EE208 LAB 2

Controller design on MATLAB platform using analog root loci.

Group 9
Aryan Bansal 2020EEB1162
Hiya Kwatra 2020EEB1173
Ishaan Sethi 2020EEB1174

OBJECTIVE

1. The project requires design of a cascade feedback controller for a given analog transfer function, according to desired specifications.
2. A sensitivity analysis for variation of key parameters is further required.

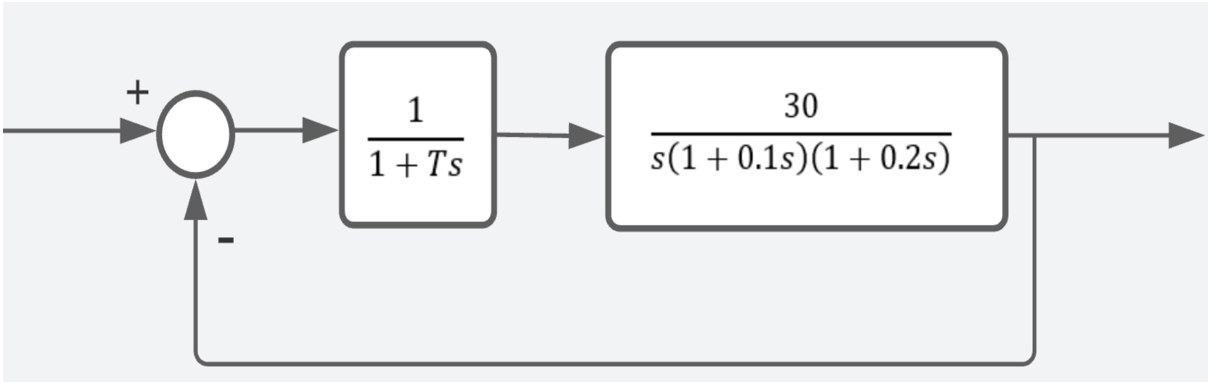
PROJECT STATEMENT

The following analog OLTF is to be operated in closed loop with a choice of **T** such that the complex poles have a damping ratio of 0.2 to 0.25.

$$G_{OL}(s) = \frac{30}{s(1+0.1s)(1+0.2s)} \times \frac{1}{1+\mathrm{Ts}}$$

Once the system is automated however, it is found that each of the parameters in the *denominator polynomial* are prone to variation up to $\pm 20\%$ of the original values. The value of T is to be selected so as to realize the required damping ratio regardless of parameter variations.

BLOCK DIAGRAM



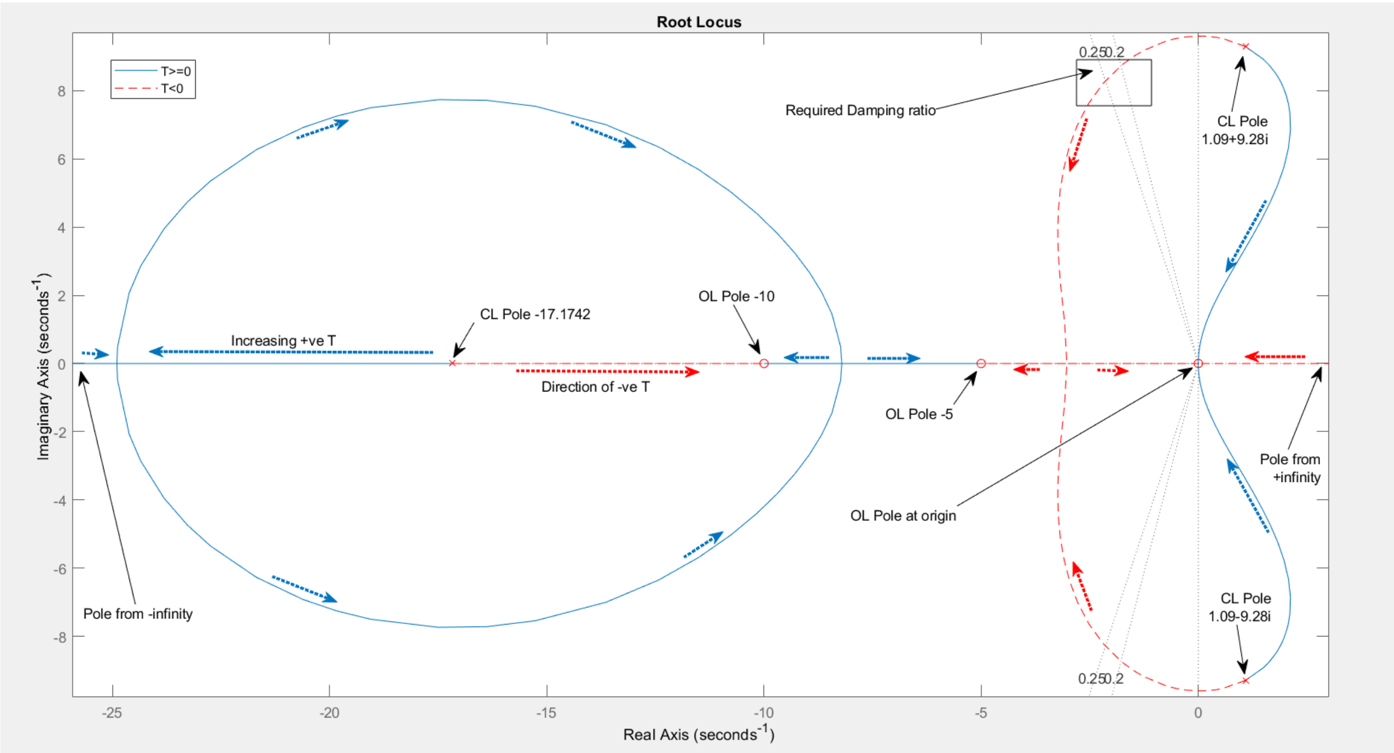
Our task is to choose T such that the damping ratio for CL Complex poles is between 0.2 and 0.25. We also need to realise this damping ratio regardless of parameter variations.

