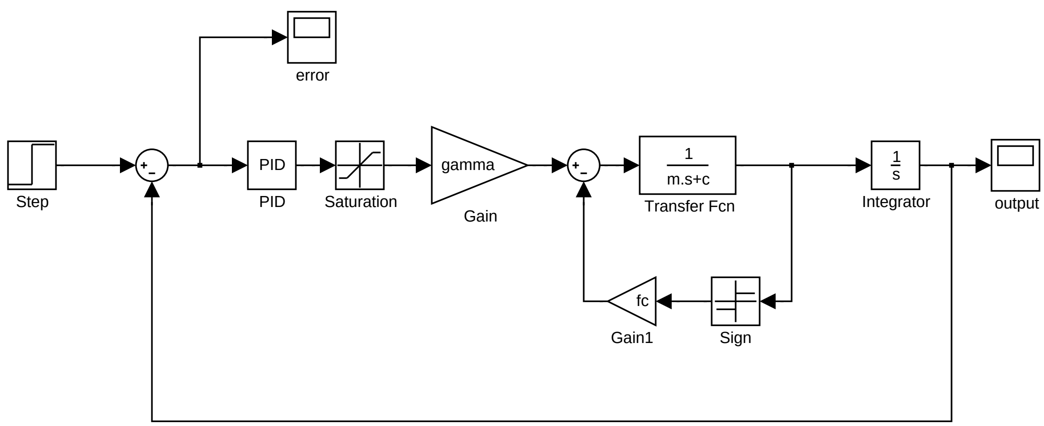

Figure 5: The feedback model with PI and Coulomb friction.

The above analysis shows that $e(\infty) = 0$ if and only if k and k_i are chosen such that all the roots of $ms^3 + cs^2 + \gamma ks + \gamma k_i$ are contained in $\{s : \Re s < 0\}$. Dividing (4.5) by $ms^3 + cs^2 + \gamma ks$, we see that

$$0 = ms^3 + cs^2 + \gamma ks + \gamma k_i \quad \text{if and only if} \quad 0 = 1 + \frac{\gamma k_i}{ms^3 + cs^2 + \gamma ks}. \tag{4.8}$$

Let $H(s)$ be the function defined by

$$H(s) = \frac{\gamma}{ms^3 + cs^2 + \gamma ks}.$$

Then the roots of $ms^3 + cs^2 + \gamma ks + \gamma k_i$ are the zeros of $1 + k_i H(s)$. For fixed k one can use the root locus to find the roots of $ms^3 + cs^2 + \gamma ks + \gamma k_i$. In particular, one can use the root locus to find all the values of k_i such that e is stable and $e(\infty) = 0$.

Run the experiment for $k = 10$ V/m and $k_i = 15$ V/ms and graph $e(t)$. Then $e(\infty)$ should be close to zero. In other words, the integral controller eliminates the effect of the Coulomb friction.

4.1 The Lab steps to integral control.

(i) In the same directory, open labcartcolint and click `Quarc - Build`.

(ii) Put the cart at the left end of the track.

(iii) In the Simulink window, select `Tools - External Mode Control Panel - Signal & Triggering`, and change the duration to 15,000.

(iv) Double-click the error scope and select Parameters. Un-check the *Limit to 5000 data points* check-box.

(v) In the Simulink window, set the simulation time to 29 seconds.

(vi) The PID controller should be set to $k_p = 10$, $k_i = 15$ and $k_d = 0$. The step should be set to 0.4.

(vii) Click `Quarc - Build`. Select `Simulation - Connect to Target`, and `Quarc - Start`.

(viii) Once the cart stops, hit the stop button.

(ix) Save the error e in MATLAB as a Mat file. Take this data file with you.

4.2 In your lab report include the following under Part (iv):

(a) In Simulink compute the error e using Figure 5 with saturation and your value of f_c. Hand in the experimental plot of e and this Simulink plot of e on the same graph. Are these plots approximately the same? Does the steady state error go to zero?

(b) For $k = 10$ V/m, use root locus to find the largest value of k_i such that all the roots of the polynomial $ms^3 + cs^2 + \gamma ks + \gamma k_i$ are in $\{s : \Re s < 0\}$. Hand in this value of k_i and the root locus.

