MSE 381: Feedback Control Systems

Project Report: Applications of Negative Feedback

Monday, August 5th, 2019

Sepehr Rezvani, 301291960

Project Group 22

SCHOOL OF MECHATRONIC SYSTEMS ENGINEERING

Contents

Abstract

This report details the process of designing a negative feedback control system in electronics. A PID controller and a lead-lag compensator were added to the system in order to improve the step response. The system and its components were designed and simulated using Matlab and LTSpice. First, a simple open loop system was investigated, composed of a LRC circuit. Given values of the components, the open loop transfer function was determined. From this, the step response, bode and Nyquist diagrams, poles, and root locus were obtained from Matlab and LTSpice simulations. Next, negative feedback was implemented into the circuit using a differential amplifier which produced the error signal of the system. A PID controller was then added to the system to improve the response. The controller consisted of various op amp circuits each corresponding to proportional, integral, or differential gain. A sufficient response was found in Matlab by adjusting the gains, and because the components of the controller circuits were correlated with the gain coefficients, component values were found and the system was simulated in LTSpice. After PID control, the system was tested using a lead lag compensator. First, a lag compensator was implemented which demonstrated that it could increase response and settling speed but would cause the system to become unstable for certain values of the components. Next, a lead compensator was added to the system which improved stability and settling time but increased the steady state error. Combining the two, a lead-lag compensator was designed in Matlab and tested in LTSpice, resulting in an improved response.

Introduction

The primary objective of this project is to gain hands on experience designing controllers using negative feedback. Various control design tools such as root locus, bode and Nyquist plots will be employed. To do this, Matlab and LTSpice will be used throughout the report to analyze the plant's behavior.

Certain files Matlab and LTSpice files have been provided for this project, and they are used to help model the transfer functions and systems without having to do extensive research on how to use the two programs.

To begin, we investigate the open loop behavior of a simple LRC circuit and its parameters. This eventually will lead into the circuit's closed loop characteristics which will be necessary when determining the effect of negative feedback control. Following this, a PID control will be designed for the LRC system, the end goal being a lead-lag compensator.

Theory

The purpose of this project is to experiment and examine various Feedback control systems theories and concepts in an attempt to witness the practical aspects of Feedback control systems. These experiments involve the creation and experimentation with Mat lab Simulink models representing the systems in question. These models are then tested using different signals including, but not limited to step responses. Theoretical models and responses are calculated and compared to their simulated counterparts to examine the accuracy of feedback control system concepts.

PID controller

A PID controller is a very common solution to practical control problems because of its simplicity. The controller acts on the error of a system, which is the desired system state minus the current system state$e(t)$. The output of the controller $u(t)$ is as follows:

$$u(t) = kp * e(t) + ki \int_0^t e(\tau)d\tau = KD * \frac{de}{dt}$$

Three distinct parts of the controller can be identified as the proportional term, integral term, and derivative term. Increasing the proportional term kp decreases rise time, and steady state error, while increasing overshoot. Proportional control alone is not sufficient for certain applications that require fast response time and zero steady state error. A proportional control only makes the system rise faster but has many others down sides.

One way to reduce the steady state error is the integral term ki acting on the integral of the error. However, this also increases the overshoot and oscillatory response of the system. The integral term takes the sum of all prior error and increases the system output by that amount multiplied by integral constant. An increase in system due to integral also leads to overshoot most of the times as these constants are very hard to optimize using trial and error method.

The derivative term kd acting on the derivative of the error anticipates the future state of the system and adjusts the controllers output accordingly to reduce overshoot and oscillatory response. The derivative term of the PID controller constantly compares current output to the reference signal and reduces the speed of system as the output approaches reference signal.

Proportional

The proportional response gives an output response that is proportional to the current error. This response relies on Proportional gain, which is the ratio between the response of the output and the given input. The proportional gain also affects the speed of the response, with a larger proportional gain leading to a higher speed response. However, a higher proportional gain value could negatively affect the systems stability, therefore it should be limited to a certain range to keep the speed of the system relatively consistent without threatening the systems stability.

Integral

The Integral control sums e(t) and its previous values over time, decreasing the steady state error of the system until it reaches zero. This response removes any remaining error through addition of this control effect. The integral control increases corresponding to the error of the system, and as the error reaches zero, the integral control ceases to increase.

Derivative

The derivative controller anticipates the expected change of e(t) and responds in accordance. The output is proportional to the change of error over time, multiplied by a derivative control constant. Derivative response also has the ability to increase the speed of the response of the system, while also improving the stability. This is beneficial to a PID controller system because it allows for higher proportional and integral gains without putting the stability of the system in jeopardy.

Lag Compensators

Lag compensators can drastically reduce the steady state error in a system. A phase lag compensator lags the system, decreasing the phase for a range of input frequencies. The transfer function for the lag compensator of figure 7 can be seen below.

$$\frac{Vo(s)}{Vi(s)} = \frac{1}{\alpha}\left(\frac{s+\frac{1}{\tau}}{1+\frac{1}{\alpha\tau}}\right)$$

Where $\tau = R_2C$ and $\alpha = \frac{R_2}{R_1+R_2}$.

Lead Compensators

Lead compensators can increase the stability as well as the speed of the systems response. A phase lead compensator leads the system, increasing the phase for a range of input frequencies. The transfer function for the lag compensator of figure 8 can be seen below.

$$\frac{Vo(s)}{Vi(s)} = \beta * \frac{Ts+1}{\beta Ts+1}$$

Where $\tau = R_1C$ and $\beta = \frac{R_2}{R_1+R_2}$.

