Rolling Mill Controller Project

Dr. Mohammad Naghnaeian's ME 4030 Control Systems Term Project

Dylan Holtzhauer

Introduction

This project centers around designing a controller for a rolling mill. The rolling mill is depicted below in Figure 1 [1].

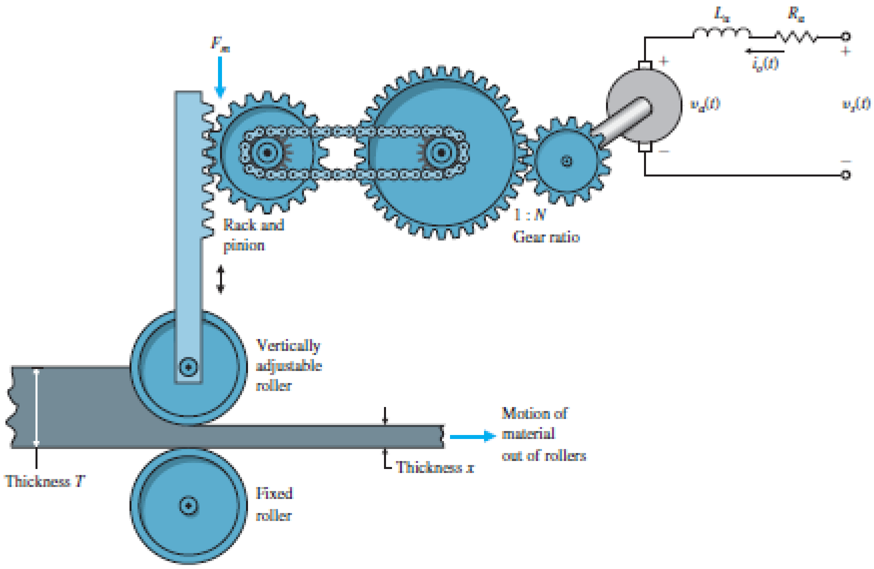


Figure 1. Diagram of Rolling Mill [1]

With this diagram, the equation of motion for the roller, equation for the circuit, and equation for the motor and gear train can be determined. Those equations can be combined to determine the transfer function of the plant. The work for this is shown below.

Determining Plant Transfer Function

ME 4030 Term Project Equation Work

Dylan Holtzhauer

Roller

Variables to consider

b - damping coefficient

m - mass of the roller

T - material thickness

C - thickness coefficient

- Eqn. of motion

$$m\ddot{x} = c(T - x) - mg - b\dot{x} - F_m$$

$$m\ddot{x} = cT - cx - mg - b\dot{x} - F_m$$

$$x = y + \delta, \quad \delta = \frac{cT - mg}{c}, \quad \dot{x} = \dot{y}, \quad \ddot{x} = \ddot{y}$$

$$\Rightarrow m\ddot{x} = cy - b\dot{y} - F_m$$

$$\boxed{m\ddot{y} = -cy - b\dot{y} - F_m}$$

- Laplace

$$ms^2X(s) = -cX(s) - bX(s) - F_m(S)$$

~~Fm(S)~~

$$(*) \quad \boxed{(ms^2 + bs + c)X(s) + F_m(S) = 0}$$

Dylan Holtzhauer

Circuit

Variables to consider

L_a - inductance

R_a - resistance

K_e - back EMF constant

- Eqn.

1) $$V_s(t) = R_a i_o + L_a \frac{di_o}{dt} + V_a(t)$$

- Laplace

$$V_s(S) = R_a I_o(S) + sL_a I_o(S) + V_a(S)$$

$$\Rightarrow I_o(S) = \frac{V_s(S) - V_a(S)}{R_a + sL_a}$$

- Eqn

2) $$V_a(t) = K_e \omega = K_e \dot{\theta}, \quad \theta = \frac{Nx}{R} \Rightarrow \dot{\theta} = \frac{N\dot{x}}{R}$$

$$V_a(t) = K_e \frac{N\dot{x}}{R}$$

- Laplace

$$V_a(S) = \frac{K_e N}{R} sX(S)$$

2→1:

$$I_o(S) = \frac{V_s(S) - \frac{K_e N}{R} sX(S)}{R_a + sL_a}$$

Dylan Holtzhauer

Motor

Variables to consider

K_t - torque constant

N - gear ratio

R - rack and pinion effective radius

- Eqns.

$T_{motor} = K_t i_o$

$T_{rack} = N K_t i_o$

$T_{rack} = R F_m$

$\Rightarrow F_m = \frac{N K_t i_o}{R}$

- Laplace

$F_m(s) = \frac{N K_t}{R} I_o(s)$

Insert circuit eqn.

$$(**)\quad \boxed{F_m(s) = \frac{N K_t}{R}\left(\frac{V_s(s) - \frac{K_e N}{R} s X(s)}{R_a + s L_a}\right)}$$

Dylan Holtzhauer

Combining Roller Eqn. of Motion and Motor Eqn.

(*) and (**)

$$(ms^2+bs+c)X(s)+\frac{NK_t}{R}\left(\frac{V_s(s)-\frac{K_eN}{R}sX(s)}{R_a+sL_a}\right)=0$$

$X(s)=Y(s)$

$$\frac{V_s(s)}{Y(s)}=\frac{NK_tR}{R^2(R_a+sL_a)(ms^2+bs+c)+sN^2K_tK_e}$$

Plant

Determining Parameters

Through determining the equations and ultimately the transfer function for the plant, several parameters need to be determined to see the characteristics of the plant. None of these are exact, though an attempt was made to justify each.