5 Part (v): Moving the cart with PID controller

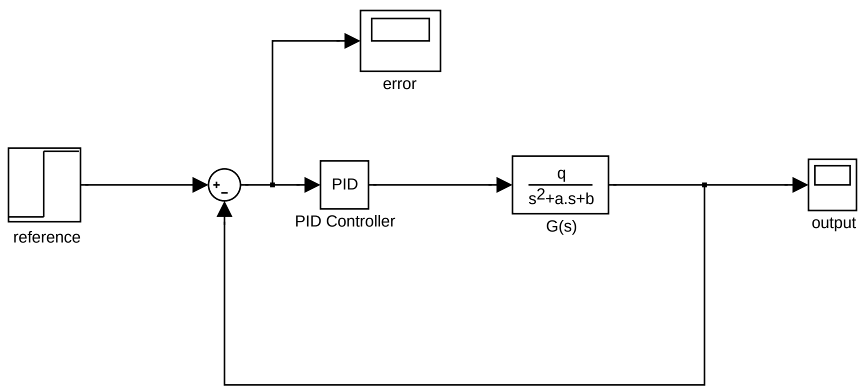

Figure 6: A PID control system.

In this section we will demonstrate how one can use a PID controller to move the cart to various positions on the track. Let us begin be recalling some properties of the proportional integral and derivative (PID) controller. Consider the closed loop system in Figure 6 where G is the open loop plant. The error is denoted by e and the output by y. Recall that a PID controller is a controller of the form

$$H(s) = k_p + k_d s + \frac{k_i}{s}. \tag{5.1}$$

The proportional gain is k_p while k_d is the derivative gain and k_i is the integral gain. The closed loop transfer functions from the reference signal r to the output y, and from the reference signal r to the error e are given by

$$\begin{aligned} \frac{Y(s)}{R(s)} &= \frac{H(s)G(s)}{1+H(s)G(s)} = \frac{(k_d s^2 + k_p s + k_i)G(s)}{s + (k_d s^2 + k_p s + k_i)G(s)} \\ \frac{E(s)}{R(s)} &= \frac{1}{1+H(s)G(s)} = \frac{s}{s + (k_d s^2 + k_p s + k_i)G(s)}\,. \end{aligned} \tag{5.2}$$

Now assume that the gains k_p, k_i and k_d are chosen such that the closed loop system is stable. In other words, assume that the gains k_p, k_i and k_d are chosen such that all the zeros of $1 + (k_p + k_d s + \frac{k_i}{s})G(s)$ are in the open left hand plane $\{s : \Re s < 0\}$. If the reference signal $r(t) = r_0$ is a constant for all t, then we have

$$y_{ss}(t) = \lim_{t\to\infty} y(t) = r_0 \quad \text{and} \quad e_{ss}(t) = \lim_{t\to\infty} e(t) = 0. \tag{5.3}$$

The steady state output $y_{ss}(t) = r_0$ and the steady state error $e_{ss}(t) = 0$. In our problem, this means that by setting the reference signal to $r(t) = r_0$, the cart will move to the position r_0 on the track.

In general one can choose the gains k_p, k_i and k_d to place the poles of the closed loop system at any three distinct locations in the open left hand plane $\{s : \Re s < 0\}$. However, placing three poles does not guarantee that the closed loop system will be stable. If the McMillan degree of G (the minimum order of system) is greater than or equal to three, then one does not have any control over the remaining poles of the closed loop system, and the closed loop system can still be unstable. In other words, all we can do is place three poles. If the closed loop system has more than three poles, then we have no control over these poles, and thus, placing three poles does not necessarily guarantee the closed loop system will be stable.

One way to design a PID controller is to vary the gains k_p, k_d and k_i to arrive at a desired closed loop response. For example, the proportional gain k_p acts likes a spring. By increasing k_p, in general, one decreases the rise time and increases the overshoot. The derivative gain k_d acts like damping or a shock absorber. By increasing k_d, in general, one decreases the overshoot and settling time. So one can vary k_p, k_d and k_i in a simulation of the step response of the closed loop system to achieve the desired response. This method is demonstrated in the web page www.engin.umich.edu/group/ctm/. The following table is taken from this web page and shows what happens when one increases the gains k_p, k_d and k_i in the PID controller.