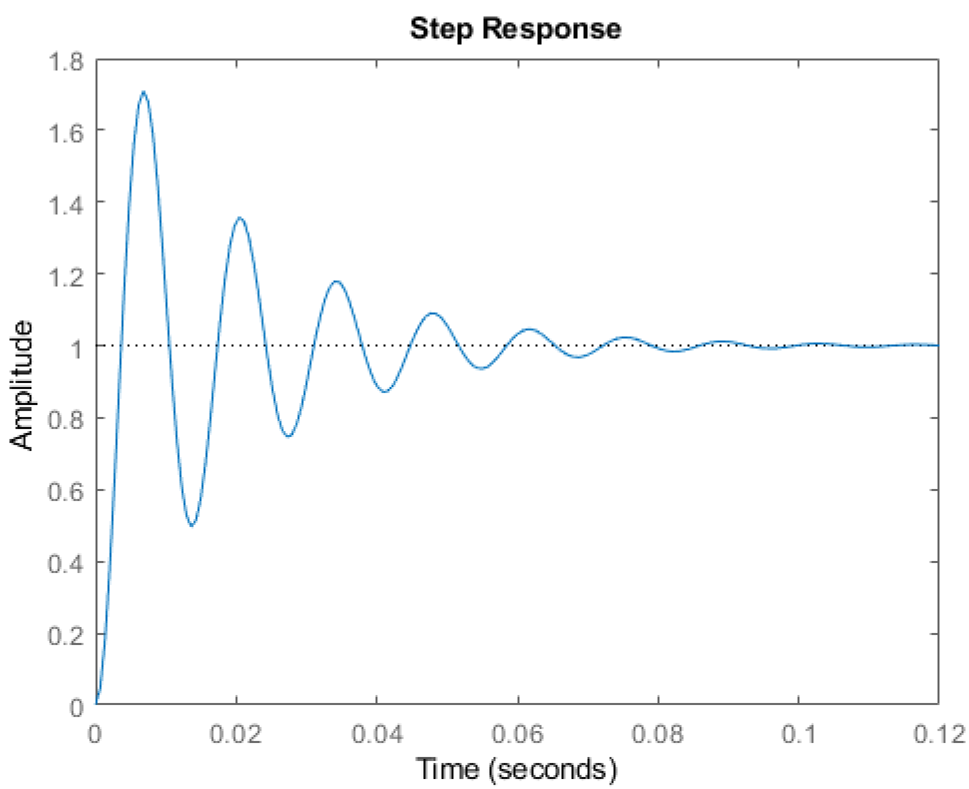
Results

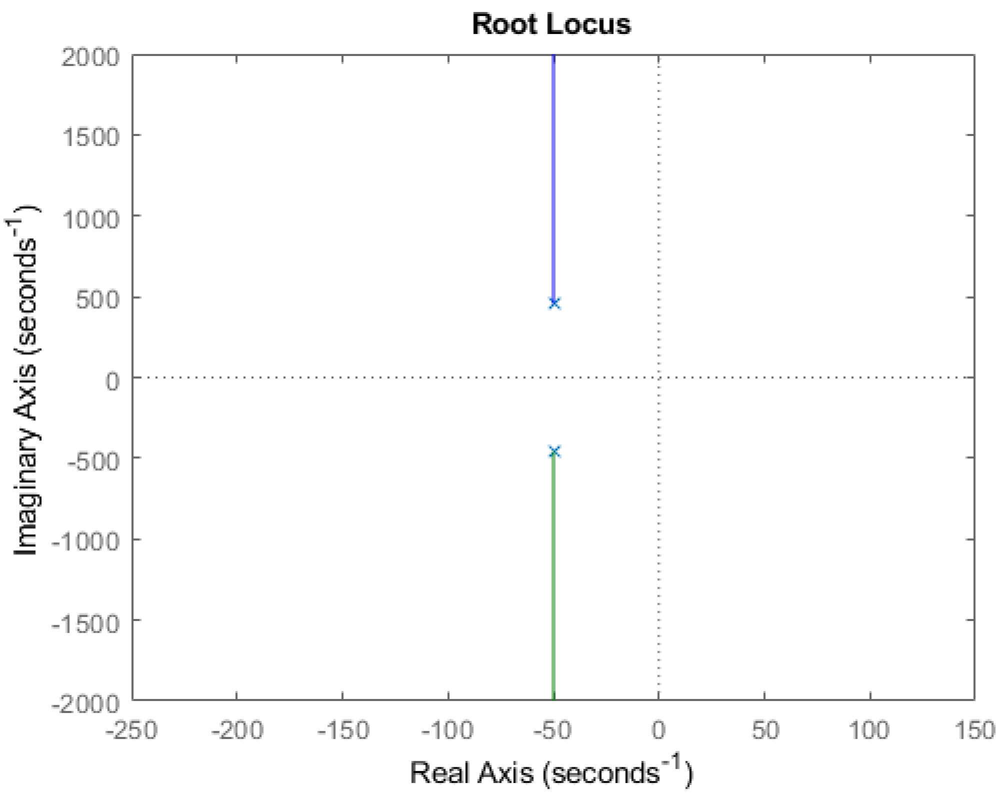
System Analysis

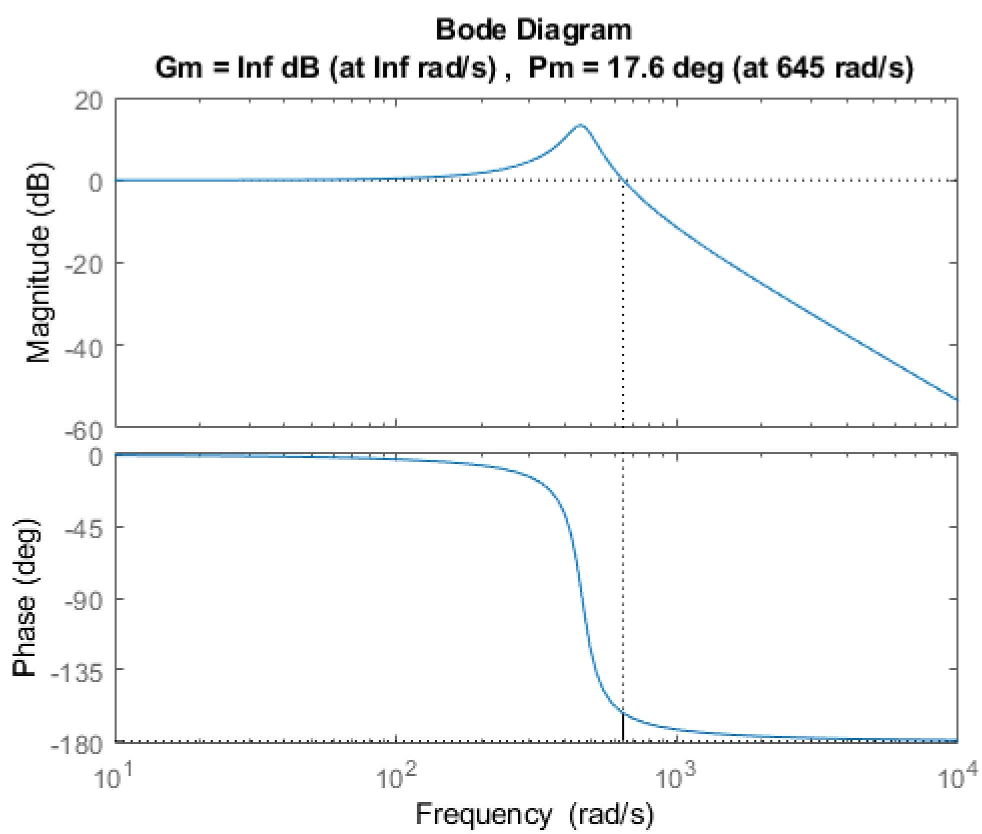
The open loop behavior of the LRC circuit can be found by first finding the transfer function of the circuit. Using the impedances for the circuit produces the transfer function:

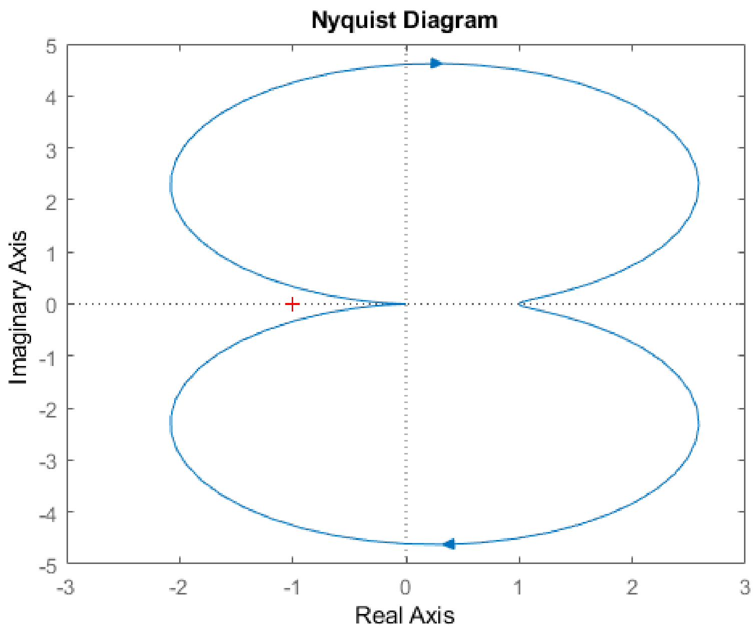
$$G(s) = \frac{Zc}{\frac{Zl}{Zc} + Zr * Zc + 1}$$

where Zl = Ls, Zc = 1/Cs, and Zr = R. Using this transfer function with the provided matlab files **openLoopTesting** and **LRC** outputs the following plots in Matlab:



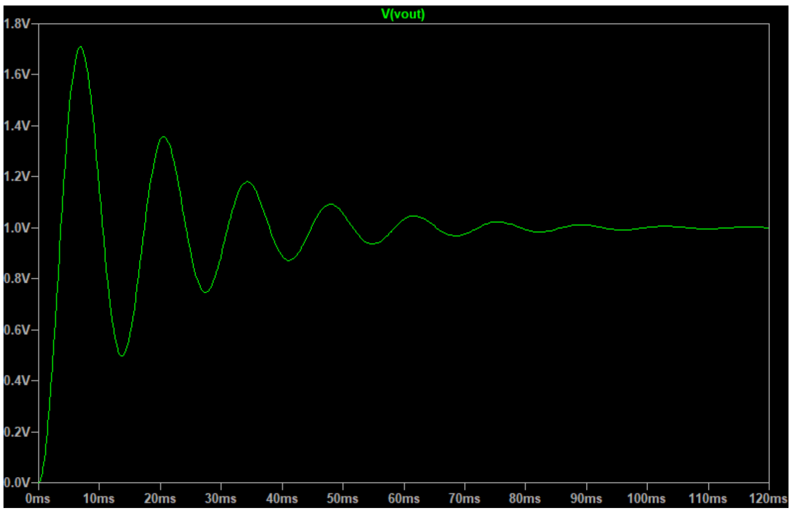

Root Locus
Imaginary Axis (seconds^{-1})
Real Axis (seconds^{-1})
2000
1500
1000
500
0
-500
-1000
-1500
-2000
-250
-200
-150
-100
-50
0
50
100
150

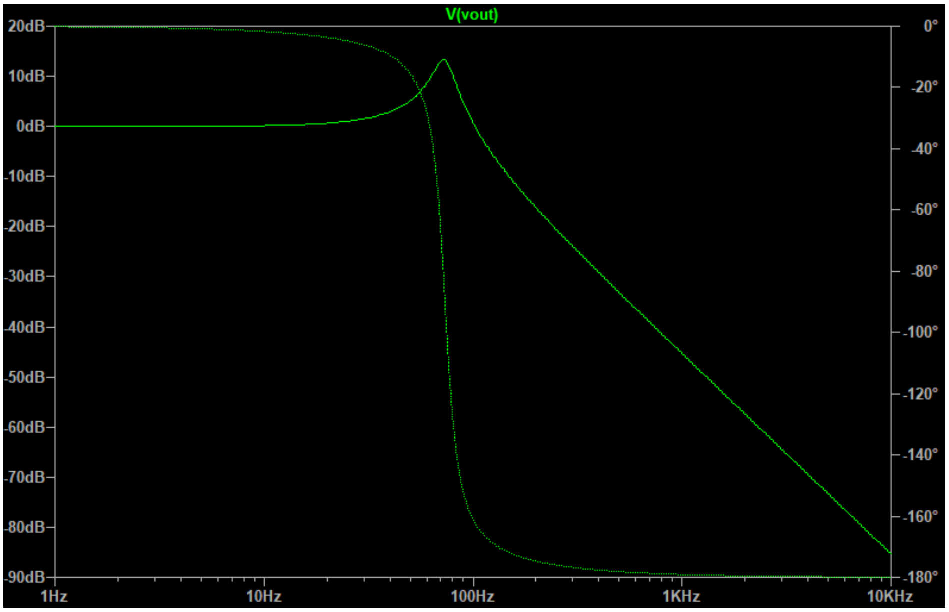

Bode Diagram
Gm = Inf dB (at Inf rad/s) , Pm = 17.6 deg (at 645 rad/s)
Magnitude (dB)
20
0
-20
-40
-60
Phase (deg)
0
-45
-90
-135
-180
10^1
10^2
10^3
10^4
Frequency (rad/s)



The open loop transfer function passes asymptotic and BIBO stability since it has no poles in the right-hand plane. As well as this, the gain margin is infinitely positive, and the phase margin is 17.6, a positive value.

The LTSpice files **stepresponse** and **frequencysweep** produce the following diagrams:





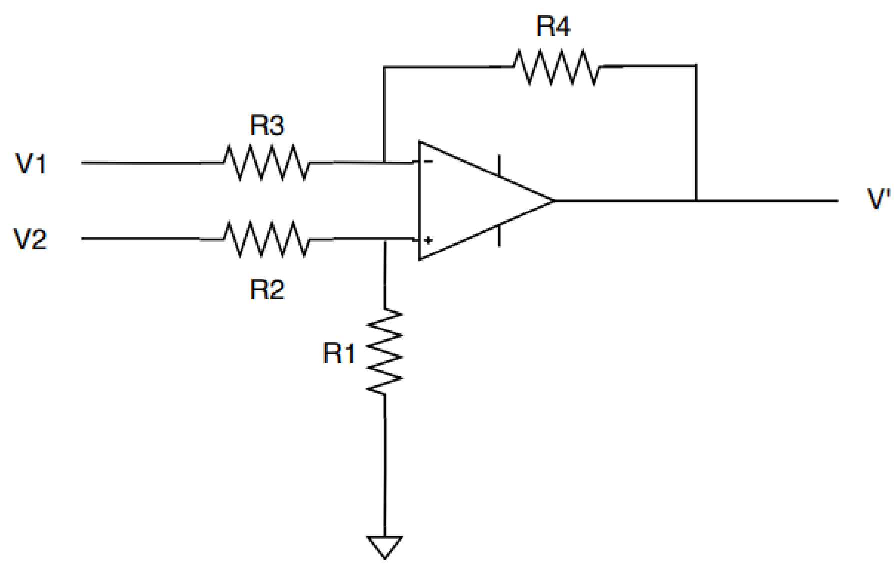
The plots produced in Matlab and LTSpice are essentially identical. The only difference is the x axis in Matlab is in rad/s and is Hz in LTSpice.