Parameter	Constant Chosen	Justification
b – damping Constant	.05	Assumed to be fairly low, regardless shouldn't matter much as the roller isn't spinning all that fast.
m – mass of the roller	10 kg	Given that rollers are probably around 1.5ft long and probably 3" in radius, with a decent section hollowed out, 10kg seems reasonable.
T – material thickness	.0065 m	Assuming the upper bound of commonly found sheet metal is ~5mm, the input thickness is probably around 1.5mm thicker. The paper, "Modelling and control of a hot rolling mill" **[2]** shows a similar difference in input and output material thickness being around 4.65mm, with an input of 6.12mm and output of 4
c – thickness coefficient	375,000	This parameter is in the equation F=c(T-x). Knowing T-x is ~2mm, and from the paper, "Modelling and control of a hot rolling mill" **[2]** it can be seen that the force due to rolling is around 750 metric tons. This can be used to solve for c.
L_a – Inductance	0.01 H	Looked up inductance in other motors, found an example with 10mH
R_a – Resistance	20 Ohms	Found examples online of large 3-phase motors online with 20 Ohm resistance.
K_e – back EMF constant	.8	In most examples online, I this being around .8 or so.
K_t – torque constant	50	In most examples online, I saw a ratio of about 1:50 for the back EMF constant and torque constant, so I'm just choosing 50.
N – gear reduction ratio	50	Looking at the force due to rolling, the gear ratio is going to need to be pretty high, starting out with 50. Motor doesn't need to take a full load as the rack can be preloaded, but it needs a decent bit of range.
R – rack and pinion effective radius	.075m	Hard to determine without have a real physical setup, just going to go with 75mm.

Performance Criteria

There are several performance criteria parameters that need to be determined. Below is a table with the selected values and justifications

Rise Time

The rise time isn't important to the system in terms of how fast it can compress the incoming sheet from its initial offset, so a strictly low value for rise time isn't needed for this purpose. However, the oncoming sheet isn't perfectly consistent. It will have variations itself, so it's ability to quickly respond to those changes are important. For this reason, I'm going to decide the rise time must be at least 0.5 seconds.

Settling Time

The settling time is less important in this situation than the rise time. The settling time will determine how long it takes to reach within 1% of the final value. This will have less to do with the system's ability to react to changes in input, and more how long it takes to respond to a large step. For these reasons I'm choosing an acceptable settling time of 5 seconds.

Peak Overshoot

The overshoot has a similar situation to rise time, in that the initial overshoot isn't of much importance, but in the event the controller doesn't result in a smoothing of the input, it's important that it doesn't start overshooting the incoming inconsistency. Therefore, I'm choosing a Mp of 0. If that is not possible, something small such as 5% should be acceptable. In the even that it can deal well with incoming fluctuations, there is no specification for Mp, as the initial overshoot isn't important as the first 10 or so seconds of rolling can be discarded.

Error at Steady State

Error at steady state is probably the most important performance criteria. If at all possible, the error at steady state in response to a step should be zero. If that is not attainable, .01 should be adequate. Looking at offerings on the market, there is a tolerance of ±0.15mm, which is an error of 3%. This also has to consider things like the inconcsistency in the roller's thickness radius, temperature variations during rolling, so if error at steady state can't be 0, 1.5% should be good enough.

Analysis of Performance Criteria

Quick Note:

Due to the large value for the thickness coefficient, the coefficients and constants for the controller are likely going to need to be fairly large.

Steady State Error

$$e_{ss} = \frac{1}{1+C(0)P(0)}$$

For error at steady state to be zero as a response to a step input, the above can be evaluated. For this to be truly 0, the denominator must be infinity. We don't have control of the plant, which when evaluated equals 185/42190. As of right now, to get an steady state error of 1.5%, C(0) has to be 14815 or greater. An alternative choice is to use an integrator to make the steady state error truly zero, as an integrator evaluated at zero is infinity.

Peak Overshoot

When ζ is equal to or greater than 1, the overshoot will be zero. Otherwise, the equation below can be used to determine a damping ratio for a specific percentage overshoot. Using this equation, with a max percentage overshoot of 5%, ζ must be greater than or equal to 0.472.

$$\%Mp = e^{\frac{\pi\zeta}{\sqrt{1-\zeta^2}}}$$

Rise Time

The rise time can be determined by the equation below. This means ω_n needs to be at least 3.6 to meet the rise time of 0.5 seconds or less.

