

Homework 2

Course: Automatic Control Systems – ASEN 5114-001 – Spring
2025

Professor: Dale Lawrence

Teaching Assistant: Anantha Dhruva

Steve Gillet

February 9, 2025

"Use the plant and controller models from Homework 1 to explore the effect of control gains on closed loop system poles, and hence the effect on the closed loop natural response of the load shaft angle."

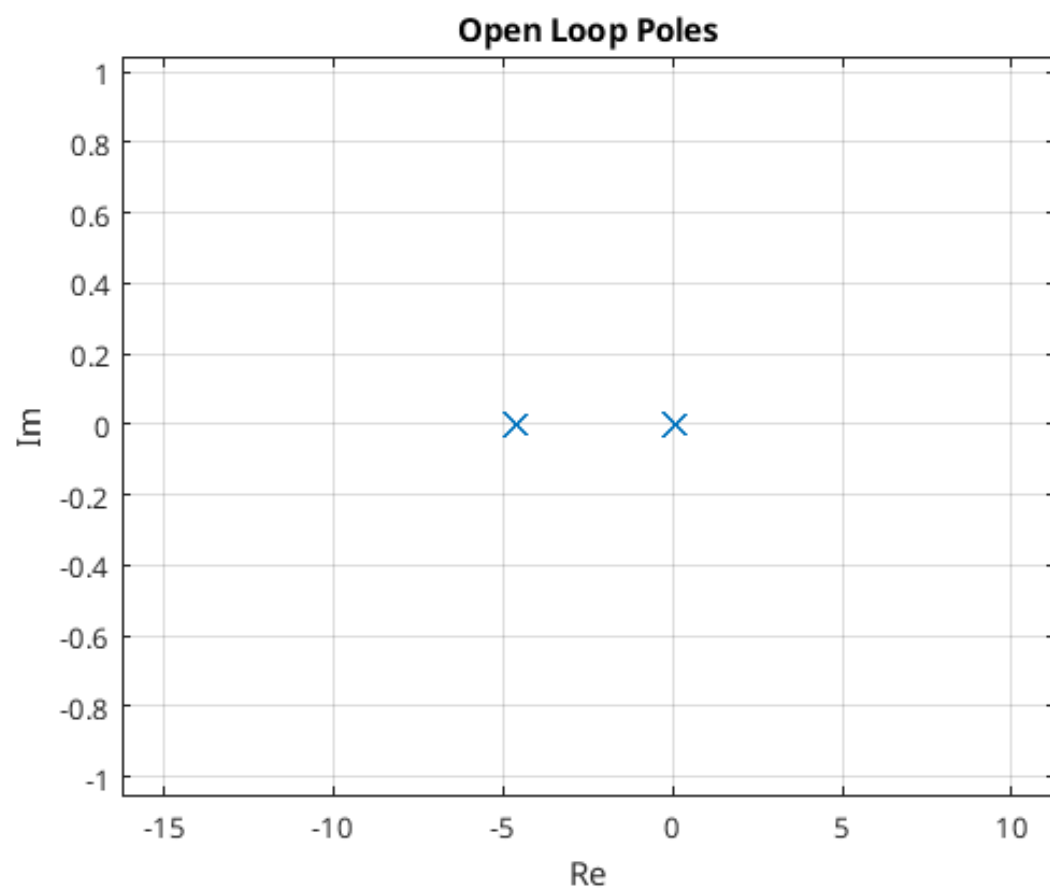
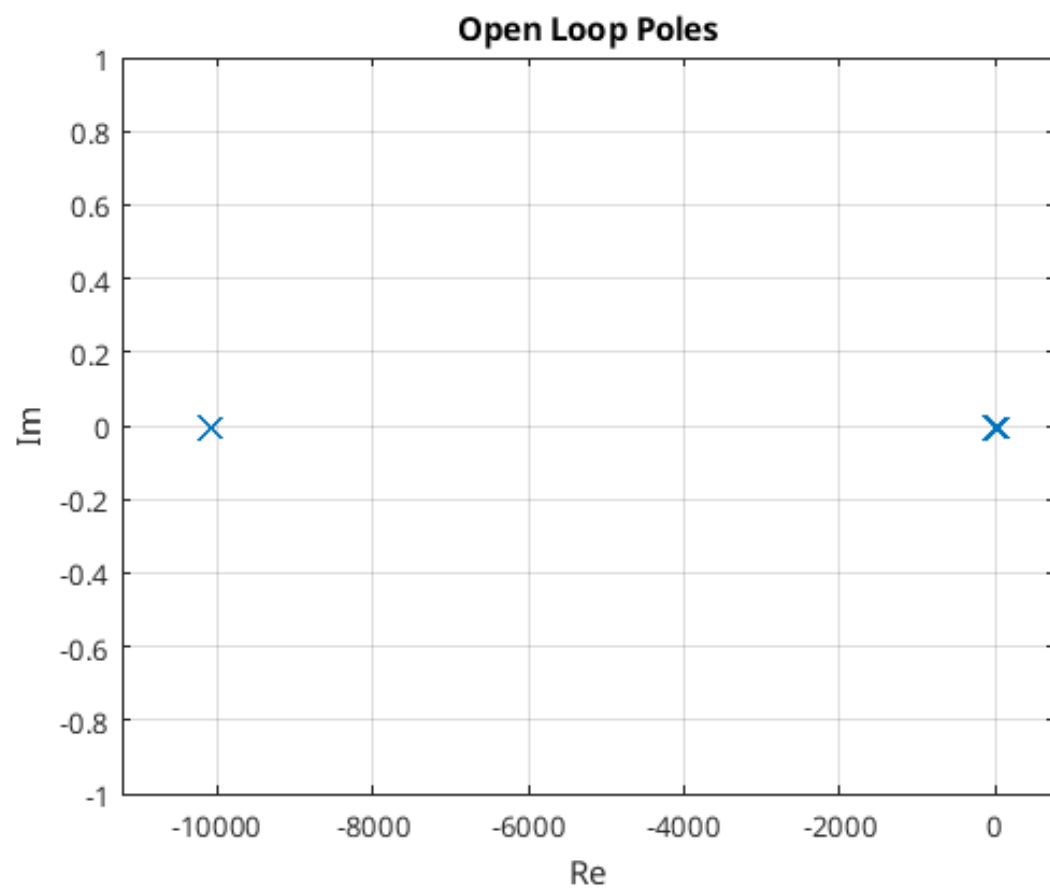
1.