Individually reducing each components value by half in Matlab had the following effects:

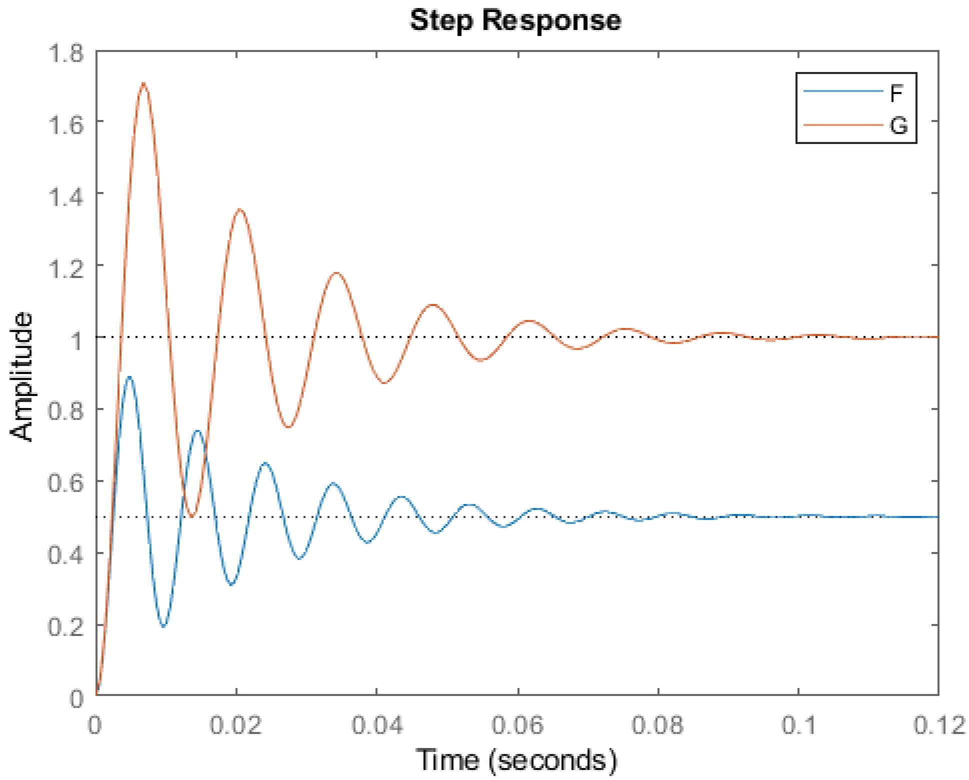
C	• Increased phase margin, • reduced percent overshoot • moved poles to the left • reduced settling time
R	• Increased percent overshoot, • Increased settling time, • increased frequency of oscillation, • lowered phase margin
L	• Increased percent overshoot • increased frequency of oscillation • reduced phase margin

Negative Feedback

The circuit shown in the figure below is an example of a differential amplifier. This circuit acts as the negative feedback component of the system. If R1=R2=R3=R4, then the output of the circuit will be V' = V2 - V1. Relating this circuit to a negative unity feedback system, we can consider V'=Verror, V2=Vref, and V1=Vout because Verror = Vref - Vout.

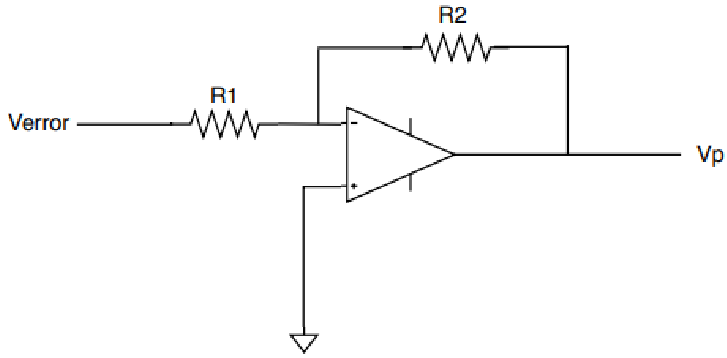


The closed loop response of the system without a controller was tested in Matlab, shown in the figure below. The closed loop transfer function is $F(s) = \frac{G(s)}{1+G(s)}$ and the poles are: $-0.5 \pm 6.5041i$ and $-0.5 \pm 4.5855i$. The system response time is slightly faster, but the steady state error is large.

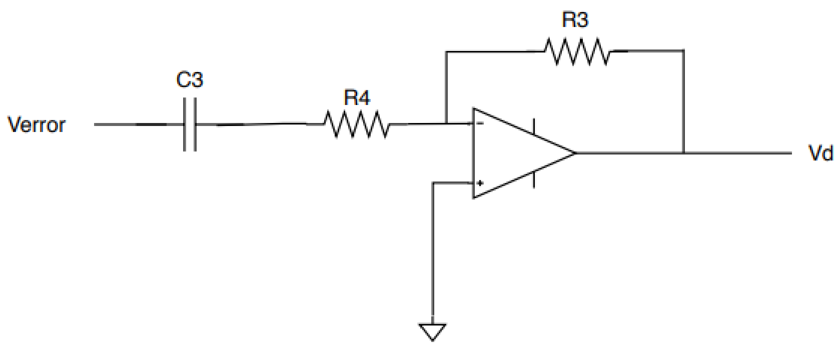

Step Response
Amplitude
Time (seconds)
F
G
0
0.2
0.4
0.6
0.8
1
1.2
1.4
1.6
1.8
0.02
0.04
0.06
0.08
0.1
0.12

PID Controller Design

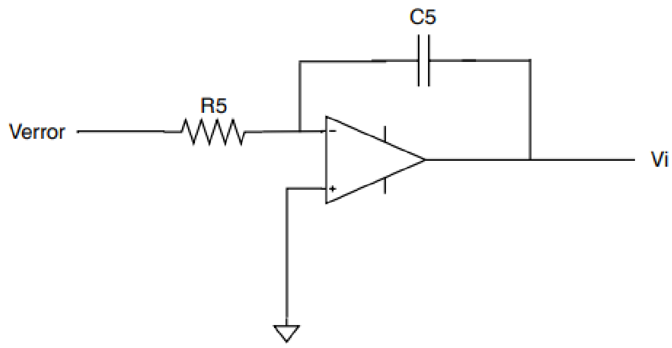
The PID controller consists of several different op amp circuits. For proportional gain, the circuit model is shown below. This is an example of an inverting amplifier circuit which has an output $Vp\ =\ -Verror\ \frac{R2}{R1}$. Therefore, the proportional gain coefficient Kp is equal to R2/R1.



For differential gain, the circuit model is shown below. This is an example of an op amp differentiator circuit. Assuming the resistance of R3 is much larger than R4, the output will be $Vd\ = -R3 * C3\frac{dVerror}{dt}$. Therefore, the differential gain coefficient Kd is equal to R3*C3.

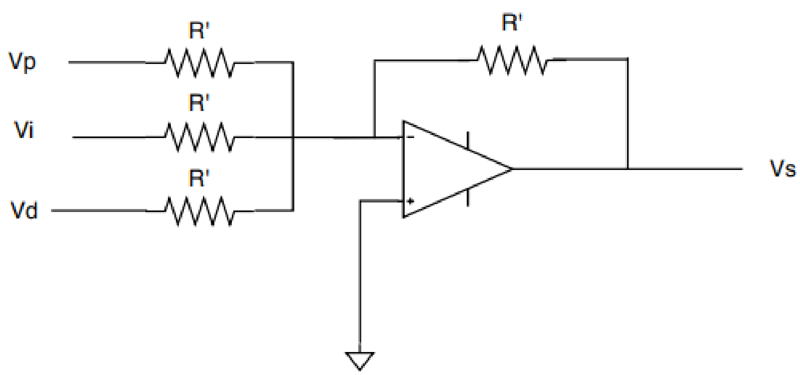


For integral gain, the circuit model is shown below. This is an example of an op amp integrator circuit which has an output of $Vi\ =\ -\frac{1}{R5*C5}\int Verror\ dt$. Therefore, the integral gain coefficient Ki is equal to 1/(R5*C5).



To combine all the signals, a summing amplifier circuit is used, shown below. Assuming the resistances of all the resistors are equal, the output is simply the inverted sum of all the inputs: Vs = – (Vp + Vi + Vd). Therefore, the output is calculated as:

$$Vs = \frac{R2}{R1} Verror \; + \frac{1}{R5 * C5} \int Verror \, dt + R3 * C3 \frac{dVerror}{dt}$$



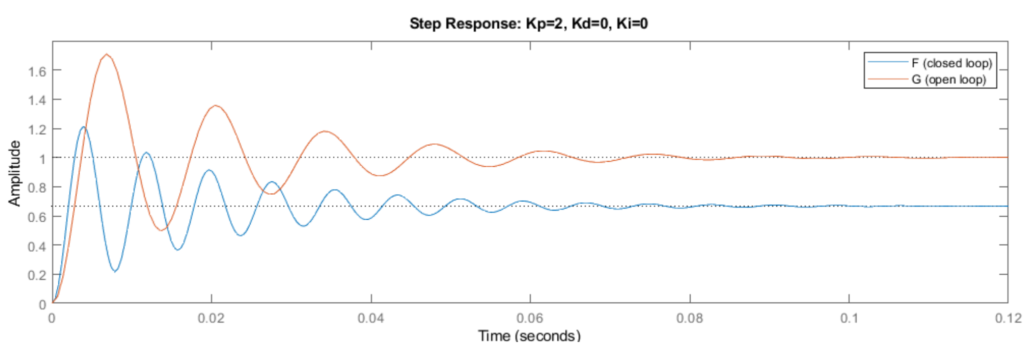
The PID controller gains are summarized here:

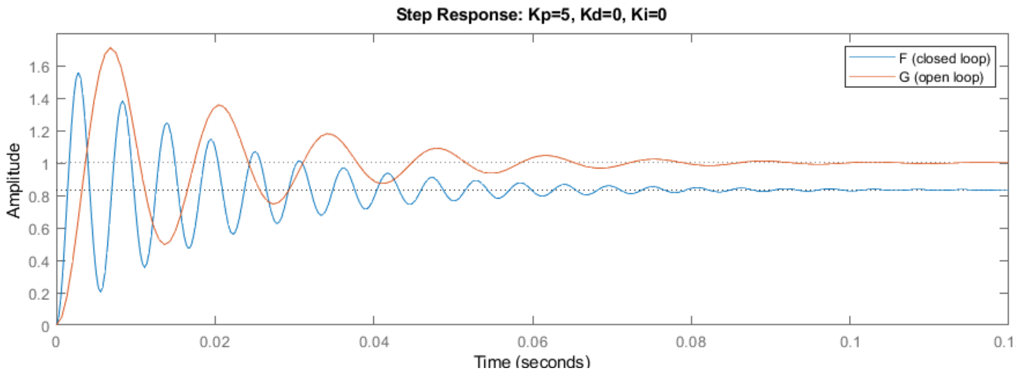
	Time Domain	Laplace Domain
Proportional	$\frac{R2}{R1} Verror(t)$	$\frac{R2}{R1} Verror(s)$
Integral	$\frac{1}{R5 * C5} \int Verror(t) \, dt$	$\frac{1}{R5 * C5} \frac{Verror(s)}{s}$

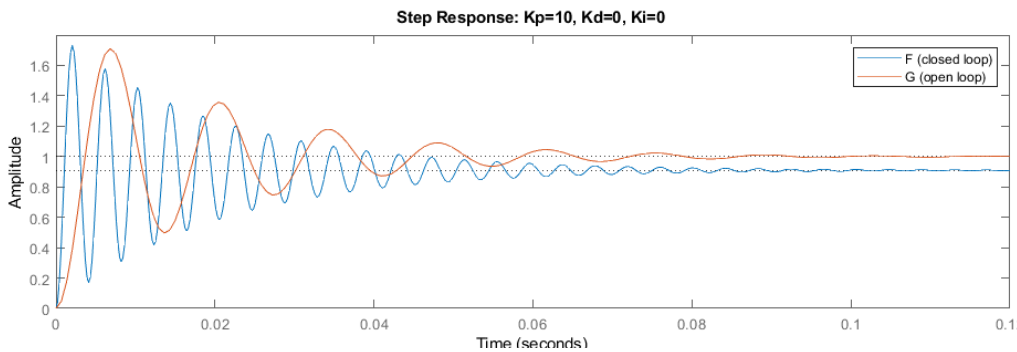
Differential	$R3 * C3 \frac{dVerror(t)}{dt}$	$R3 * C3 * Verror(s) * s$
Sum	$Vs(t) = \frac{R2}{R1} Verror(t) +$ $\frac{1}{R5*C5} \int Verror(t)\, dt +$ $R3 * C3 \frac{dVerror(t)}{dt}$	$\frac{Vs(s)}{Verror(s)} = \frac{R2}{R1} + \frac{1}{R5*C5*s} +$ $R3 * C3 * s$