$$t_r = \frac{1.8}{w_n}$$

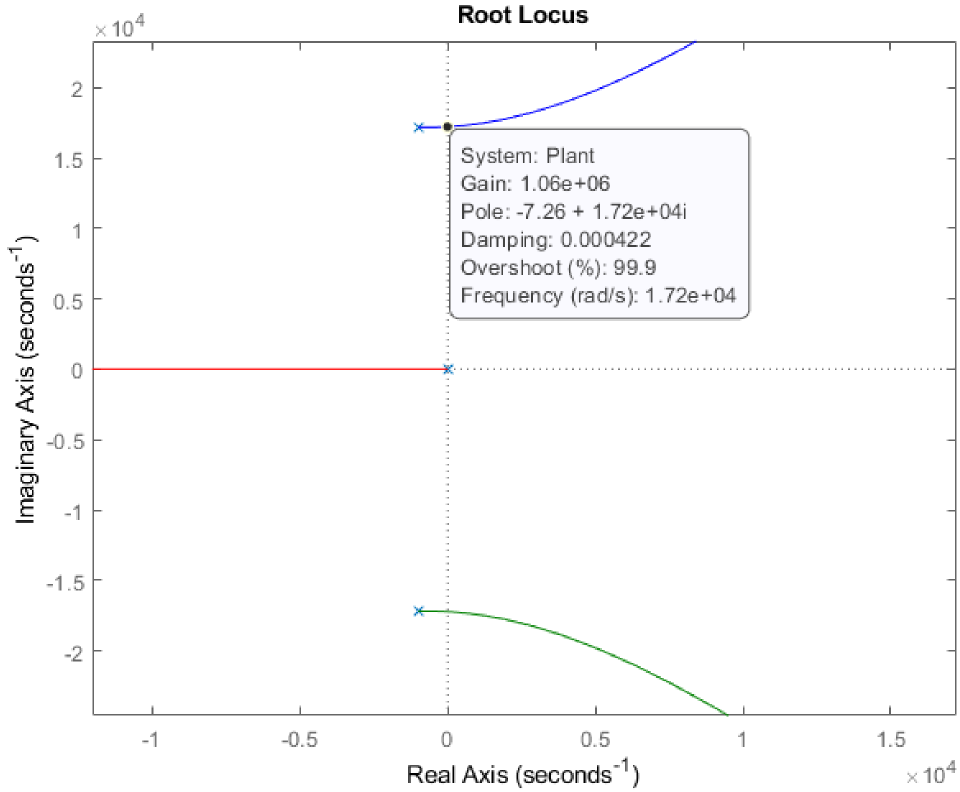
Settling Time

With the settling time known to be at least 10 seconds and with the equation below, that means $\zeta\omega_n$ equals at least 0.46.

$$t_s = \frac{4.6}{\zeta w_n}$$

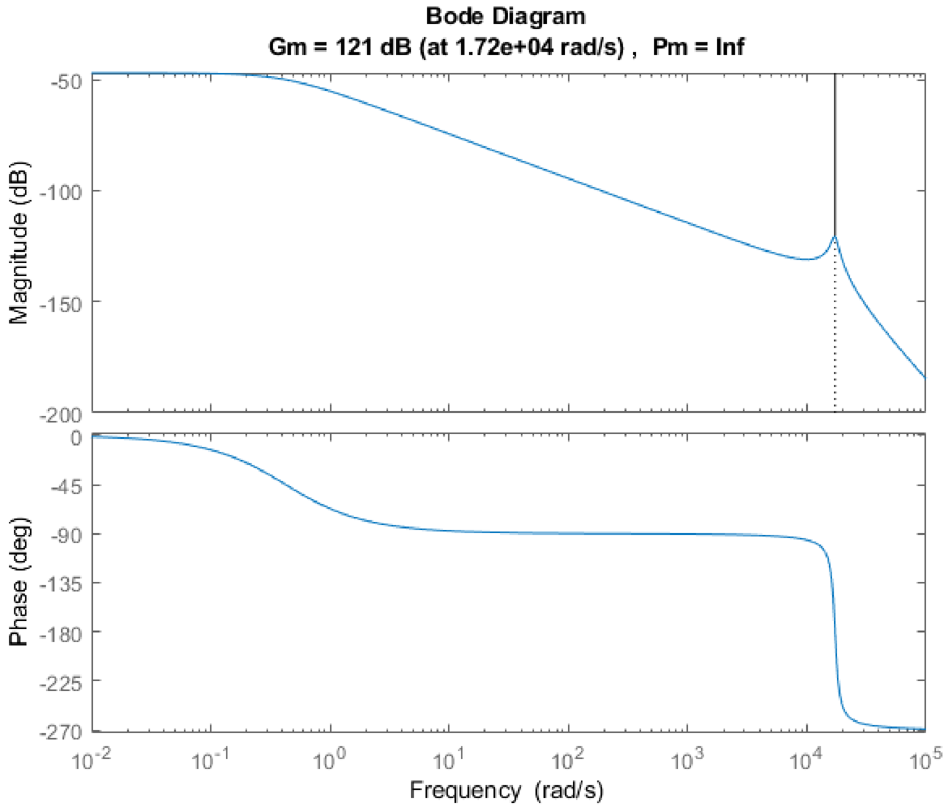
Plant Characteristics

Root Locus of Plant



It can be seen that the Plant has a poles on the real axis going from 0 to negative infinity. This shouldn't cause a problem as a proper gain can be selected along it to meet performance specifications. There are two complex conjugate poles starting at -1000 along the real axis, ±1720 along the imaginary axis. This will be the main point of concern for reaching the required performance criteria. There is also a max gain 1.06*10^6 before the system is unstable, with only using a proportional controller.

Bode Plot of Plant



The plant's Bode plot shows that it is a type zero system with negligible slope at lower frequencies. There technically a gain margin of 121 dB, though it doesn't make much difference with the phase margin not existing. This means the plant is naturally unstable without a controller. At the very least there needs to proportional controller to move the magnitude plot up to get a gain crossover frequency, eventually producing a positive gain margin and phase margin. There is a phase crossover frequency of approximately 20,000 rad/s.