"Find the poles of the plant from your transfer function model from Experiment 1. Plot these in the complex plane."

This is the Matlab code I used to get the roots of my Open Loop Transfer Function using 'roots' Matlab function.

```
1 coeff3=0.000028291; % [Vs^3]
2 coeff2=0.285890679; % [Vs^2]
3 coeff1=1.332; % [Vs/rad]
4 num = [-1];
5 den = [coeff3, coeff2, coeff1, 0];
6 format longG;
7 poles = roots(den);
8 disp(poles);
9
10 figure;
11 plot(real(poles), imag(poles), 'x', 'MarkerSize', 10, 'LineWidth', 2);
12 title('Open Loop Poles');
13 xlabel('Re');
14 ylabel('Im');
15 grid on;
```

You can just feed the coefficients into that function and it will get the roots of that polynomial for you and then I graphed the real and imaginary components using 'real' and 'imag' functions and the results are below.



I had to take one picture zoomed out and then one zoomed in because the roots were 0, -4.66127366088196, and -10100.6965786596 and when the graph is zoomed out enough to see -10,100 you can't distinguish 0 and -4.66.

2.

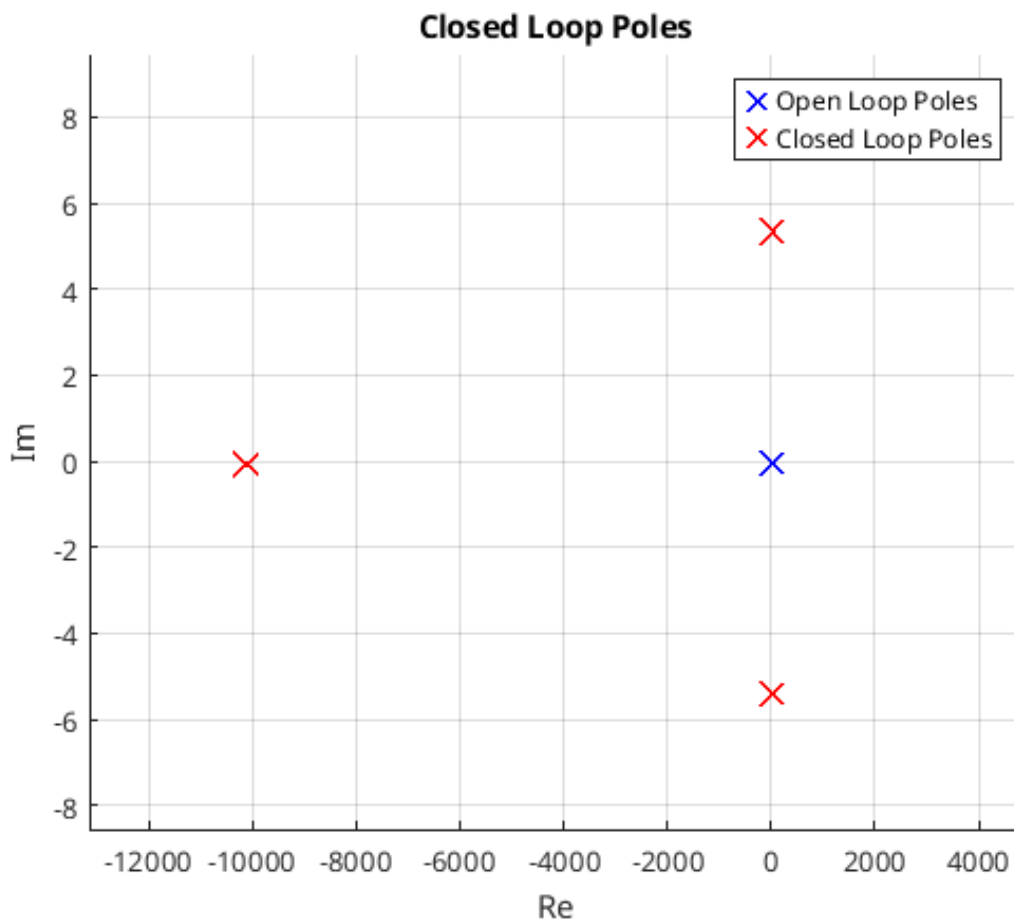
"Find the poles of the closed loop system from part 5 of Homework 1. Plot these on the same plot as in part 1. above."