PID Control

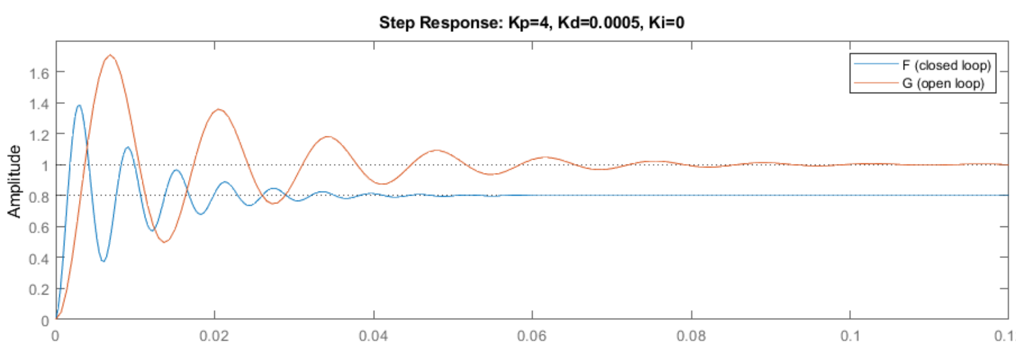
The PID controlled system could then be simulated in Matlab with the file **PIDControl**, first by testing proportional gains. With Ki=Kd=0, proportional gain was tested at Kp = 2, 5, and 10, shown below. Increasing proportional gain tends to reduce steady state error and results in faster response times, but settling time remains constant.

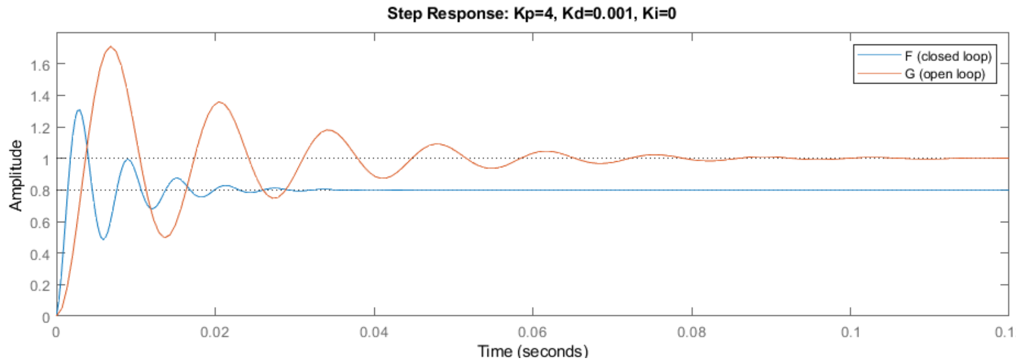


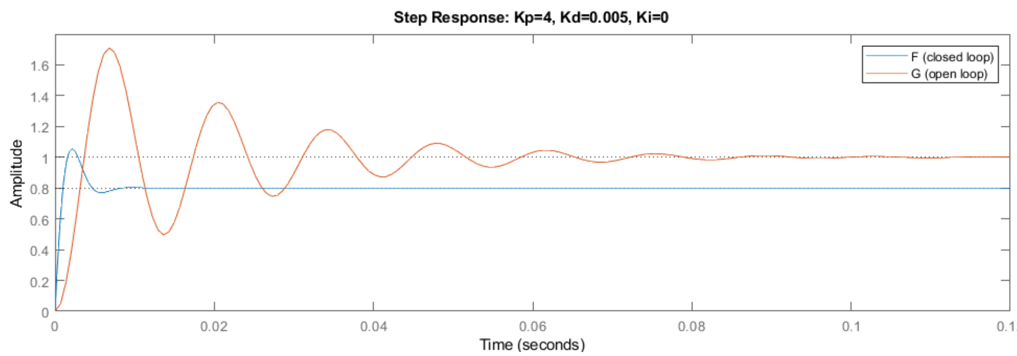




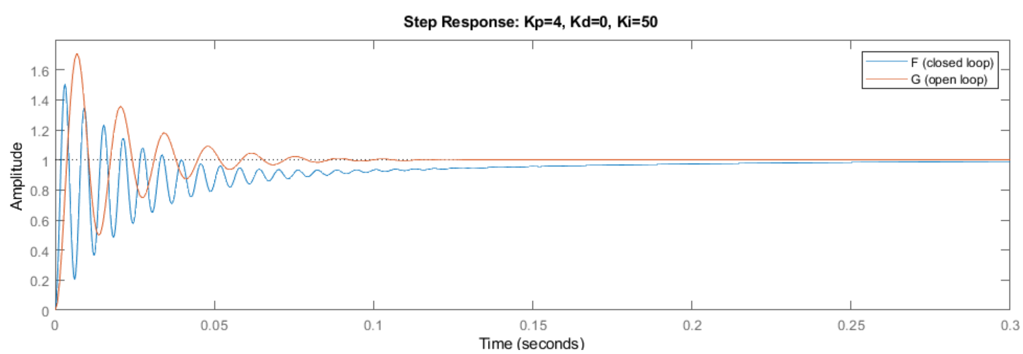
The system was then tested with derivative and proportional control. Leaving Kp=4 and Ki=0, different values for differential gain were tested at Kd = 0.0005, 0.001, and 0.005. Increasing Kd caused a reduction in settling time and overshoot.

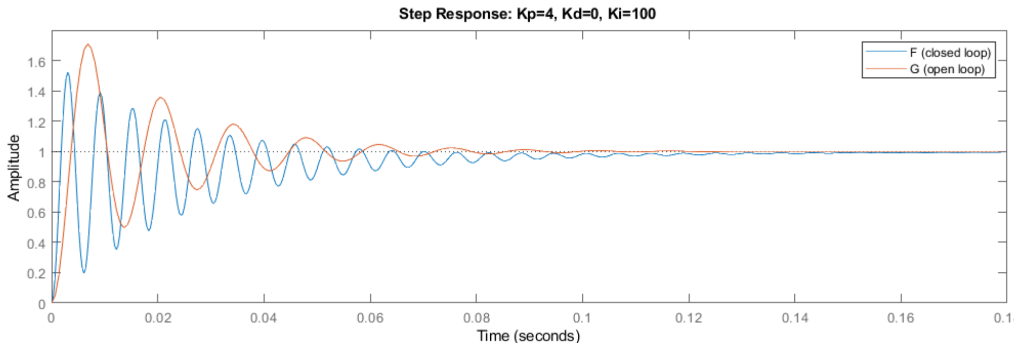


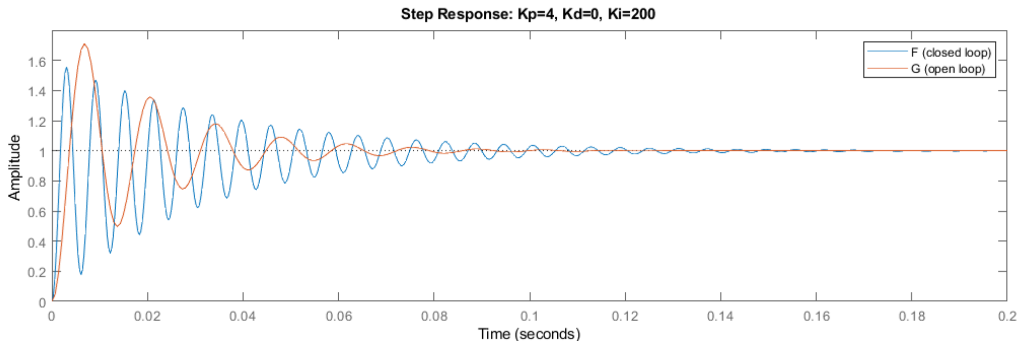




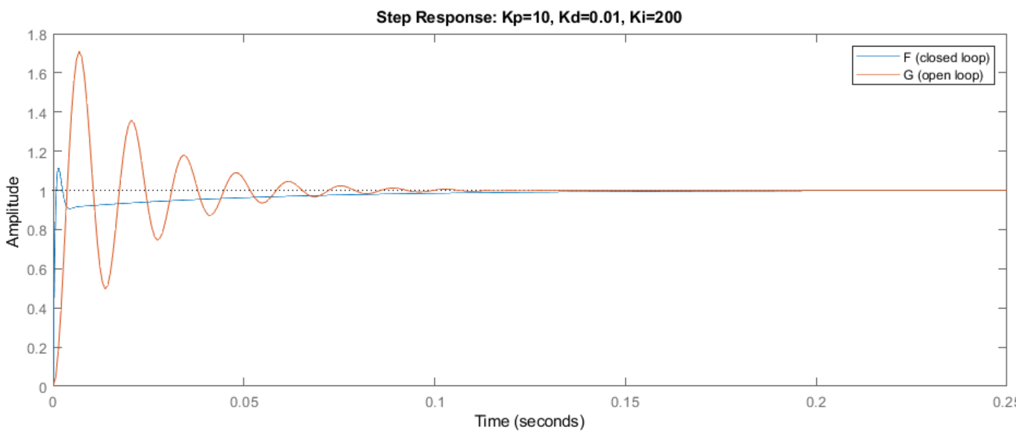
The system was then tested with integral and proportional control. Leaving Kp=4 and Kd=0, different values for integral gain were tested at Ki = 50, 100, and 200 Increasing Ki seems to reduce steady state error but slightly increases the settling time.







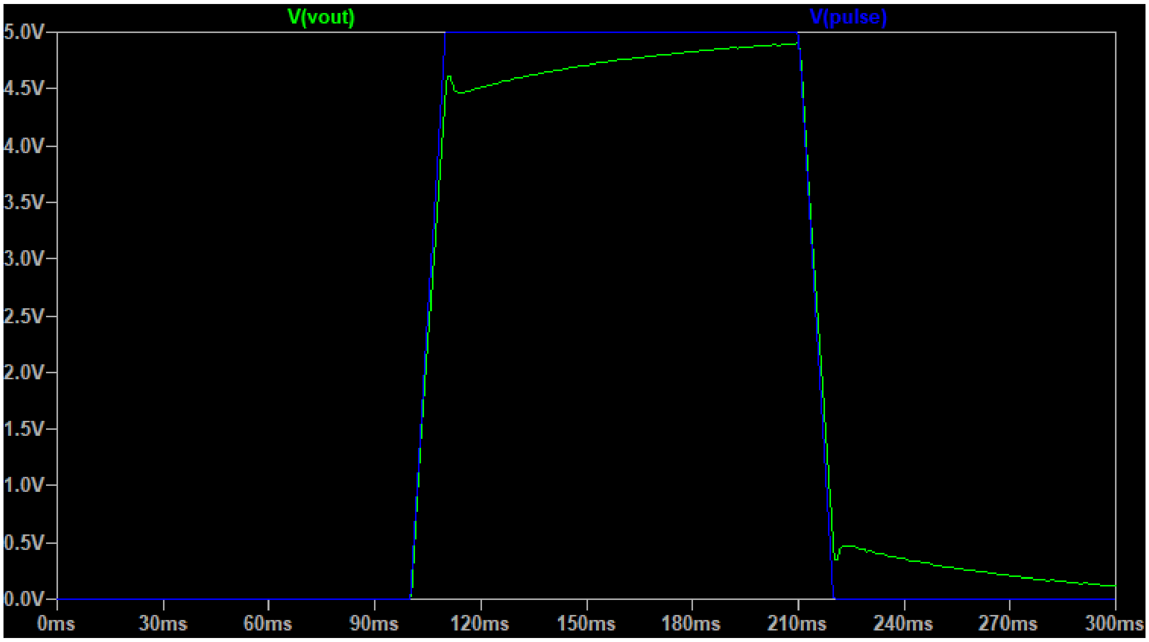
Testing different gain coefficients, a suitable response was found from Kp=10, Kd=0.01, and Ki=200. This response has an overshoot of 10%, an initial steady state error of 0.1, and a 2% settling time of 0.1 seconds.