The parameters in question are 0.1 and 0.2, which may vary by up to 20%.

CHARACTERISTIC EQUATION

- The given OLTF is $\frac{30}{s(1+0.1s)(1+0.2s)(1+Ts)}$.
- The corresponding CLTF is $\frac{30}{s(1+0.1s)(1+0.2s)(1+Ts)+30}$.
- The Characteristic Equation is thus $s(1+0.1s)(1+0.2s)(1+Ts) + 30 = 0$.
- Putting it into standard form, we obtain $1 + T \times \frac{s^2(1+0.1s)(1+0.2s)}{s(1+0.1s)(1+0.2s)+30} = 0$.
- We can compare this with $1 + k\,\frac{n(s)}{d(s)}$.
- Thus, we shall use the **rlocus** command for $\frac{s^2(1+0.1s)(1+0.2s)}{s(1+0.1s)(1+0.2s)+30} = \frac{0.02s^4+0.3s^3+s^2}{0.02s^3+0.3s^2+s+30}$.
- The root loci depend only on the CL denominator polynomial (i.e., the Characteristic Equation), hence this method will indeed give us the root loci for the CLTF.

STUDYING THE ROOT LOCI



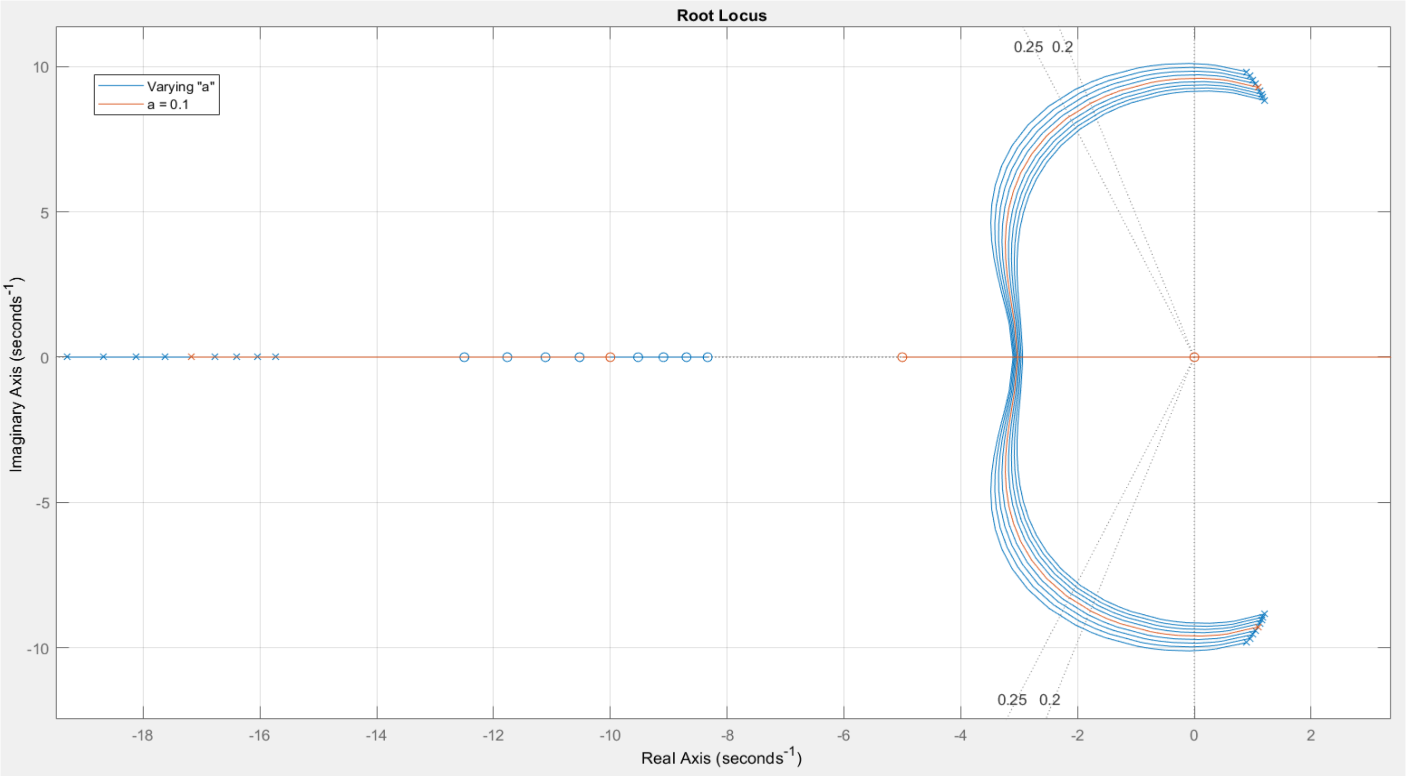
- We have plotted the root loci for all values of T, positive (blue) and negative (red).
- A positive value of T **cannot** achieve the desired damping ratio, the root loci never cross the dotted region.
- Negative values of T can achieve the desired damping ratios, but this adds an unavoidable *real pole in the RHP*, making the system **unstable**.
- Using **rlocfind** we can find the range of T that satisfies the design condition.
- The range is **-0.1067 to -0.1416**.
- The range of unstable poles is from +10.3446 to +12.3251.