Closed loop response	Rise time	overshoot	Settling time	steady state error
k_p	decrease	increase	small change	decrease
k_i	decrease	increase	increase	eliminate
k_d	small change	decrease	decrease	small change

Table 1: The effect of the gains in PID.

There are many procedures to design a PID controller for system of the form in Figure 6. However, our problem has saturation, and saturation can cause major problems in PID control design. The integral gain k_i is a double edge sword in PID design with saturation. The integral gain will eliminate the steady state error due to a step. However, in the presence of saturation a large integral gain can greatly increase the settling time and may even cause the system to become unstable. This is called *integral windup.* This integral windup problem is even more pronounced when the open loop system G is unstable. Since our open loop system has a pole on the imaginary axis, we have to be careful about integral windup. Therefore in our control design we have to face reality and design a PID controller for a system of the form in Figure 7 with the saturation included in our design. Finally, it is noted that we are ignoring the Coulomb friction in our model.

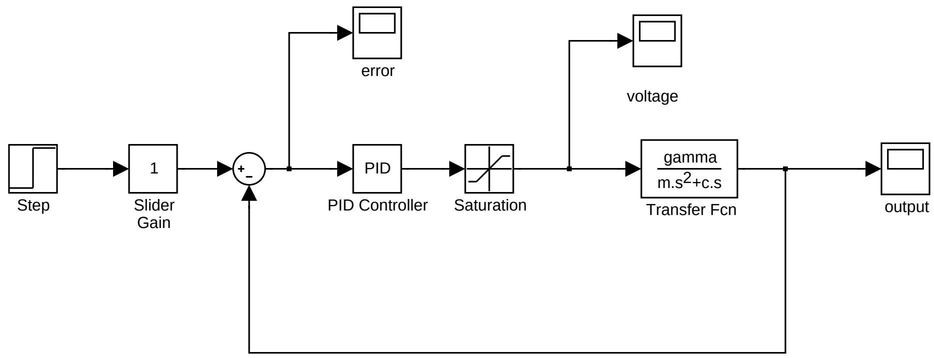

Figure 7: The feedback model with slider gain.

The Design Problem. We want to move the cart 0.5 meters in less than one second with an overshoot of less than 5%. Part of your pre-lab is to design a PID controller using Simulink. The open loop system is given by

$$G(s) = \frac{\gamma}{ms^2 + cs} \qquad (m = 1.0731\, kg \text{ and } B_{eq} = 5.4\, kg/s)\,. \tag{5.4}$$

Here we are including the mass of the cart and the cart weight. The numbers m, γ and B_{emf} you have already computed from the table in the Appendix. Recall that $c = B_{eq} + B_{emf}$. We have also provided an estimate of the viscous friction $B_{eq} = 5.4$ kg/s. It is emphasized, this value of B_{eq} may be different from the viscous friction B_{eq} that you will calculate experimentally. To design a PID controller, start with some reasonable gains, for instance $k_p = 40$ V/m, $k_d = 5$ Vs/m and $k_i = 5$ V/ms. Then start varying these gains by the rules provided in Table 1. Then for different choices of k_p, k_d and k_i simulate the output y using the Simulink model in Figure 7. Compute the rise time, percent overshoot, settling time, and steady state error. You may want to look at the voltage scope to see how much time you are in saturation. Saturation is not necessarily bad. This simply means that one is using the maximum force available to move the cart. Try to keep k_i small. Your design may even work with $k_i = 0$. Bring your best values of k_p, k_d and k_i that you found in your Simulink simulation to the Lab, along with the rise time, percent overshoot, settling time, and steady state error of your simulation.

For the experiment, set the step to 1 and the slider gain to 0.5 meters. Put the cart at the left end of the tack. Now run the experiment and plot the position of cart. Hand in your values for k_p, k_d and k_i from your Simulink simulation using Figure 7. Compare it with the plot of the experimental position of the cart. Did your PID controller actually achieve what you expected?