I used the same technique as for the open loop except the gains were added in of course.

```

1 Gp = -10;
2 Gd = -0.1;
3 numCL = [-Gd -Gp];
4 denCL = [coeff3, coeff2, coeff1 - Gd, -Gp];
5 polesCL = roots(denCL);
6 disp(polesCL);
7
8 plot(real(polesCL), imag(polesCL), 'rx', 'MarkerSize', 10, 'LineWidth', 2);
9 legend('Open Loop Poles', 'Closed Loop Poles');
```

I plotted the closed loop poles (red) on top of the open loop poles (blue) and the results are below.



The -10100 open loop pole is still there but you have to zoom in to see it since the closed loop is basically right on top of it.

3.

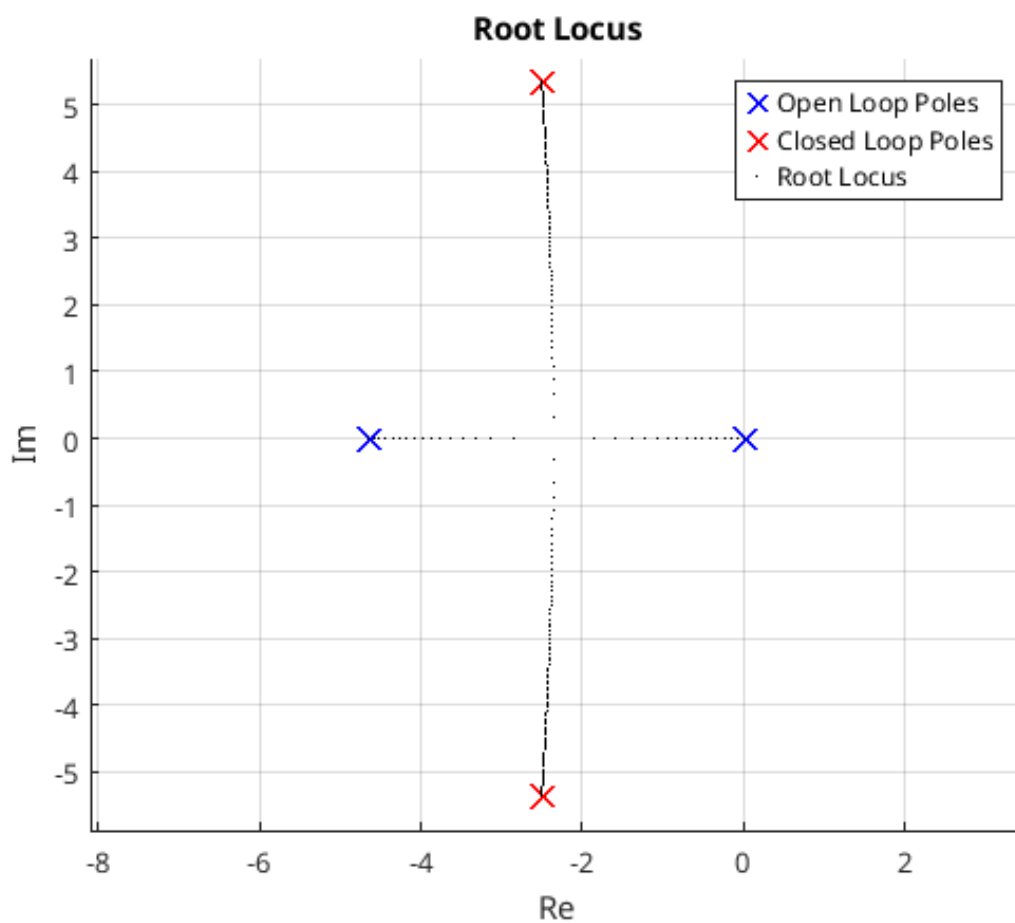
"Repeat part 2. above, using a gain factor g in the control law that varies between 0 and 1, in 100 steps. Plot this 'root locus with respect to g ' on the same plot as above. Describe the effect of this control system