Poles are: 0 + 0i, –0.0185 + 0i, –1.1046 ± 1.0390i, and –0.05 ± 0.4585i

Next, the system was simulated in LTSpice with the file **PID**. Considering the equations relating gains and component values from before, the resistances and capacitances of the PID components could be determined. LTSpice simulation results are shown below.

R1	R2	R3	C3	R5	C5
1kΩ	10kΩ	10kΩ	1µF	10kΩ	0.5 µF

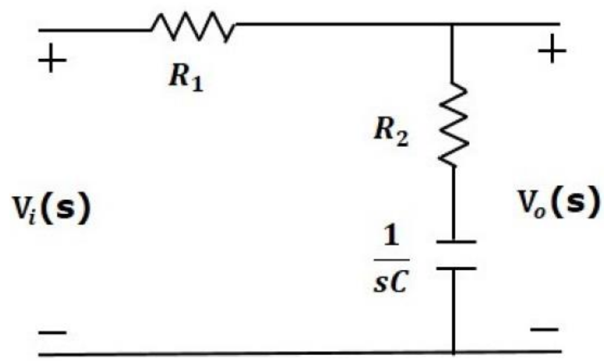


The results from the LTSpice simulation appear similar to the Matlab results. Like Matlab, the LTSpice output voltage has an initial 10% error that gradually settles closer to the desired value. However, the LTSpice response does not have a large overshoot like the Matlab plot, only a small one that doesn't pass the desired value. This may be due to the Matlab being a step input with an amplitude of 1, while the LTSpice simulation has a step response with an amplitude of 5. Also, the step input in LTSpice has a noticeable rise time, while the input in Matlab is instant. The LTSpice shows the same behavior when being turned off, gradually slowing down with a 10% error.

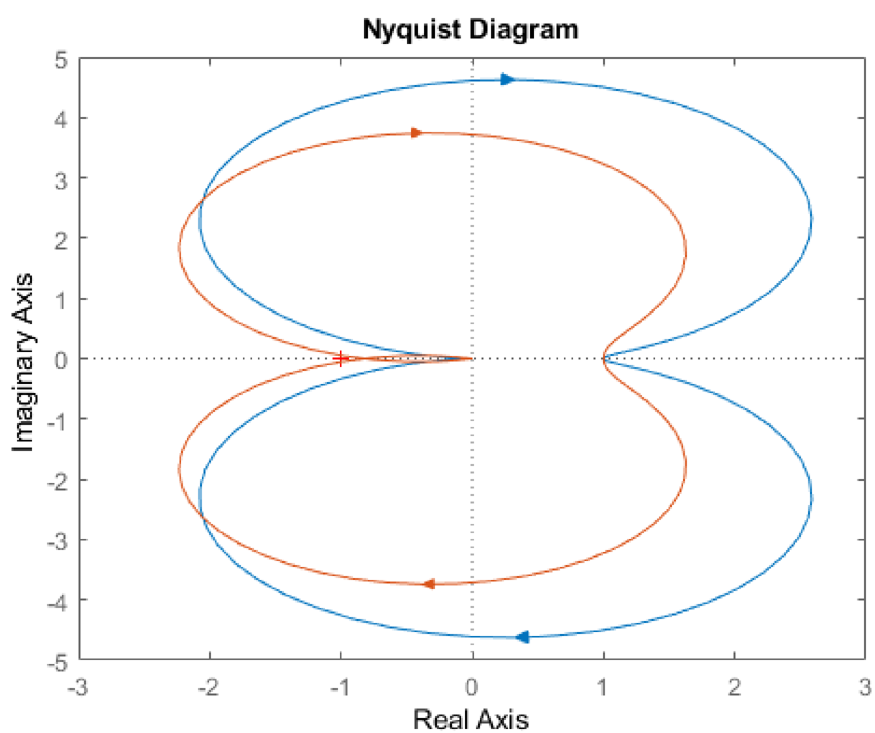
Lead Lag Compensation

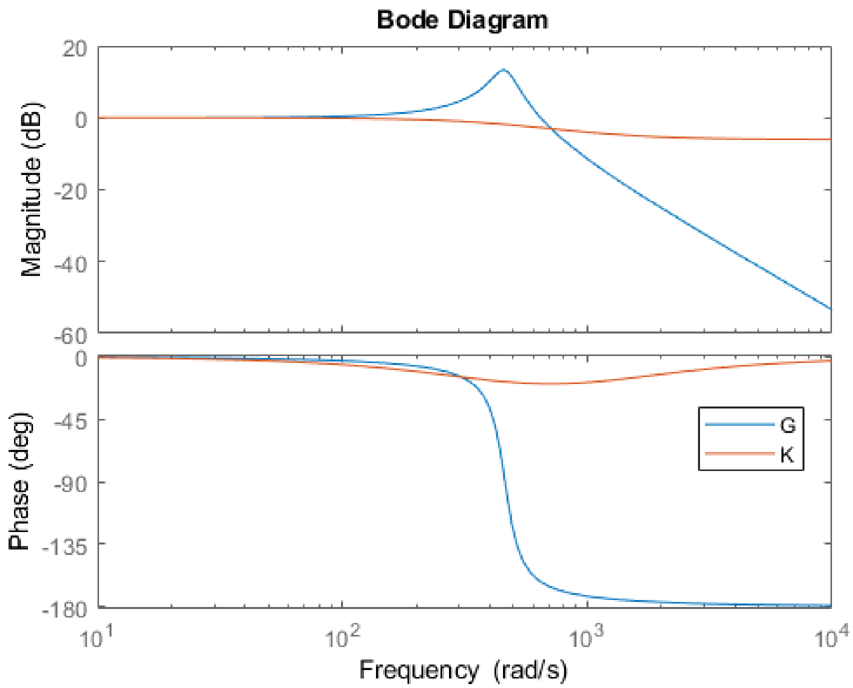
Lag Compensation

We now will consider the effect of Lead-Lag Compensation. The first section we consider is the lag compensator. Its implementation can be seen below.



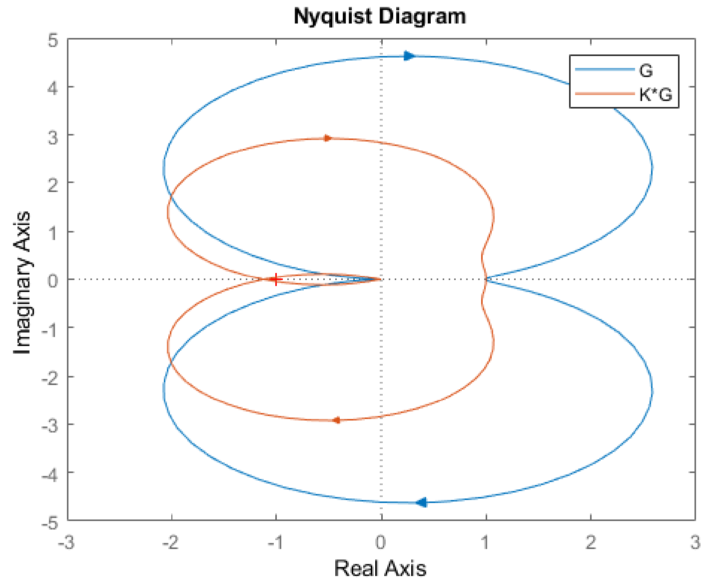
Using the **leadlagcompensator** file provided in Matlab with the values: R1 = R2 = 1k, C2= 1uF, and lead = 1, the following graphs were created.

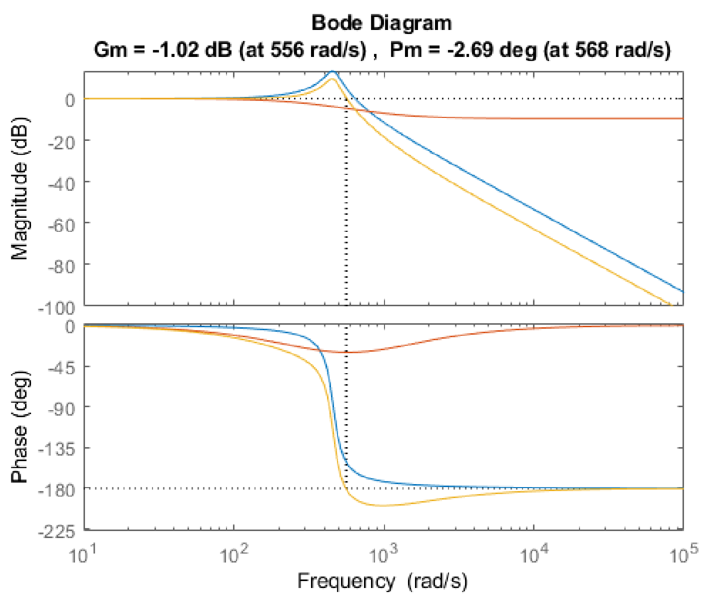




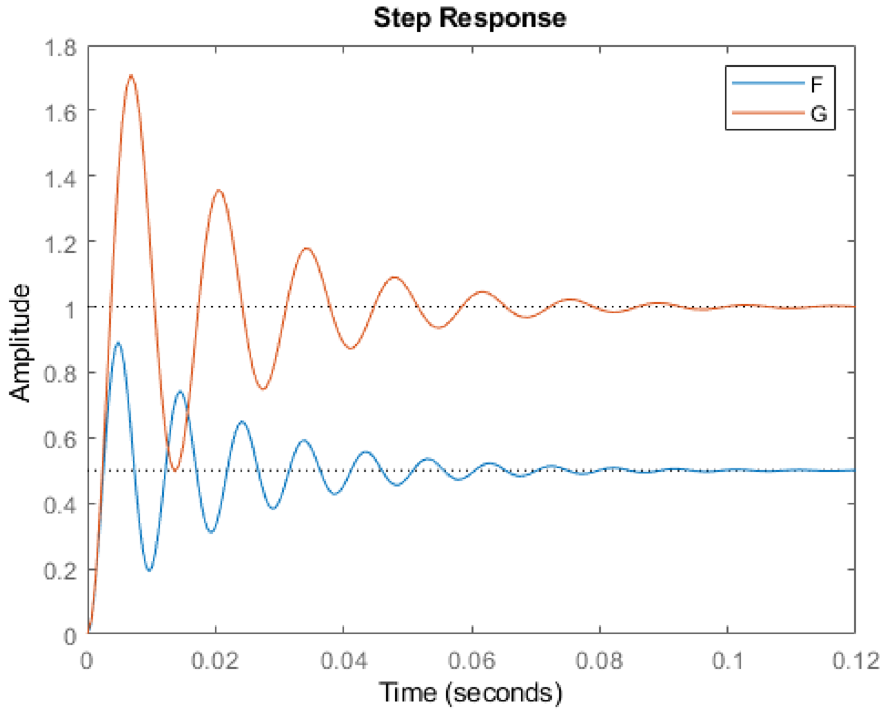
Multiplying G by K has changed the characteristics of the system as the Nyquist plot can now encircle the point at -1. Since the system has no poles in the right-hand plane, whenever the total number of encirclements is not zero, the system is no longer stable. This can also be seen in the bode diagram. Without K, the gain margin was infinite, but is now at a small positive value, as well as this, the phase margin was at 17.6 degrees, but is now at 2.95degrees. K causes the bode diagram to dip reducing the gain margin, and the magnitude to be closer to the phase margin.