Note on Designing a Controller

It was noticed that using an extremely high value using only a proportional controller, the requirements could be met. However, this doesn't necessarily mean it will work in real life. This leads to an extremely low rise time, and with the system likely using a controller in real life operating at a frequency lower than the rise time, and with the mechanical components deforming under load, the controller would not perform well in real life. Therefore, when considering what controller to use when designing with a root locus or bode plot, I'm going to decide to use different/additional controllers.

Designing a Controller with Root Locus

ζ is the angle of the from the positive imaginary axis. The angle can be determined from the following equation.

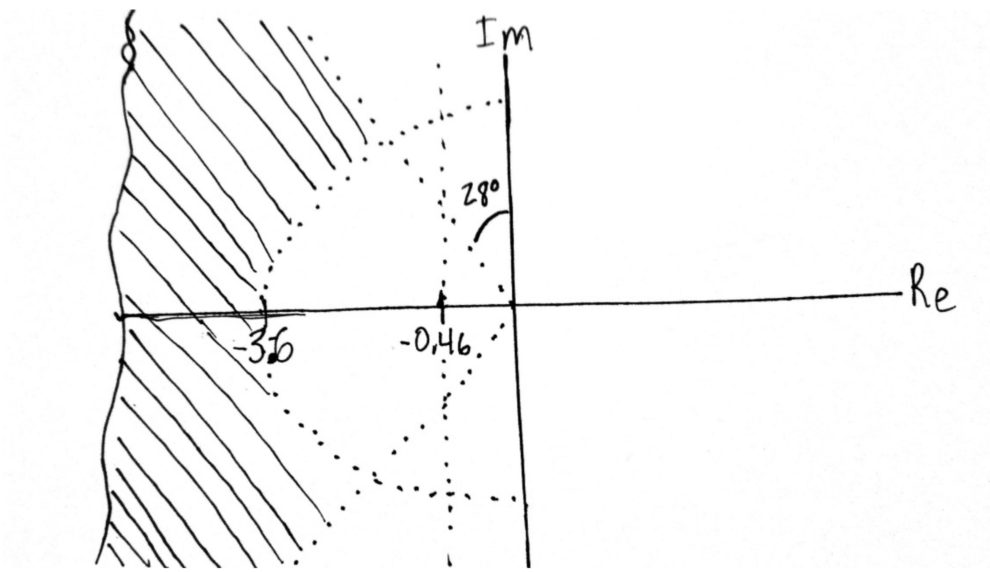
$$\sin^{-1}(\zeta) = \theta$$

Using the minimum ζ of 0.427 calculated earlier, this implies θ needs to be at least 28°.

The ω_n term is the magnitude of the vector from the origin to the, which forms a circle around the origin. As calculated earlier, it needs to be at least 3.6 away from the origin.

With $\zeta\omega_n$ known to be at least 0.46, that means the real portion of the dominant pole needs to be at least 0.46.

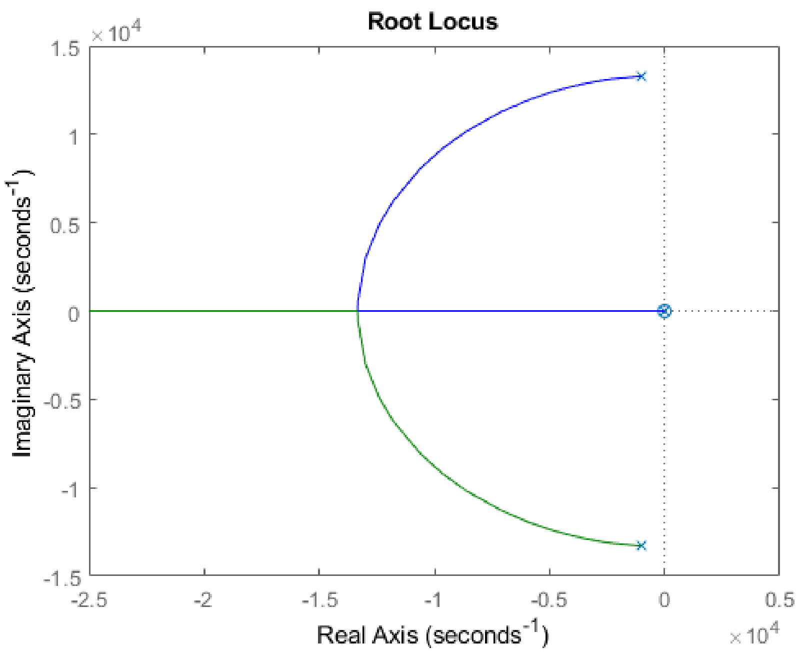
These three requirements listed above, the following diagram can be used to determine acceptable places for the dominant pole. The Shaded region is the acceptable region.