PARAMETER VARIATION

Let us replace 0.1 and 0.2 in the OLTF by "a" and "b" so that the OLTF becomes $\frac{30}{s(1+as)(1+bs)(1+Ts)}$, where a = 0.1 and b = 0.2.

Now we vary **a** from 0.08 to 0.12 (20% variation in 0.1).

The effect on (negative) root loci is –

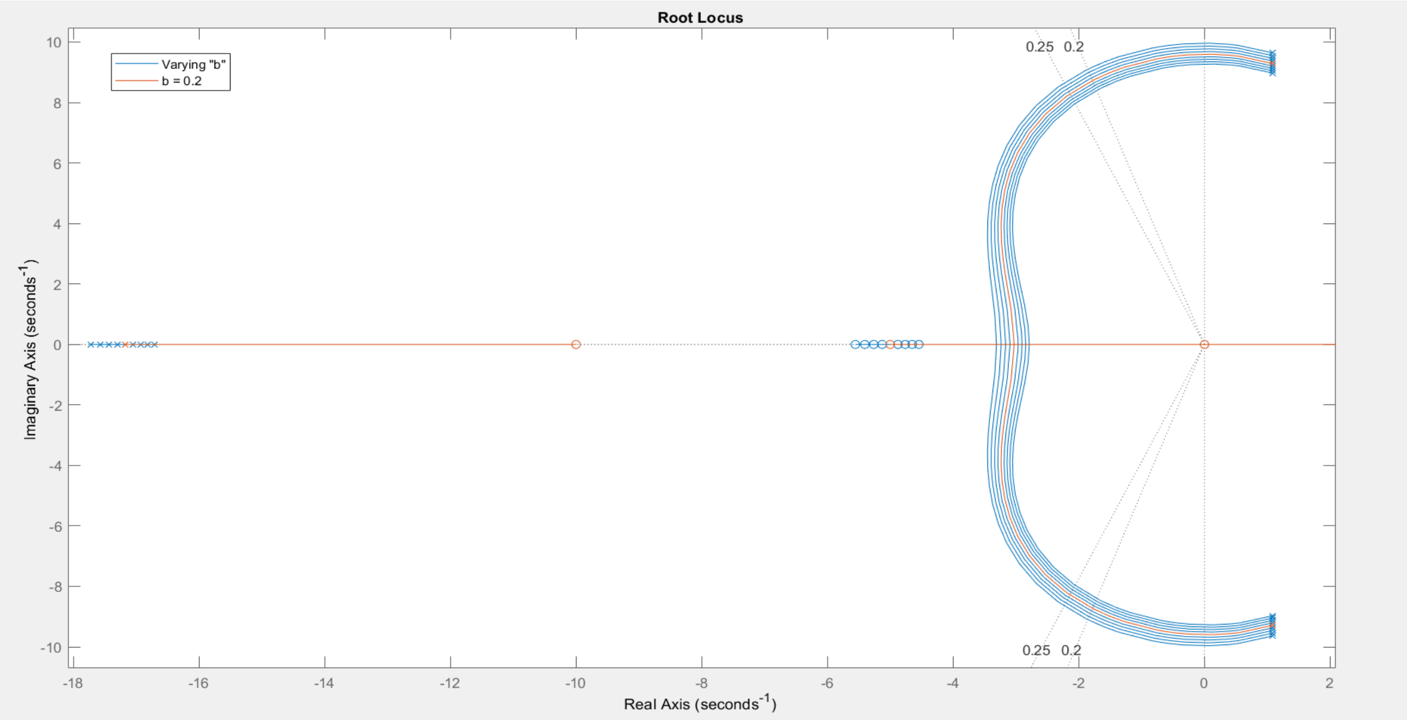


We'll now tabulate the ranges of T-values for a=0.08, a=0.1, and a=0.12.

a	T for ζ=0.2	T for ζ=0.25
0.080	-0.0862	-0.1140
0.100	-0.1060	-0.1410
0.120	-0.1260	-0.1680

- There is no interval that satisfies all the three ranges – a=0.12 requires a minimum T of -0.126, but that will violate the condition for a=0.08!
- However, we can choose an interval for variations of up to 10% – -0.116 to -0.128, by inspecting the DataTips in the plot.

Now for **b**, we vary from 0.16 to 0.24 (20% variation in 0.2).