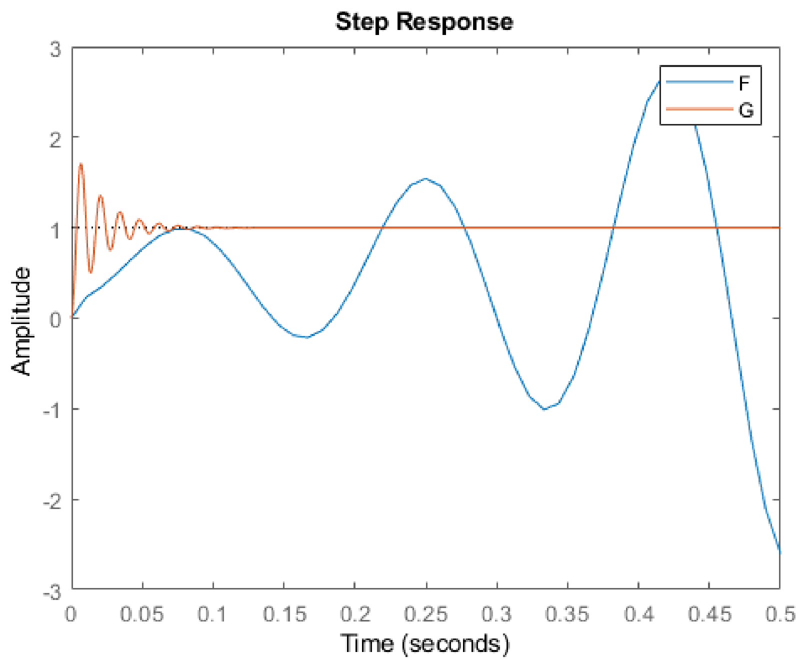
Setting R1 =2K, R2 = 1K, and C2 to 1uF, causes the closed loop behavior to be unstable. The -1 point in the Nyquist diagram is now encircled, and the gain and phase margin are both negative.

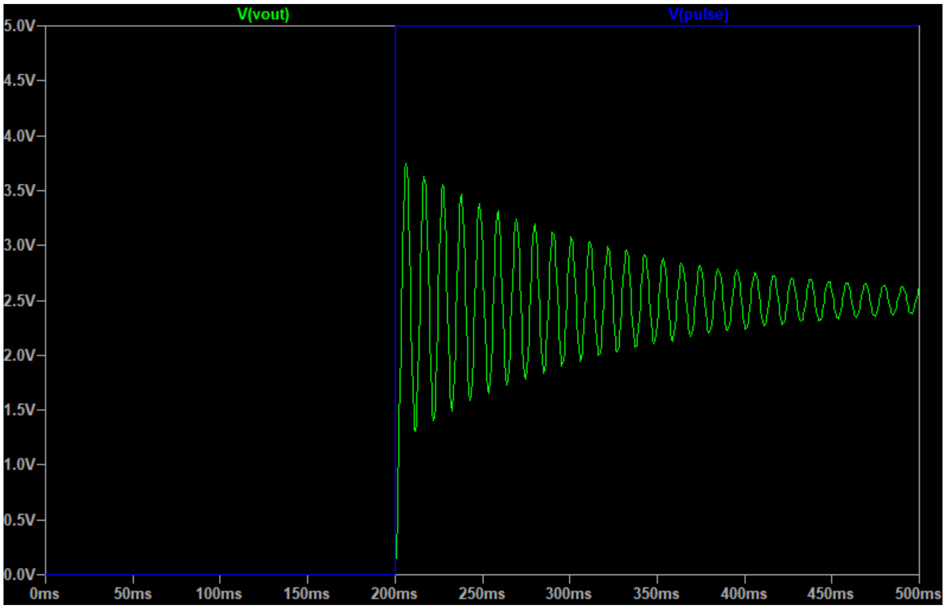


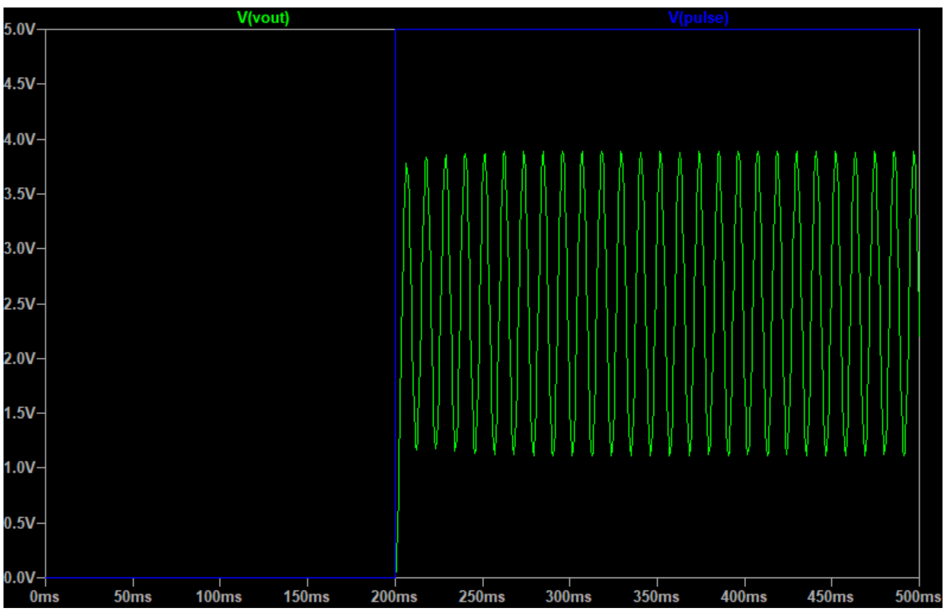


The step responses of the open loop behavior are now compared to the closed loop behavior. The first plots found in Matlab followed by the response produced in the LTSpice file **lag** can be seen below. The first respective image in each set is with both resistors set to 1K, and the second image is with R1 set to 2K.


Step Response
Amplitude
Time (seconds)
F
G
0
0.2
0.4
0.6
0.8
1
1.2
1.4
1.6
1.8
0
0.02
0.04
0.06
0.08
0.1
0.12


Step Response
Amplitude
Time (seconds)
F
G
-3
-2
-1
0
1
2
3
0
0.05
0.1
0.15
0.2
0.25
0.3
0.35
0.4
0.45
0.5

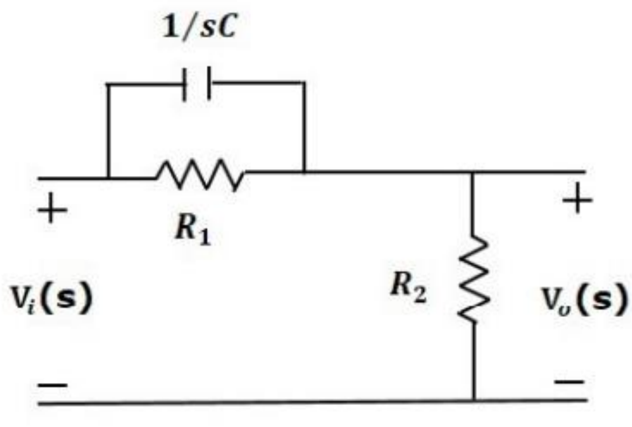




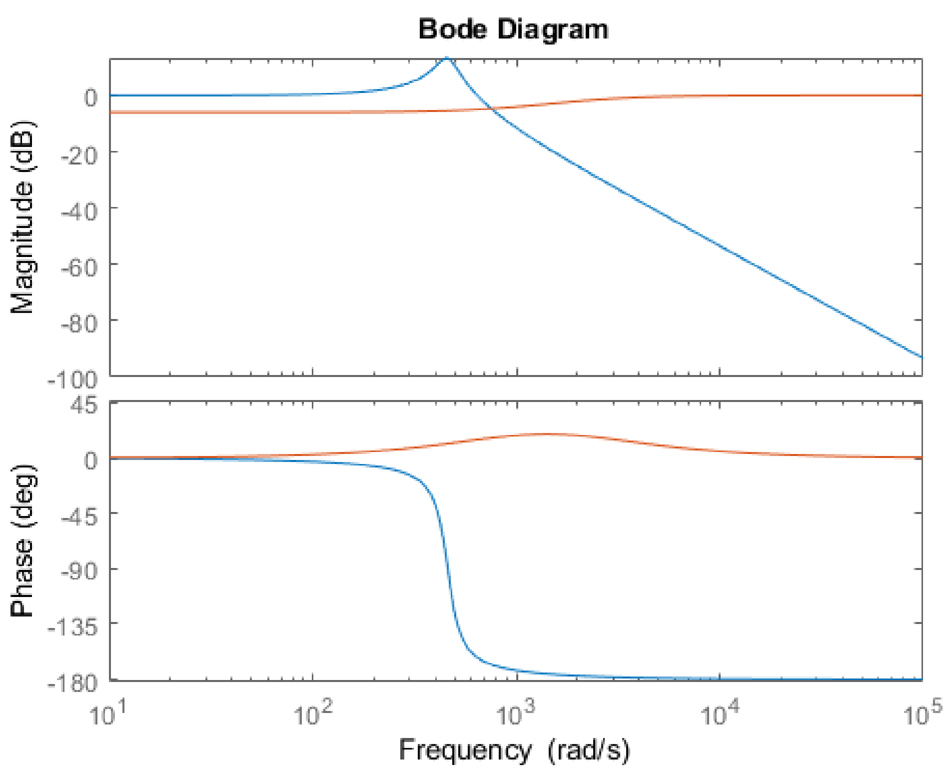
Both Matlab and LTSpice show that increasing the value of R1 causes the system to become unstable. The lag controller may be useful when wanting to improve transients and steady state error, but we must be careful not to allow the system to become unstable in the process.