Now using your PID controller, set the slider gain in the range of $[-0.2, 0.2]$ meters. You should be able to move the cart any where from -0.2 to 0.2 meters on the track. Set the cart in the middle of the track with the slider gain at zero. Push the cart **gently**. You should feel some resistance. The PID controller is trying to hold the cart at position zero. Then move the slider gain to move the cart to any position on the track that you want. For example, move the cart to 0.2 meters, then back to zero. Then go to -0.2 meters and finally back to 0.2 meters. If you stop the cart at any position on the track, the PID controller will hold the cart at that position. To check this, move the cart to any position and gently push the cart.

5.1 The Lab steps to PID control.

(i) In the same directory, open up labcartpid and hit build. In MATLAB command window, type $X\_MAX = 0.6$.

(ii) Put the cart at the left end of the track.

(iii) In the Simulink window, select `Tools - External Mode Control Panel - Signal & Triggering`, and change the duration to 15,000.

(iv) Double-click the output y scope and select Parameters. Un-check the *Limit to 5000 data points* check-box.

(v) In the Simulink window, set the simulation time to 29 seconds.

(vi) Set the PID controller to the gains k_p, k_i and k_d that you computed in your Simulink simulation using Figure 7. The step should be set to 1 and the slider gain to 0.5.

(vii) Click `Quarc - Build`. Select `Simulation - Connect to Target`, and `Quarc - Start`.

(viii) Once the cart stops, hit the stop button.

(ix) You can change the PID gains k_p, k_i and k_d to improve the performance, and run the experiment again. Remember to select `Quarc - Build` after making changes. Note that you can do this only because it can be done cheaply.

(x) Save your best output y in MATLAB, along with the corresponding k_p,k_d and k_i. Take this Mat file with you.

(xi) To see how the integral controller helps you move the cart to any desired position, you can set the slider gain to be anything between $[-0.2, 0.2]$. Put the cart in the middle of the track. Then move the cart to several positions on the track.

5.2 In your lab report include the following under Part(v):

(a) Hand in your values k_p, k_i and k_d that you first computed in Simulink, with the corresponding rise time, percent overshoot, settling time, and steady state error.

(b) Hand in your values k_p, k_i and k_d that you used in your best run of the experiment, with the actual rise time, percent overshoot, settling time, and steady state error.

(c) Hand in your plot of the position of the cart from your best run of the experiment, and the plot of position of the cart from Simulink using k_p, k_i and k_d that you used to obtain the best run of the experiment. Compare them on the same graph.

6 Pre-Lab due at the beginning of the lab experiment. You will not be allowed run the lab experiment with out a complete pre-lab.

(i) Compute γ, m, and B_{emf}, using the parameters given by the table in the Appendix.

(ii) For $k = 10$ V/m and using $B_{eq} = 5.4$ kg/s, compute the largest value of k_i such that the polynomial $ms^3 + cs^2 + \gamma ks + \gamma k_i = 0$ is stable. Hand in the root locus.

(iii) Using $B_{eq} = 5.4$ kg/s with the Simulink model in Figure 7, compute the PID gains k_p, k_i and k_d which achieve less than 5% overshoot and a settling time of less than one second. These gains will be used in your PID experiment.

(iv) Hand in your best plot of the output y from Simulink that you obtained in step (iii), with the corresponding rise time, percent overshoot, settling time, and steady state error.

7 Appendix: The parameters for the cart and track

Symbol	Description	Value	Unit
R_m	motor armature resistance	2.6	Ω
L_m	motor armature inductance	0.18	mH
K_t	motor torque constant	0.00767	$N.m/A$
η_m	motor efficiency	100%	%
K_m	back-electromotive-force(EMF)	0.00767	$V.s/rad$
J_m	rotor moment of inertia	3.9×10^{-7}	$kg.m^2$
K_g	planetary gearbox ratio	3.71	
η_g	planetary gearbox efficiency	100%	%
M_c	cart mass	0.57	kg
M_w	cart weight mass	0.37	kg
L_t	track length	0.990	m
T_c	cart travel	0.814	m
P_r	rack pitch	1.664×10^{-3}	$m/tooth$
r_{mp}	motor pinion radius	6.35×10^{-3}	m
N_{mp}	motor pinion number of teeth	24	
r_{pp}	position pinion radius	0.01482975	m
N_{pp}	position pinion number of teeth	56	
K_{EP}	cart encoder resolution	2.275×10^{-5}	$m/count$