We'll now tabulate the ranges of T-values for b=0.16, b=0.2, and b=0.24.

b	T for ζ=0.2	T for ζ=0.25
0.160	-0.0935	-0.1240
0.200	-0.1060	-0.1410
0.240	-0.1180	-0.1560

- To fully satisfy variations in **b** we can choose T between -0.118 and -0.124.
- Thus, the overall range is from -0.118 to -0.124.
- This range satisfies 10% variation in parameter a, and 20% variation in parameter b.
- We were unable to find a suitable T to satisfy 20% variation in parameter a.

ATTEMPT AT STABILIZATION

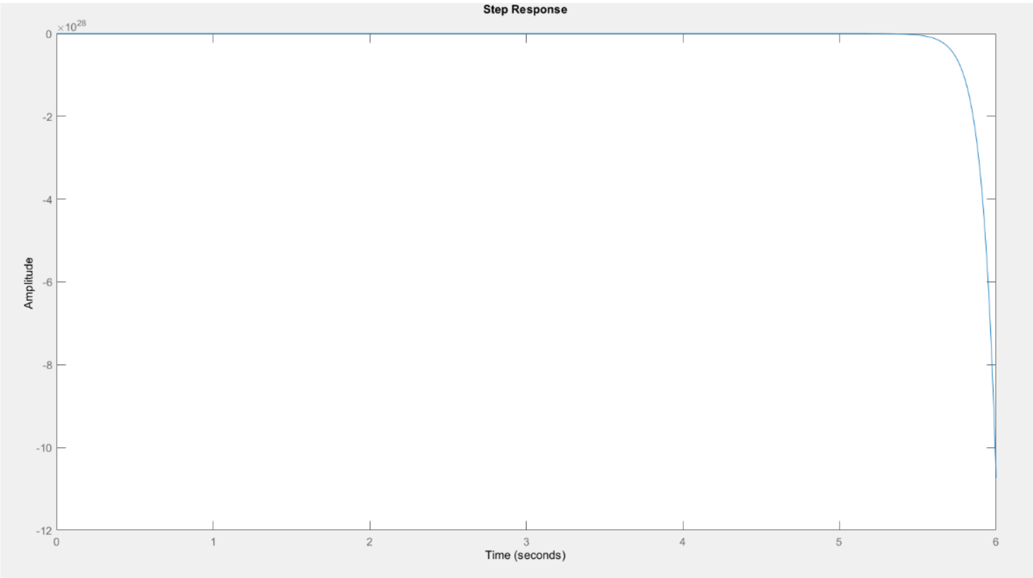
Now we are left with the issue of an unstable pole, due to us using a negative value of T (which placed our OL pole in the RHP).

Let us pick **T = -0.12**. Our CLTF is thus $\frac{-30}{0.0024s^4+0.016s^3-0.18s^2-s-30}$.