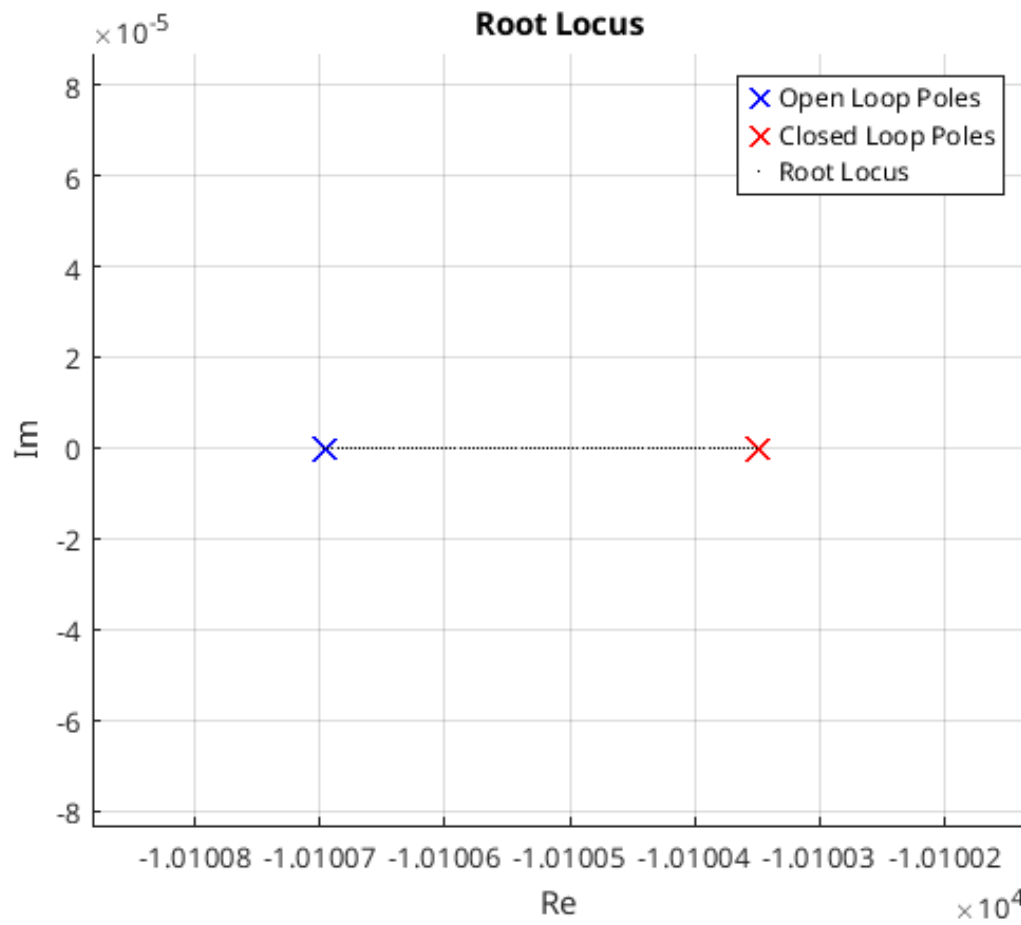
gain on the closed loop poles.”

I accomplished this with a scaling variable and a for loop.

```
1 gScalar = linspace(0, 1, 100);
2 for i = 1:length(gScalar)
3     g = gScalar(i);
4     denCL = [coeff3, coeff2, coeff1 - Gd*g, -Gp*g];
5     polesCL = roots(denCL);
6     plot(real(polesCL), imag(polesCL), 'k.', 'MarkerSize', 1);
7     disp(polesCL);
8 end
9
10 legend('Open Loop Poles', 'Closed Loop Poles', 'Root Locus');
```

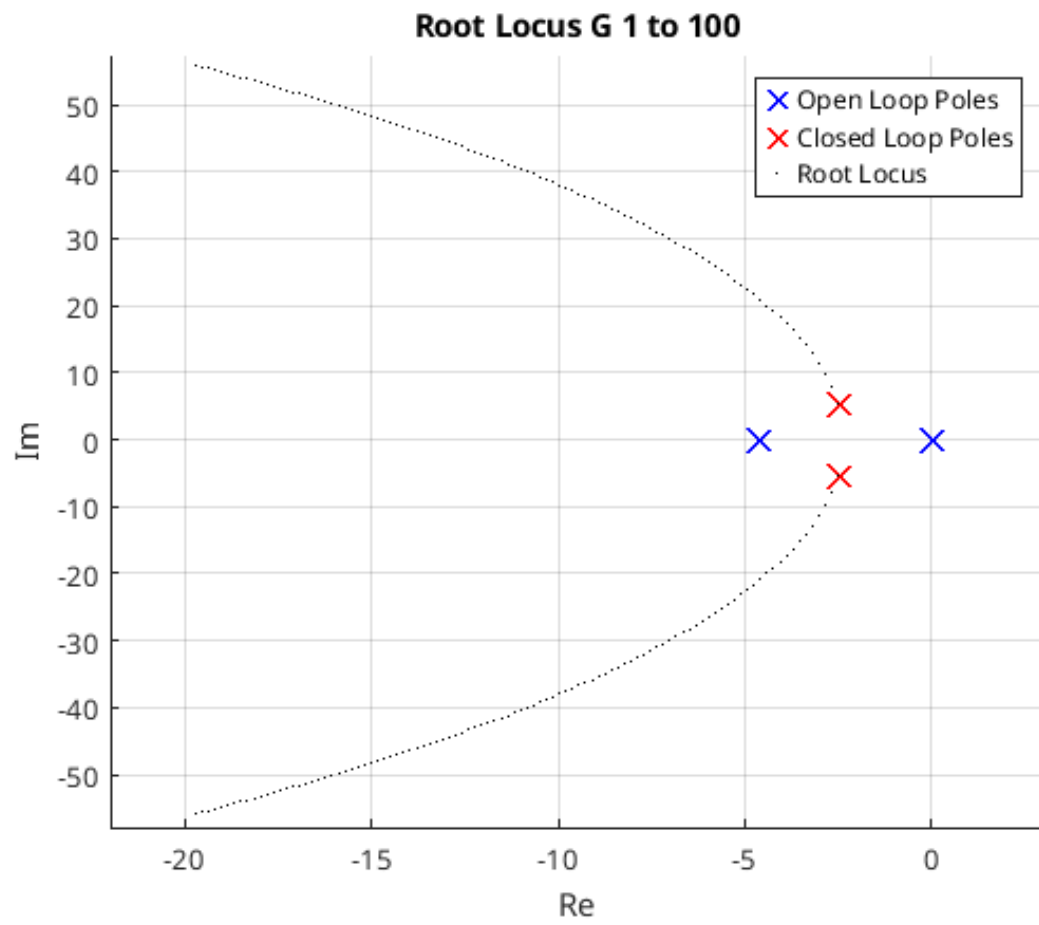
What you see on the plot and in the values of the poles is that when the gains are scaled to 0 the closed loop poles become the open loop poles and then as you scale them up to their full value they move out towards their closed loop values.