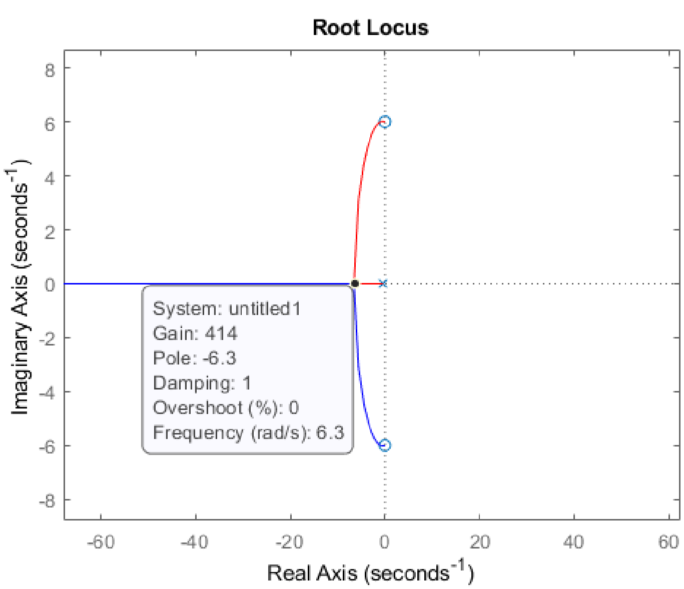


To remove the steady state error, removing the complex conjugate poles helps, or at least modifying them such that they intersect with the real axis. To do this a zero can be added to modify the complex pole's path. The zero should be somewhere on the right of there the complex pole line starts on the root locus. Knowing the locus' line starts at -1000, then that means The pole can be place around the square root of 1000, meaning, the controller shown below.

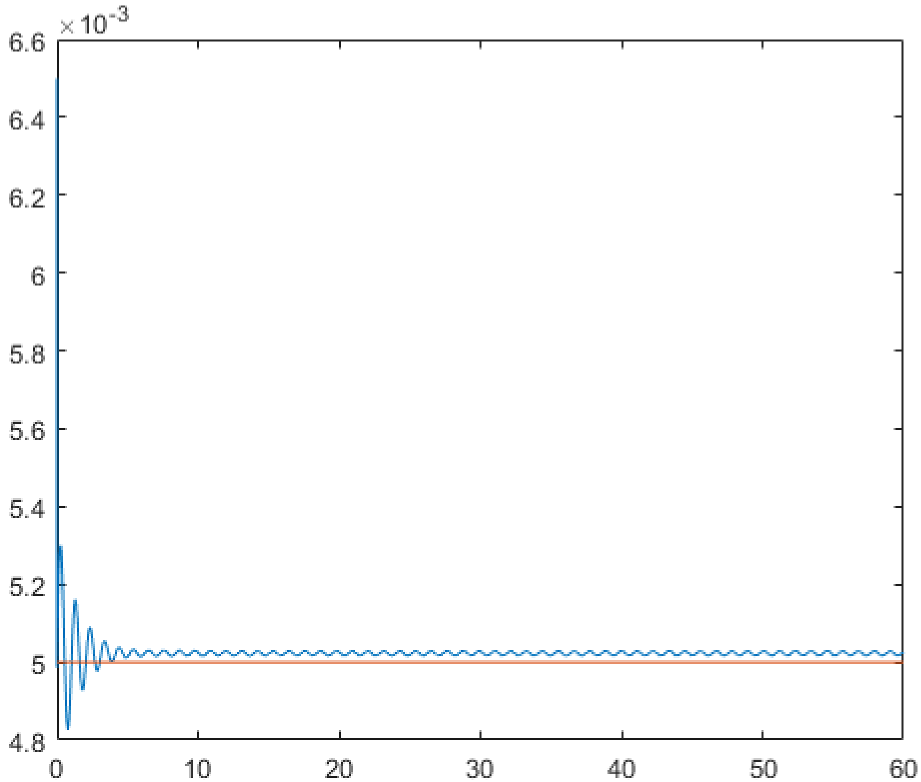
$$C(s) = k\left(s^2 + 31.62\right)$$

Plotting the root locus of the closed loop transfer function with this as the controller provides the plots below. The second is zoomed in to show the complex zeros.