We have poles at:

- -14.2238 → stable
- 11.4358 → unstable
- -1.9394 + 8.5490i → stable
- -1.9394 - 8.5490i → stable

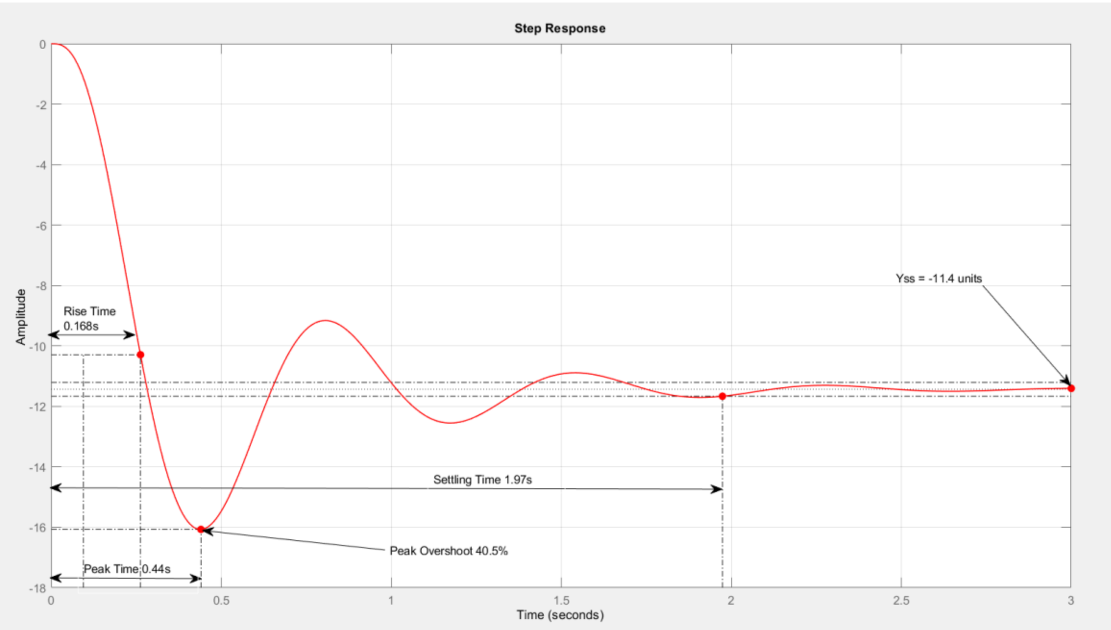
We’ll first observe the step response of this CL system. The plot is barely visible because the response reaches magnitudes of the order of 10^{25} in a few seconds.



The system is **unstable** and has a negative step response because we used a negative value of T.

We shall try to cancel the unstable pole by placing a zero at that location. We multiply our TF by (s-11.4358258). The precise position of the unstable pole can be found using the pzmap function.

The resulting step response is:



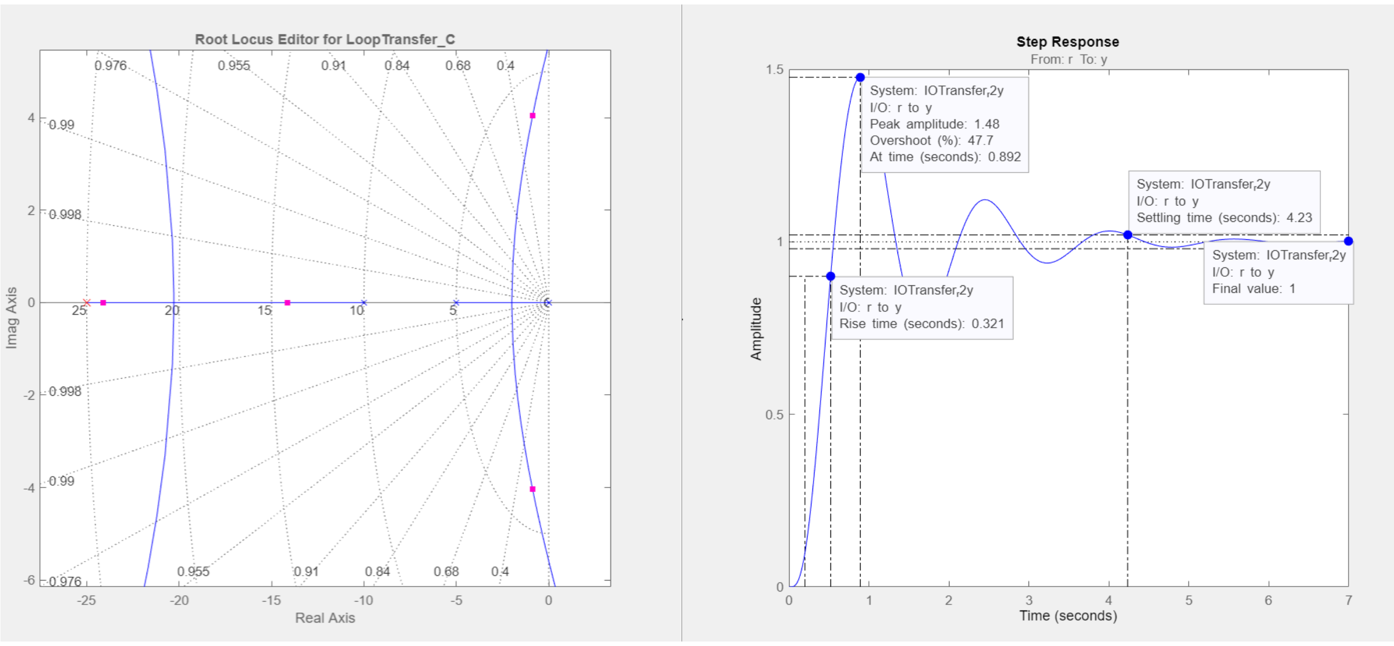
The steady-state value is negative because of using a negative value of T. This response is a suitable one, however this method is not practical due to the high precision needed.