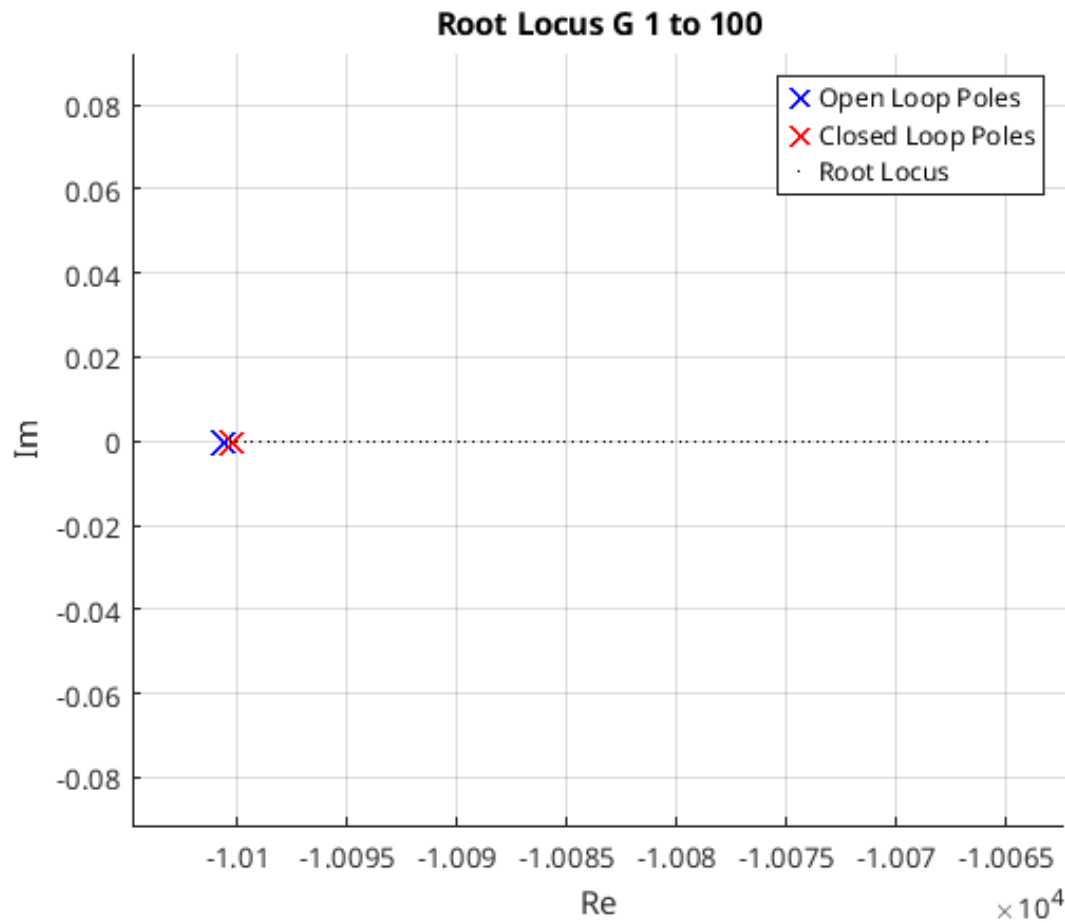




4.

"Repeat part 3. above but use gain values g between 1 and 100. Describe the effect on the closed loop poles."