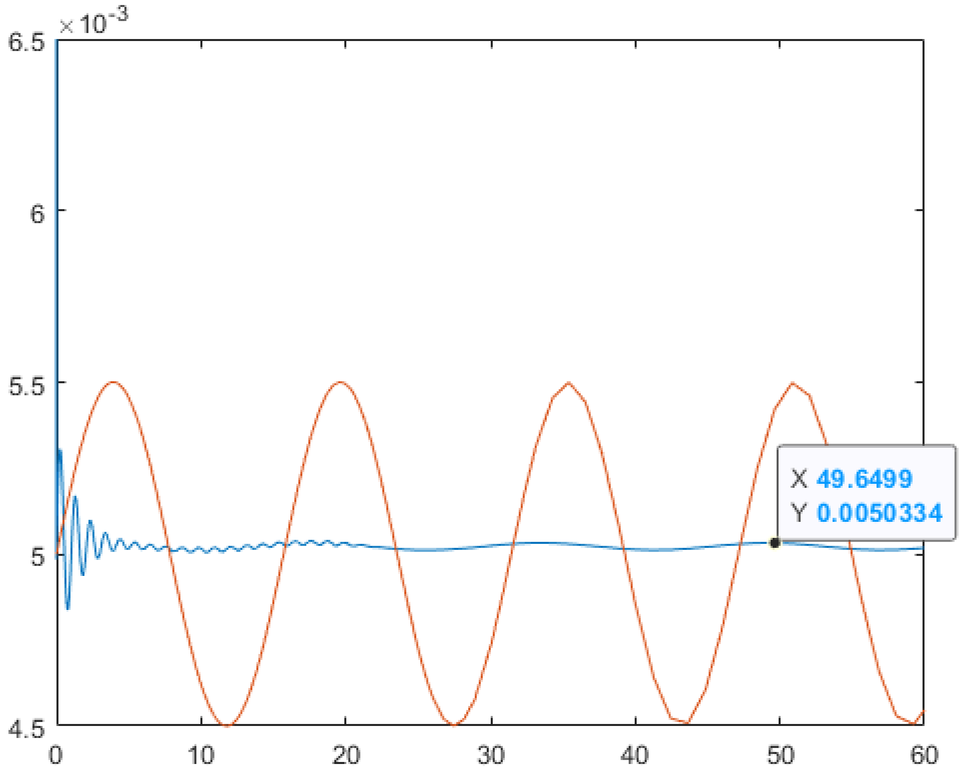


This shows that a gain around 400 should move it to the real axis, it can also be seen that it is in the acceptable pole region. Using this results in the response below when starting with a 6.5mm strip down to 5mm, along with the information provided by stepinfo.




```
        RiseTime: 2.1770e-04
    SettlingTime: 3.4759
     SettlingMin: 0.7998
     SettlingMax: 1.1152
       Overshoot: 13.2581
      Undershoot: 0
            Peak: 1.1152
        PeakTime: 0.7760
```

This is acceptable with regards to all the specification, except two. The error at steady state is not zero, but it is 0.4%,which is less than the alternative option of 1.5%, which is fine. The other is overshoot value, which is 13.25, is over the value of 5%, however, I made the stipulation that the overshoot doesn't matter if it can deal with incoming fluctuations properly. Below is the CLTF response to an incoming fluctuation of 0.5mm at ~1/15Hz, which seems like a reasonable fluctuation for the incoming sheet. The orange line is the incoming thickness minus delta between the intial and desired thickness. It can be seen the error at steady state is still acceptable, therefore it is fine that the Mp is 13.25%.



Therefore, the final controller chosen from the root locus design is:

$$C\left(s\right)=400\left(s^2+31.62\right)$$

Which has the following specifications.

```
    RiseTime: 2.1770e-04
SettlingTime: 3.4759
 SettlingMin: 0.7998
 SettlingMax: 1.1152
   Overshoot: 13.2581
  Undershoot: 0
        Peak: 1.1152
    PeakTime: 0.7760
```

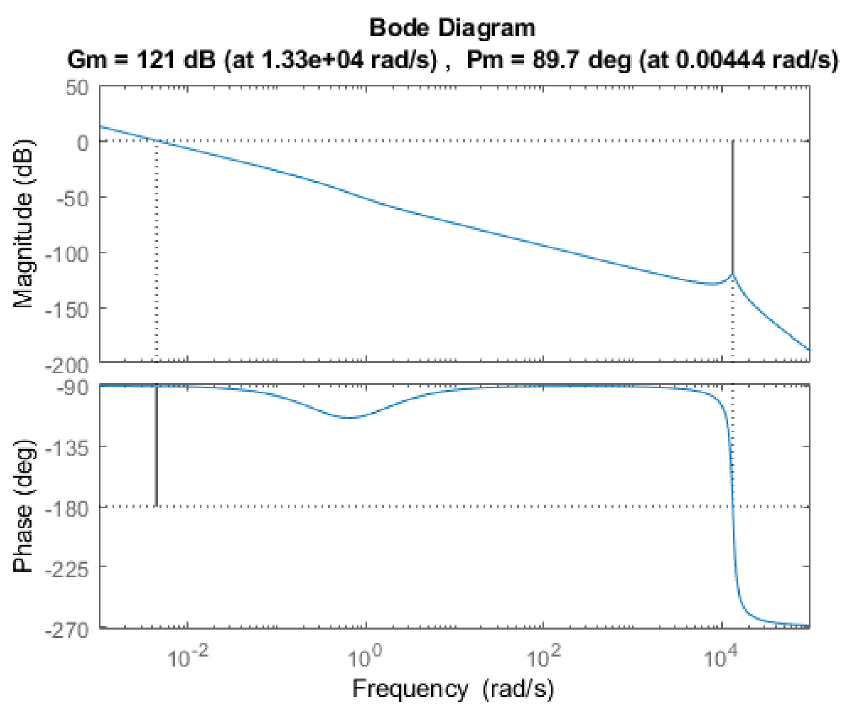
Designing a Controller using Bode Plots

It can be seen from the bode plot of the plant earlier that it is a type 0 system. To get the steady state error of the system to 0, we need to increase the type of the system. To do this an integrator can be used. The gain of the system also needs to be raised significantly to get a gain crossover frequency. Therefore, a PI controller was chosen, which has the following form.