While pole-zero cancellation is theoretically possible, even in a simulation we required a very precise value. Using a -ve value of T is not a feasible solution, and we must find another method of achieving our goal.

ADDING AN ADDITIONAL CONTROLLER

Since we cannot use a negative value of T, we *must* add another controller in cascade, and try to achieve the required damping ratio using that.

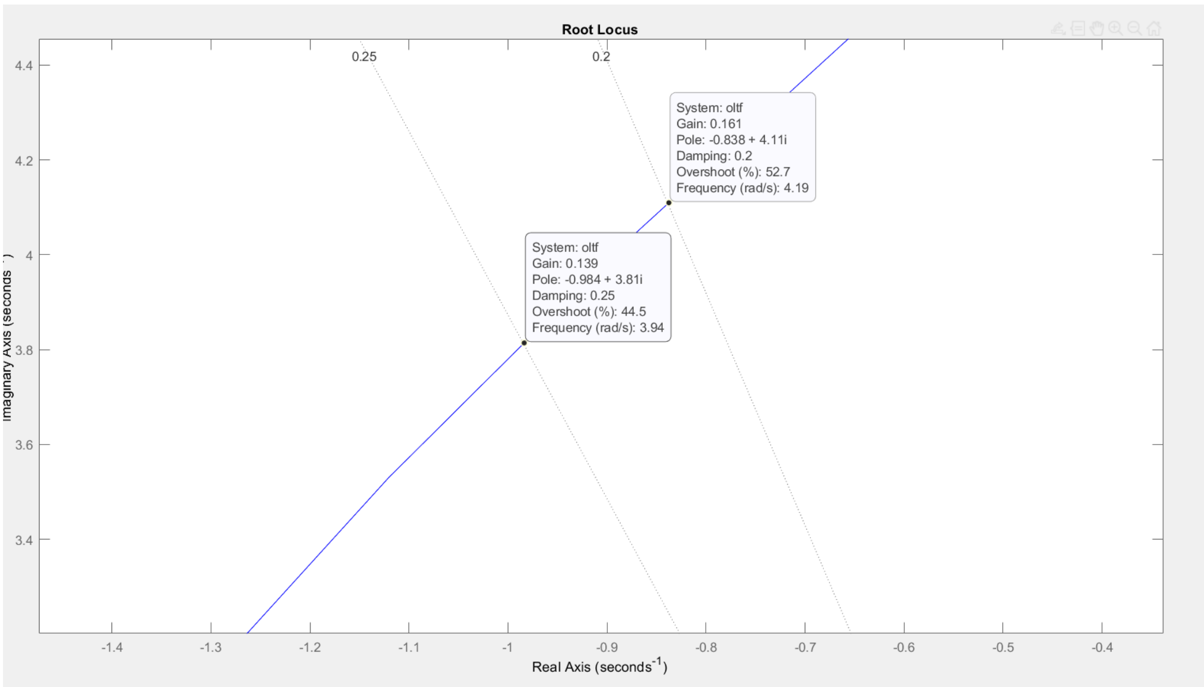
If we use a **proportional controller** cascaded with the given OLTF, we will manage to achieve our desired performance. Let us see the root loci and step response of this system –



We have chosen **T = 0.04**, and **Kp = 0.155**. The damping ratio for the complex poles is **0.212**.

The benefit of choosing a small **T** value is that we will have only one pair of complex poles at the chosen Kp. Otherwise, we would be unable to satisfy the design requirement – “complex poles should have a damping ratio between 0.2 and 0.25”.

With the root locus plot for **T=0.04**, we see that the feasible **range of Kp is 0.138 to 0.161**.