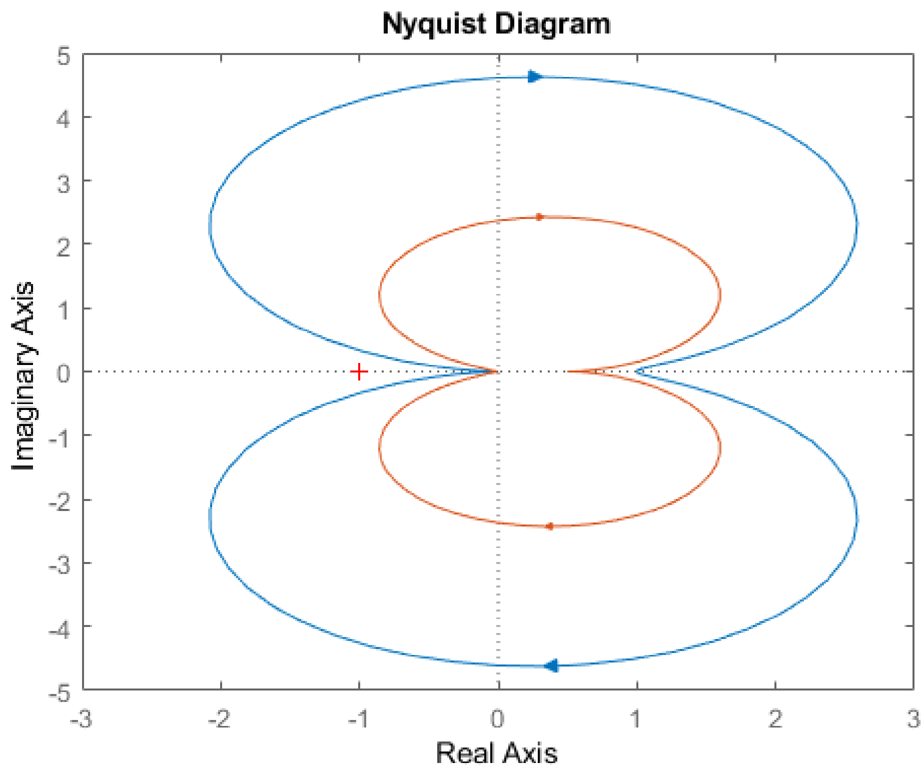
Lead Compensation

We now consider the behavior of the lead compensator. Its implementation can be seen below.



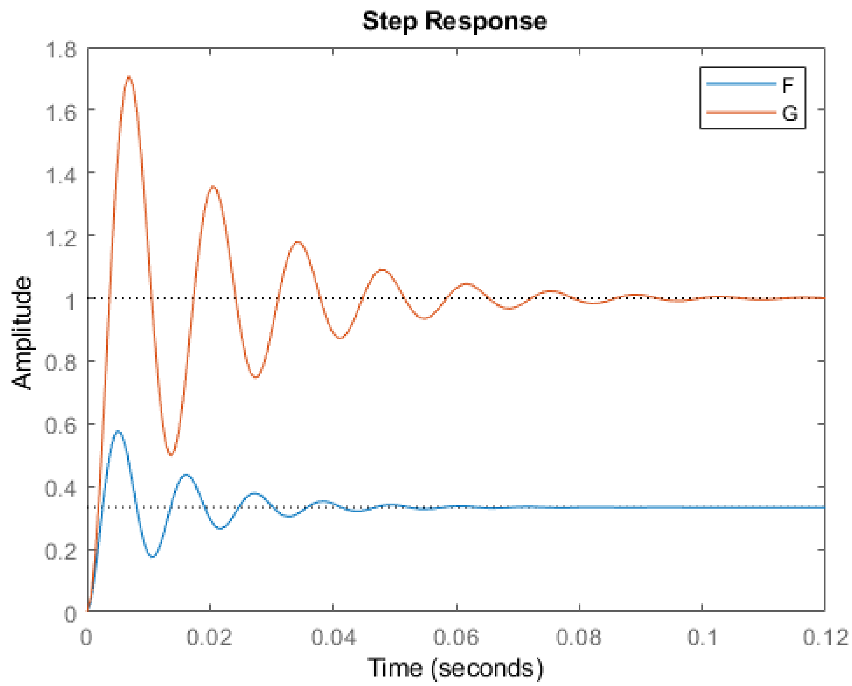
Again, setting the resistor values to 1K and the capacitor to 1uF in Matlab produces the following.





The Nyquist plot shows improved stability after including the lead compensator (seen in orange). The point at -1 is now further away from being encircled. The bode plots show that the magnitude and phase are increased by the inclusion of the lead compensator. While the gain margin remains infinite, the phase margin increased from 17.6 deg to 42.2 degrees.

The image below is the step response of the lead compensator. The setting time has improved, but the steady state error increased.

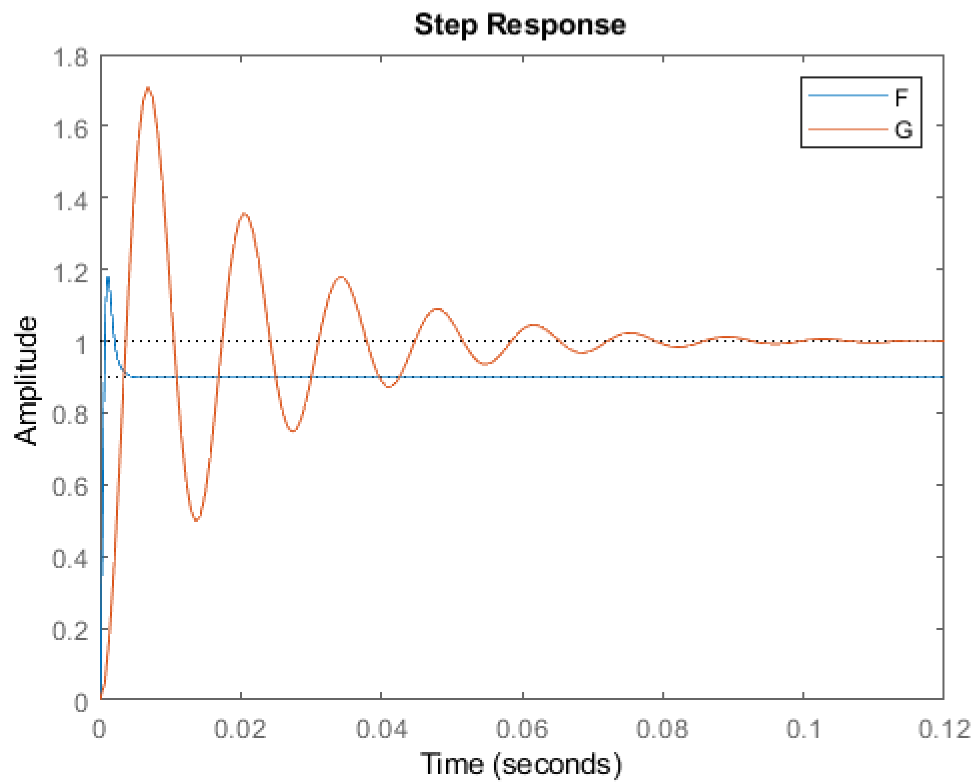


Lead-Lag Compensation

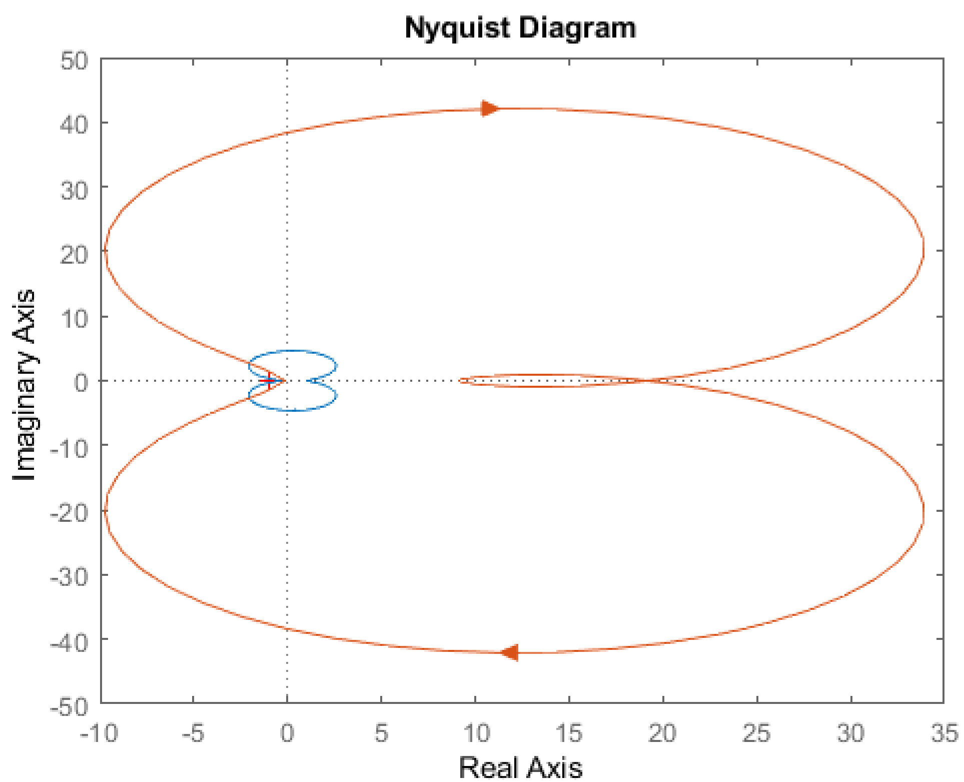
With the effects of both lead and lag compensators known, they can be combined to form a controller capable of improving the response of the system. Adjusting the parameters in the Matlab file to find a good response gave us the values:

R2	C2	R1	R3	C3	R4	Kp
2kΩ	1μF	1kΩ	2kΩ	1μF	200Ω	100

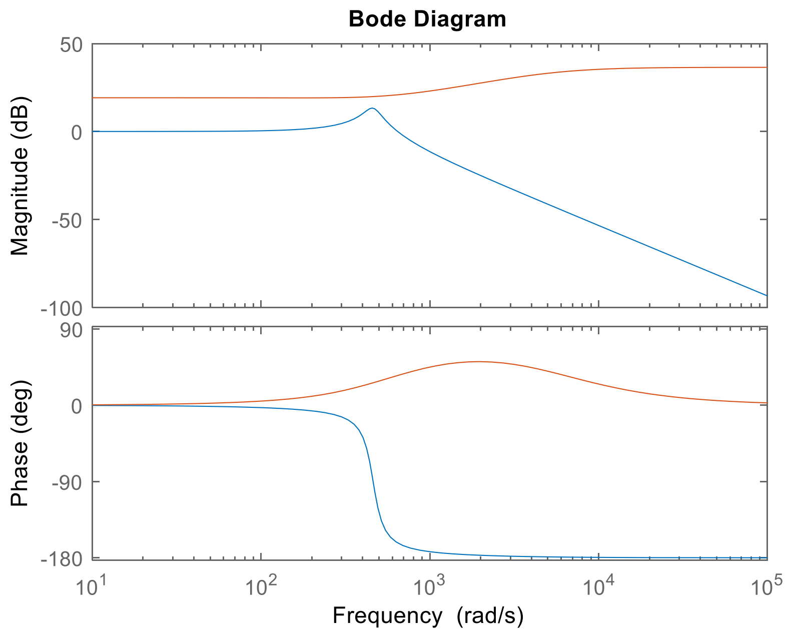
These parameters resulted in a step response with 10% steady state error, 20% overshoot, and a 2% settling time of about 0.003s. This system has a phase margin of 53 degrees. The resulting plots are as follows.