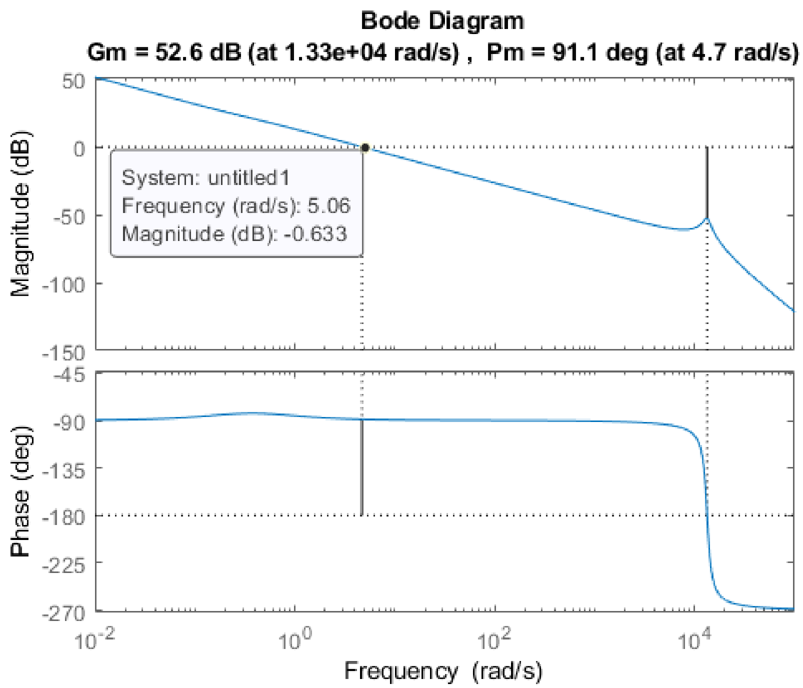
$$C(s) = \frac{K_p s + k_i}{s}$$

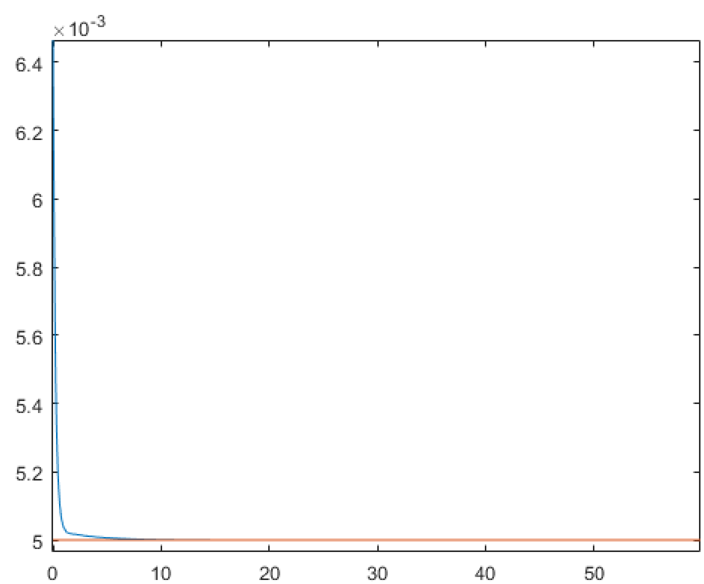
When determining the phase margin, we know we want no overshoot, therefore ζ is at least 1, which leads to a phase margin of 100°. The steady state error of the system should be zero, so and initial slope on the magnitude of the bode plot is needed. Knowing a rise time of at least 0.5 seconds and using equation for rise time with a phase margin at least 90°, $t_r=2.2/\omega_g$, which leads to an ω_g of at least 4.4.

Those calculations require a phase margin of at least 100°, and an ω_g of at least 4.4 rad/s. The bode plot of the closed loop transfer function with 1 for k_p and k_i yields the following.

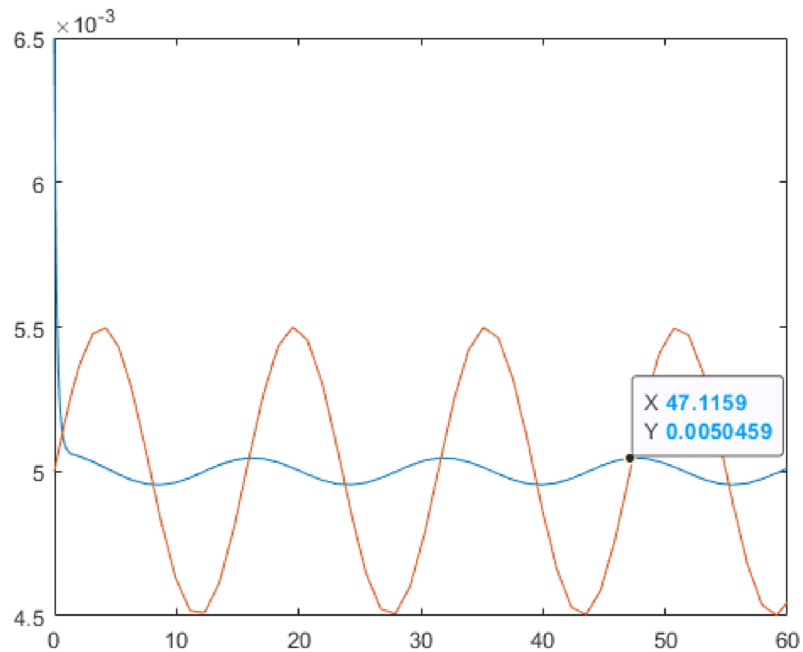


With a simple PI controller, the system is now stable, though the gain crossover frequency is very low, around 10^-2. That is lower than the required crossover frequency of at least 4.4 rad/s. Now a gain can be added in front of the controller to increase ω_g. It was found that ~60dB needed to be added to the magnitude for it to have a crossover frequency of 4.4 rad/s. Testing also showed that there was overshoot, so K_p was changed to by iteratively checking. The controller was multiplied by 837 to get the crossover frequency to 5 rad/s, resulting in the following bode plot and step response.