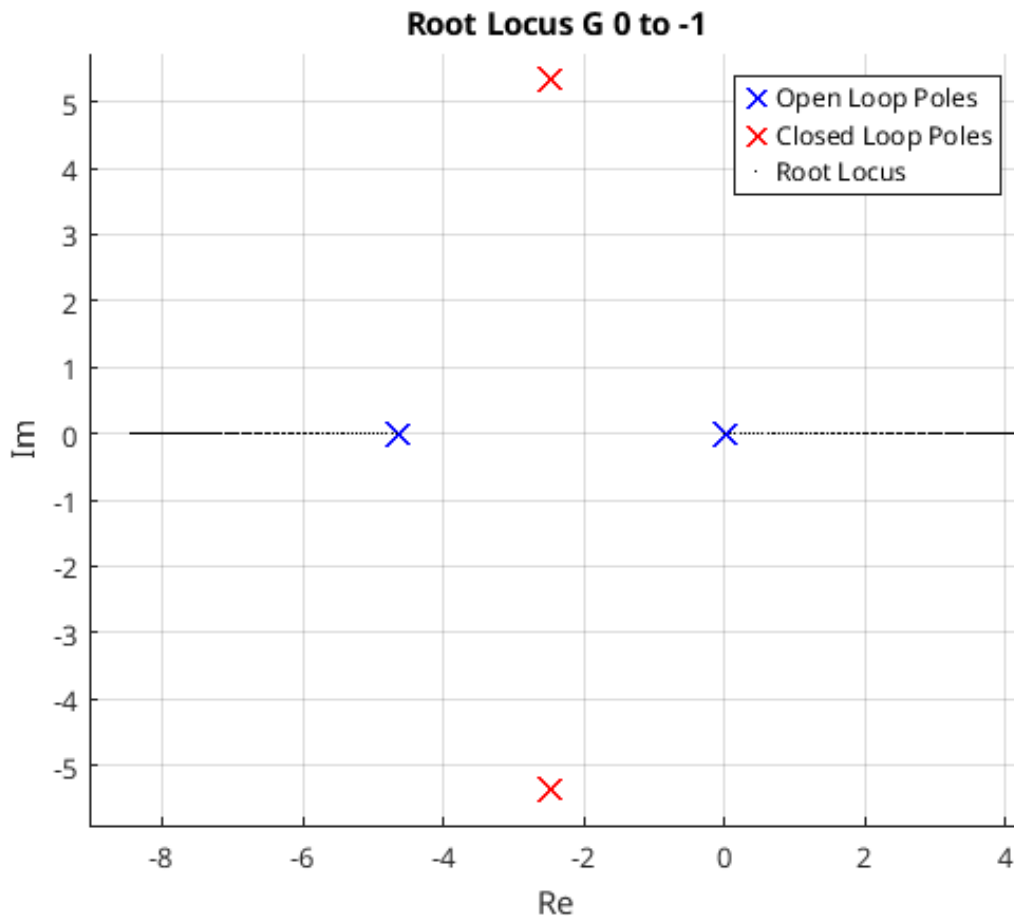




The imaginary poles grew larger in the imaginary axis and more negative in the real axis. This is going to have the effect of increasing the frequency of the response and decreasing the settling time. The real pole also shifted slightly towards the origin as the gains increased, although it was very small I don't think this means that it would ever actually go towards the origin or converge to the other poles.

5.

"Repeat part 4. but use negative g values between 0 and -1."