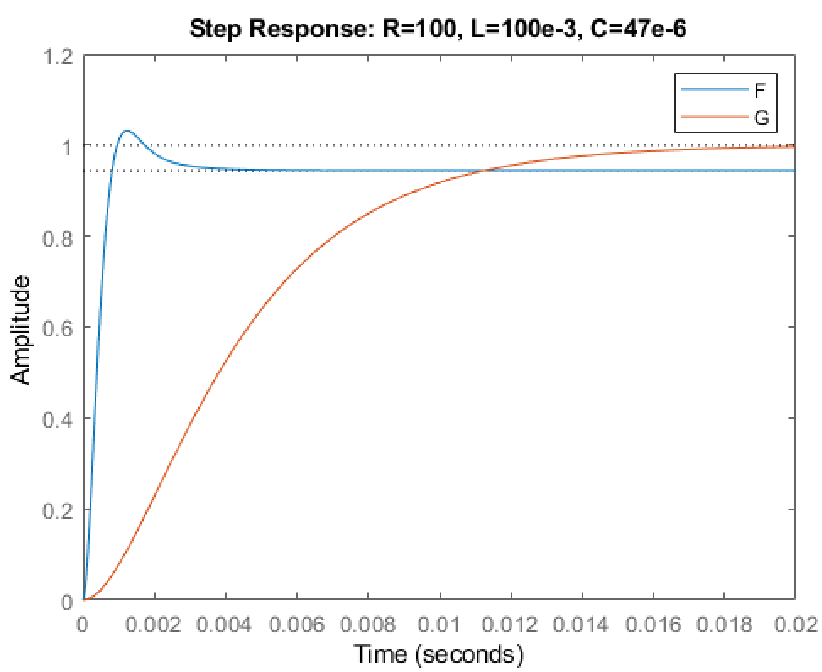
Step Response
F
G
Amplitude
Time (seconds)
1.8
1.6
1.4
1.2
1
0.8
0.6
0.4
0.2
0
0
0.02
0.04
0.06
0.08
0.1
0.12

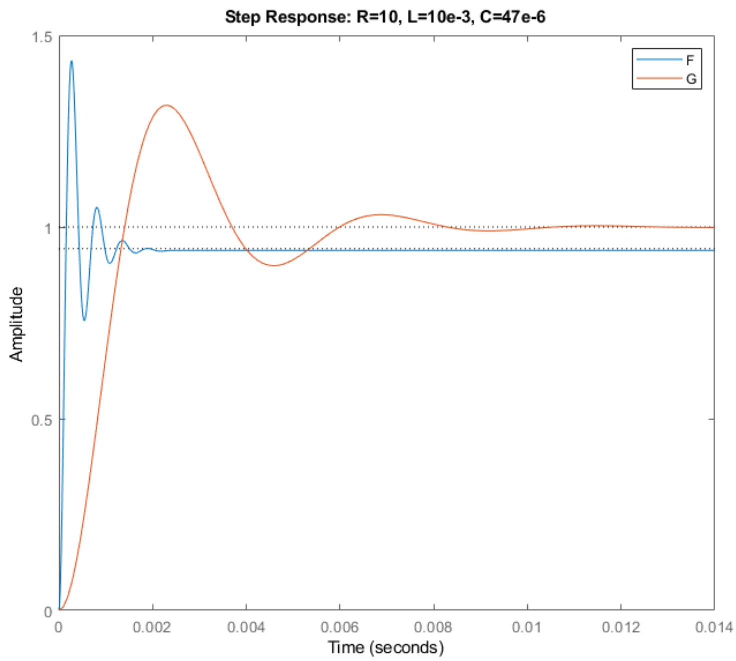


Nyquist Diagram
Imaginary Axis
Real Axis
50
40
30
20
10
0
-10
-20
-30
-40
-50
-10
-5
0
5
10
15
20
25
30
35

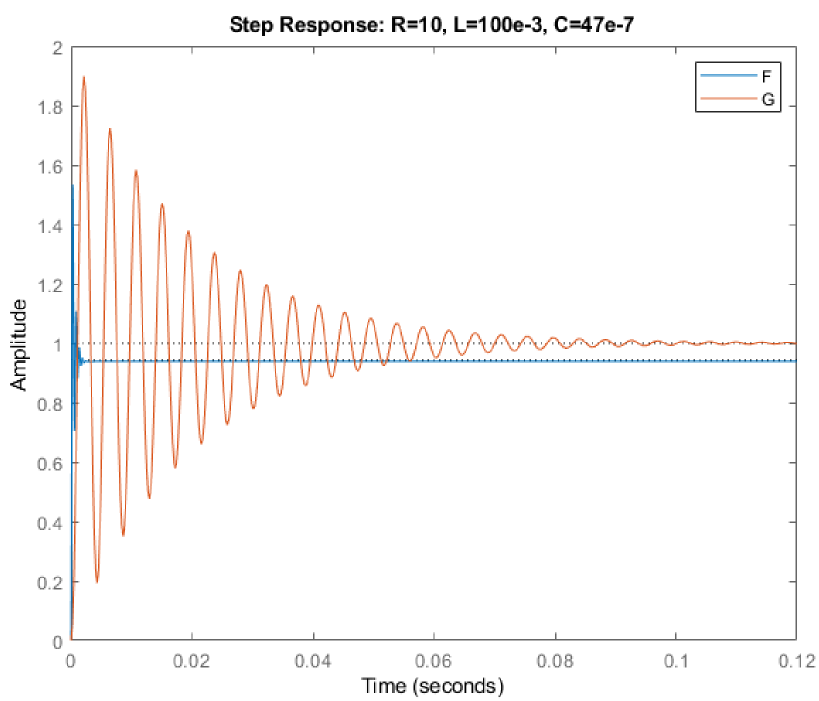


The step response of the lead-lag compensator to three different cases can be seen below. The responses are acceptable for varying RLC values, and for this reason, we conclude that the design is fairly robust. If necessary, the values of the compensator could be tweaked to meet specific requirements.



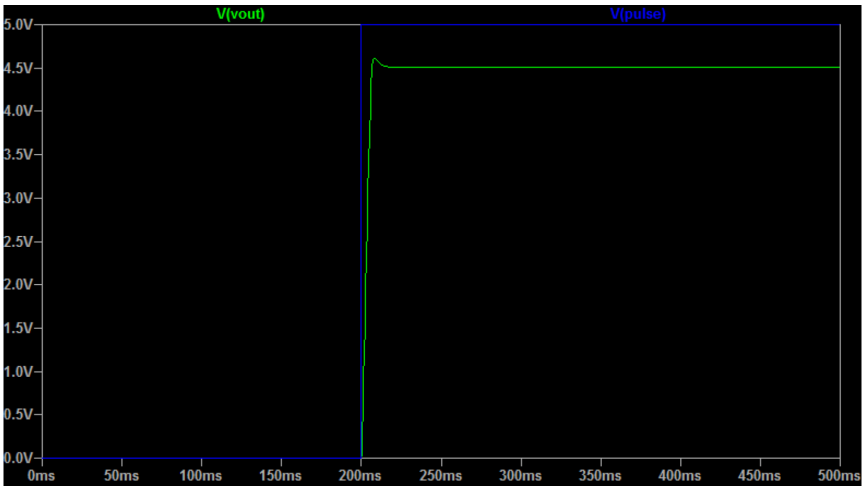


Step Response: R=10, L=10e-3, C=47e-6
F
G
Amplitude
Time (seconds)
0
0.5
1
1.5
0.002
0.004
0.006
0.008
0.01
0.012
0.014



Step Response: R=10, L=100e-3, C=47e-7
F
G
Amplitude
Time (seconds)
0
0.2
0.4
0.6
0.8
1
1.2
1.4
1.6
1.8
2
0.02
0.04
0.06
0.08
0.1
0.12

With the values of the components found, the circuit could be simulated on LTSpice, which gives the step response shown below. Similar to the Matlab simulation, the response has a 10% steady state error and a fast response time, but only a small overshoot. The small overshoot is similar to the results of the PID controller and again may be due to the difference in the magnitude of the step input.



Discussion

This project has shown the practical importance of negative feedback in designing controllers. It has shown the process of modelling a system, analyzing its behavior, and improving it using various methods. Both PID and lead-lag control are flexible options when modifying transients to achieve speed and steady state error requirements.

Matlab and LTSpice were beneficial in different ways but useful when used together. LTSpice was good for simulating realistic circuit behavior, and Matlab was good for quickly modelling systems while being able to see step responses, bode plots, and Nyquist diagrams.

A major benefit of PID control was its simplicity. There was less emphasis on considering factors such as phase and gain margins when designing the PID controller. However, varying the PID gains and finding components afterwards made choosing realistic component values slightly difficult. If there were other constraints such as cost and size, designing the PID controller would be more challenging.

Designing the lead lag controller allowed us to see the effects changing the phase and gain margins on the closed loop behavior. Theoretically, finding good values seemed simple, but choosing between increasing the proportional gain and tweaking the lag component was not immediately obvious. Designing the lead-lag controller required a model without poles in the RHP, so that the system would remain stable. However, the final design proved robust.

Conclusion

By simulating a negative feedback electrical system in Matlab and LTSpice, we were able to determine the effects of different controllers on the response. Through PID control in Matlab, suitable values for each gain were found that produced an improved step response. These values relate to the components of the circuit and could be simulated in LTSpice. It was found that proportional gain improves steady state error and response times, integral gain also reduced steady state error, and differential gain reduces overshoot and settling time. PID control is a useful way of improving a system's performance, although finding appropriate gain values can be troublesome. Using a lead lag compensator in the system caused it to produce a suitable step response. It was found that a lag compensator improved response time but had the potential to make the system unstable. A lead compensator improved settling time and stability but increased steady state error. Combining them together, the lead-lag compensator resulted in an improved step response which was fairly robust when tested with different RLC values. Lead-lag compensators are another useful way of improving the response of a system, though like PID control, the proper parameters can be hard to find. The methods used in this project were helpful in understanding the design and operation of closed loop feedback systems. Both PID control and lead-lag compensation were successful in improving the performance of the system.

References

[1] “PID controller,” Wikipedia, 21-Jul-2019. [Online]. Available: https://en.wikipedia.org/wiki/PID_controller. [Accessed: August 1, 2019].

[2] M. Narimani, “MSE 381 – Feedback Control Course Project”, from MSE 381. SFU, 2019. [Online]. Available: canvas.sfu.ca. [Accessed: August 3, 2019].