The step response is very clean in my opinion. It has more than the required phase margin, which means it has a good damping ratio which leads to no over shoot, there's no oscillation, and the steady state error is more or less non-existant, showing no difference between the desired and resulting values up to the 4 decimal point past a millimeter that MATLAB shows. Below is a test of it's response to an incoming oscillation, the same 1/15Hz and .5mm magnitude as was tested for the root locus controller.



It can be seen that there is only a 2% error with the incoming fluction of 10%, which is very successful in my opinion. Below is the final controller derive from frequency response, and the transient characteristics of the closed loop transfer function.

```
    RiseTime: 0.4937
SettlingTime: 1.0930
 SettlingMin: 0.9007
 SettlingMax: 0.9976
   Overshoot: 0
  Undershoot: 0
        Peak: 0.9976
    PeakTime: 6.7028
```

$$C\left(s\right) = 837\left(3 + \frac{1}{s}\right)$$

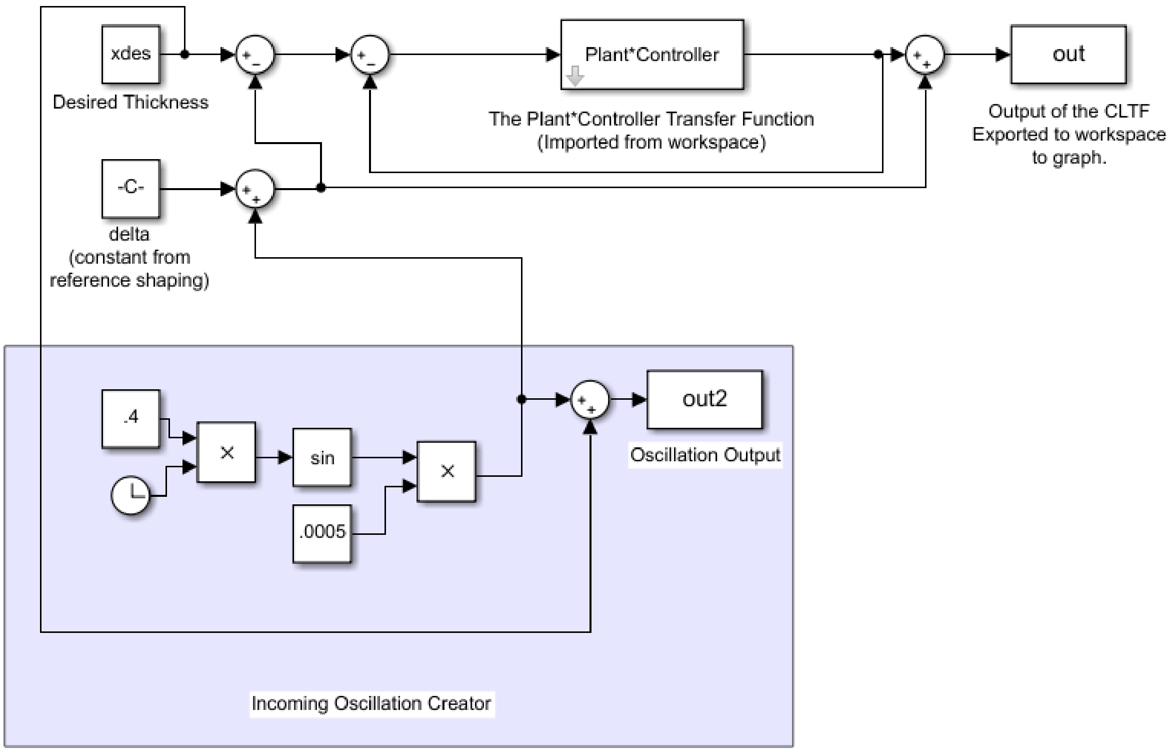
Conclusion

It was shown in the design of the controllers that when the model is ran, they both meet all specifications. For checking the robustness of the CLTF, I randomly changed most of the variable by about 10%. I even multiplied the weight of the roller x6, and the inductance x10. The frequency response controller was virtually unchanged, while the root locus derived controller essentially wasn't able to dampen out a small residual frequency like it was able to with the default parameters. In the end, both controllers seemed robust, but the Bode version more so.

Ultimately, I enjoyed designing with the Bode plot more. The root locus was nice to be able to get insights into the system, but it is hard to work with. With the Bode plot it's easier to see what kind of controller you need, and it's easier to combine controllers by knowing that they just add up.

As far as the CLTF responses, I prefer what the Bode version made. It more cleanly reaches the desired thickness and has no oscillation as it is overdamped. The transient characteristics are preferable in my opinion. Earlier I mentioned the rise time being very low isn't likely going to work out like that in reality, and the root locus controller has an extremely low rise time, while the Bode controller has the rise time of 0.49 seconds, which is quick and realistic. The Bode controller also responded better to the incoming frequency input. I found it interesting that in a way, that helps determine how fast the mill can be ran, as that frequency is going to increase, the controller won't be able to respond fast enough.

Appendix – Matlab Work



Labeled Simulink Diagram

```
Plant=(N*Kt*R)/(R^2*(Ra+s*La)*(m*s^2+b*s+c)+s*N^2*Kt*Ke)
```

Plant Transfer Function

References:

[1] Franklin, Gene F., J D. Powell, and Abbas Naeini. *Feedback control of dynamic systems*.

NY, NY: Pearson, 2019. Print.

[2] Rossomando, Francisco & FILHO, J.. (2006). Modelling and control of a hot rolling mill.

Latin American applied research. 36.