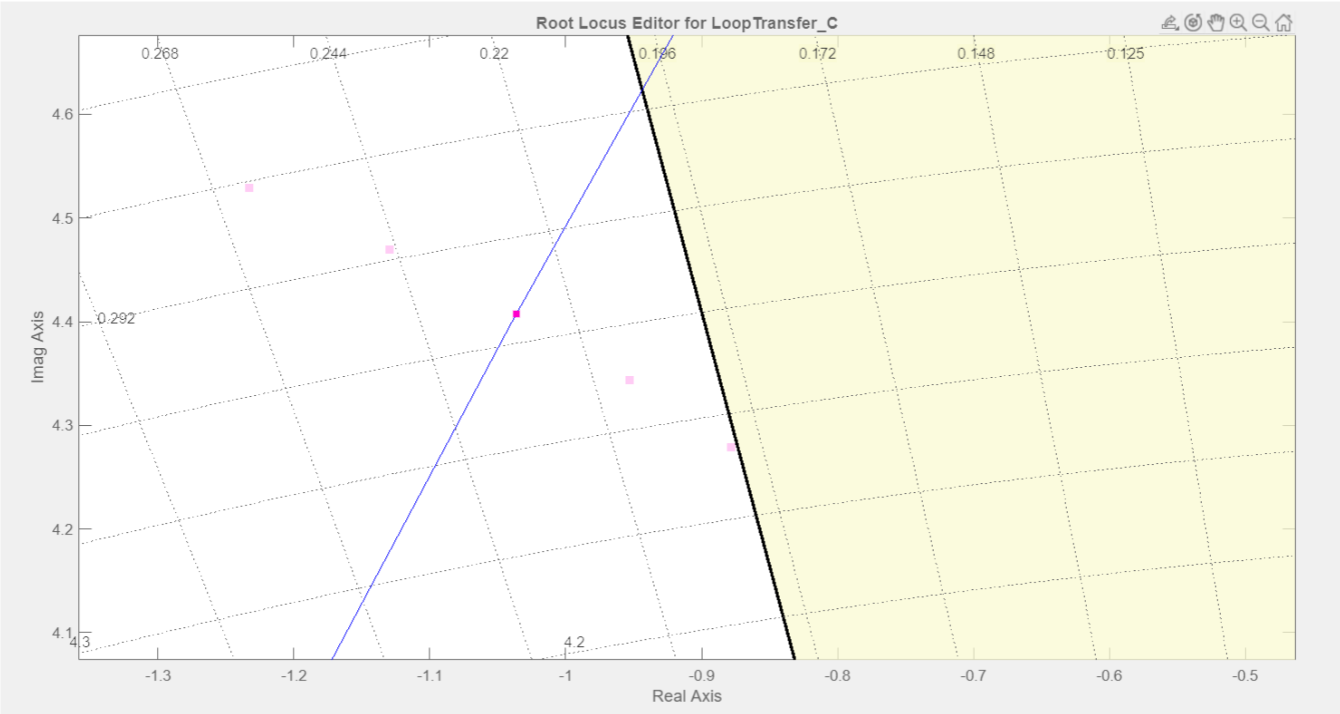


PARAMETER VARIATION

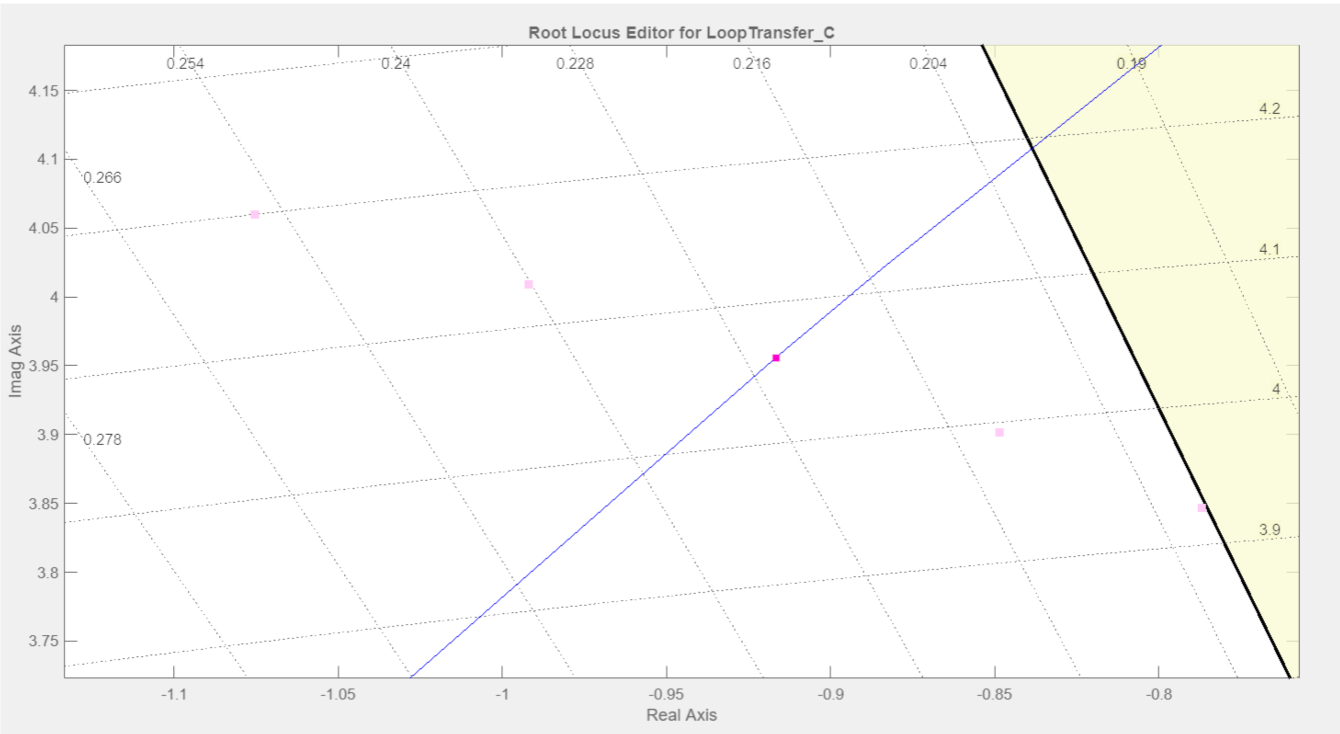
We shall vary **a** from 0.08 to 0.12, and **b** from 0.16 to 0.24, separately. We will observe the root loci for each and determine a suitable range of **T** and **Kp**.

VARYING "A"

This is the plot for T=0.01. The pink squares show the pole locations for varying a. The