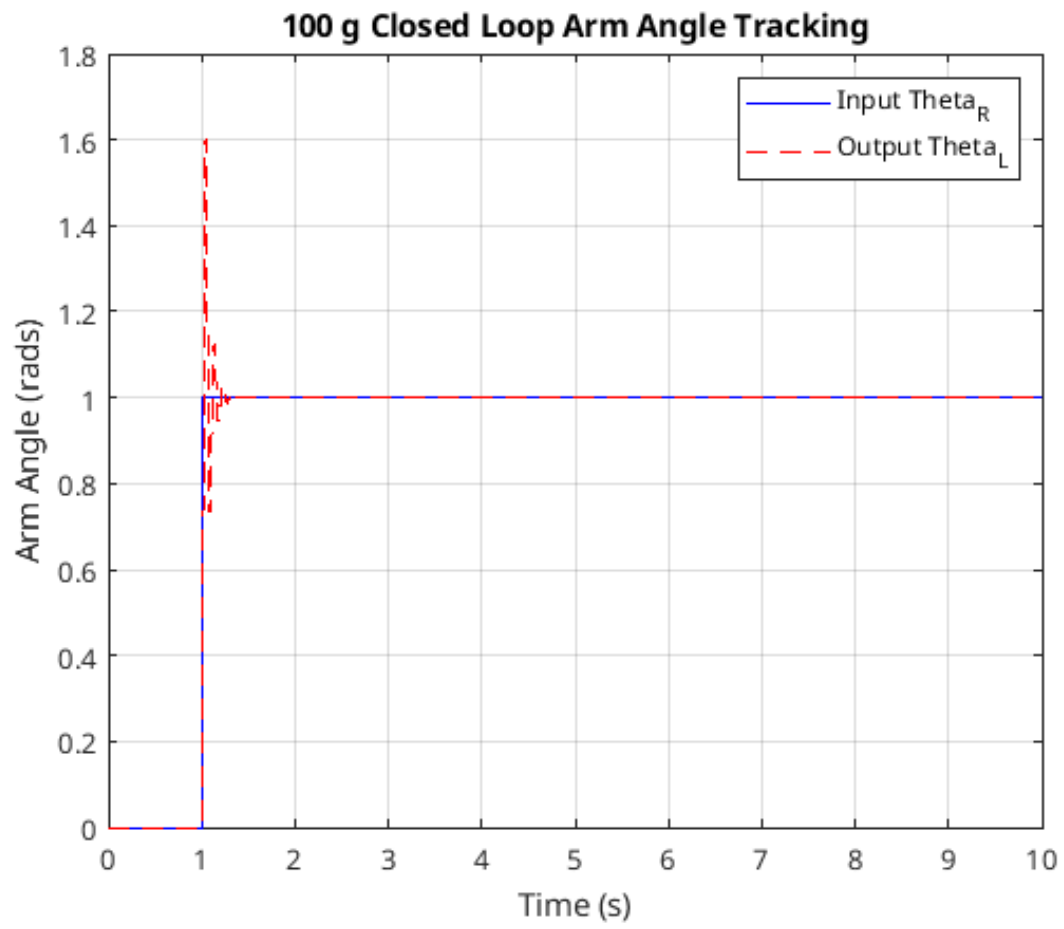


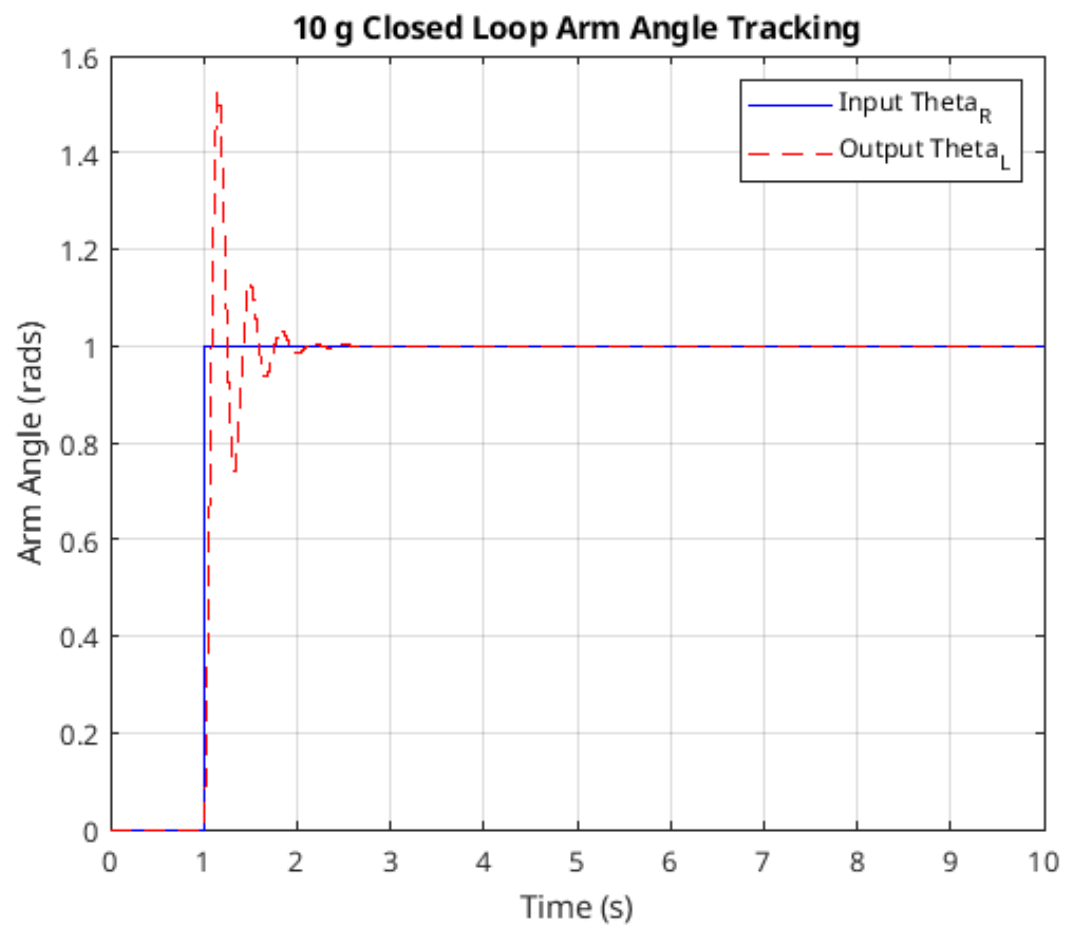
Making those gains negative pushed all of the open loop poles away from the origin, which had the unfortunate effect of creating a positive pole which will create an explosive, unstable behavior. There's this interesting overall trend where the negative gain (overall positive) closed loop poles have large real values and then as the gains go positive (negative overall) the poles converge towards -2.2 real value and then become imaginary and move back out away from the origin again increasing in real and imaginary axes.

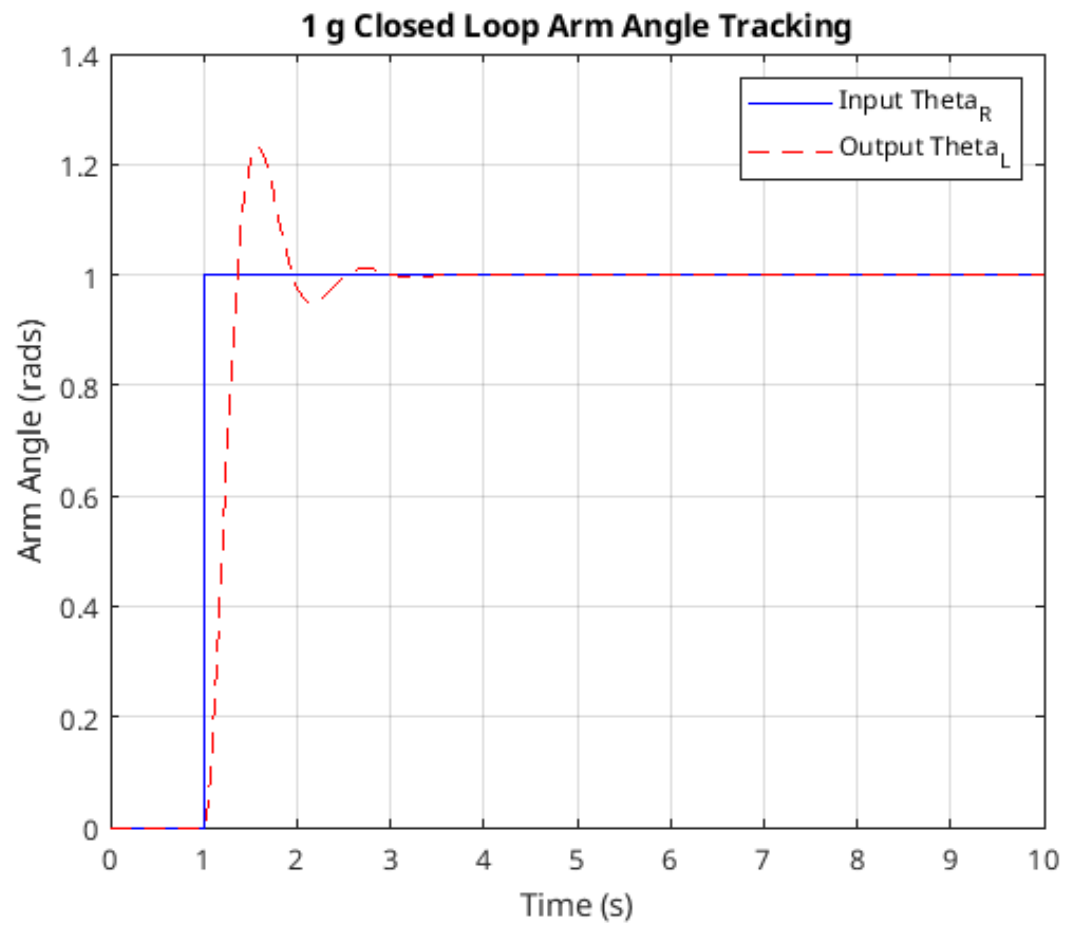
6.

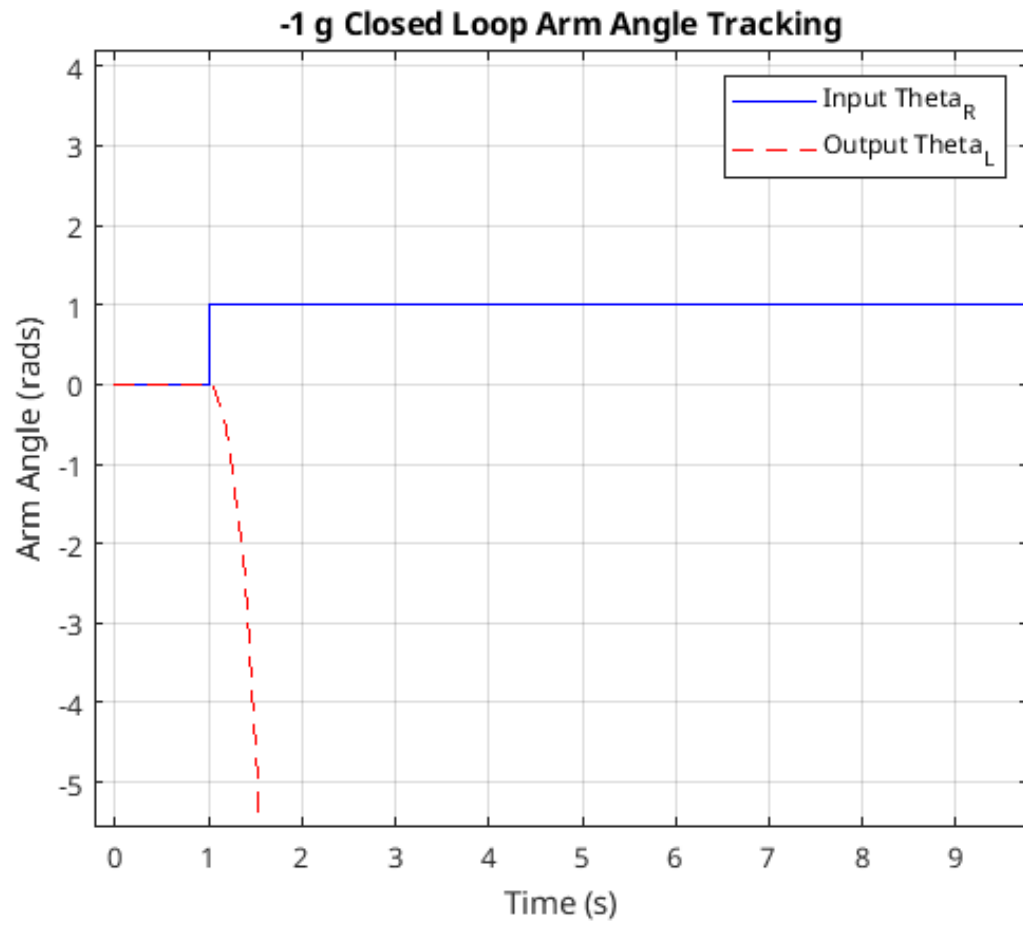
"What does the above root locus analysis tell you about the effect of the gain g on the closed loop system's natural response? Verify your conclusions by simulating the closed loop step response for several values for g ."

I basically said it above already but I think the larger positive gain scaler is going to make the natural response have a higher frequency and converge faster and the negative g is going to make the natural response explode. I used simulink to simulate a step response with 100, 10, 1, and -1 g gain scaler, the results are below.









I believe this shows my predictions to be correct. You can see the faster oscillations and quicker settling time as g is higher, when g is 100 the response settles in a third of a second and when g is 1 it takes about 3 seconds. I do think it is worth pointing out that the amplitude is also a bit higher when g is higher which I did not mention or predict before.