black line is the zeta=0.2 curve. We can see that for **a=0.08**, the damping condition is violated. If we use a lesser gain value, then the condition for **a=0.12** is violated.



We can also view the loci for T=0.04:

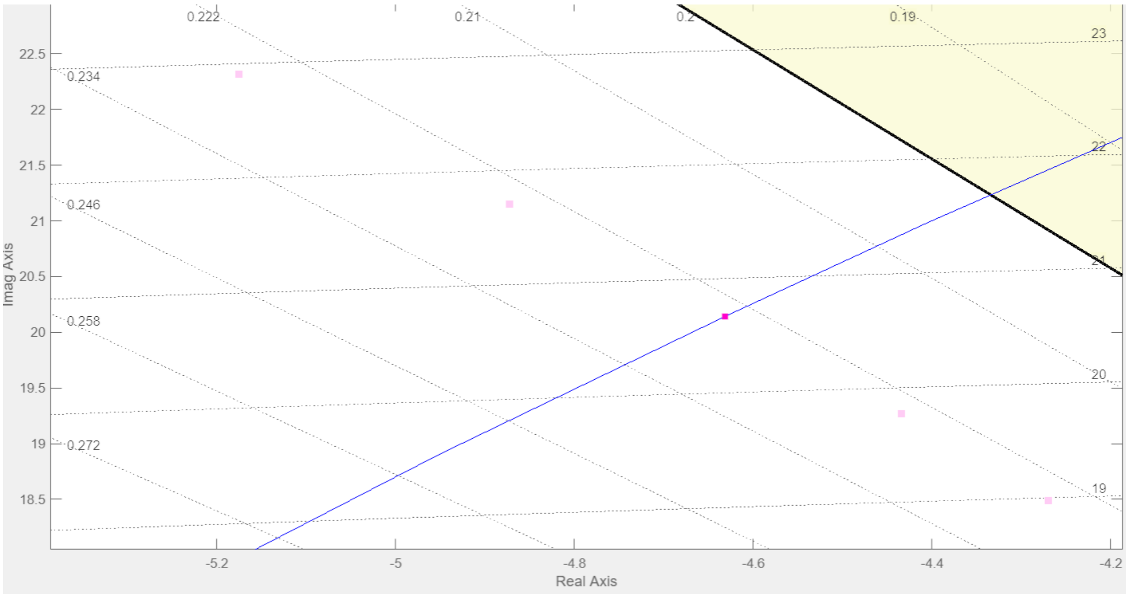


Again, we are unable to find a gain that can simultaneously satisfy conditions for **a=0.08** and **a=0.12**.

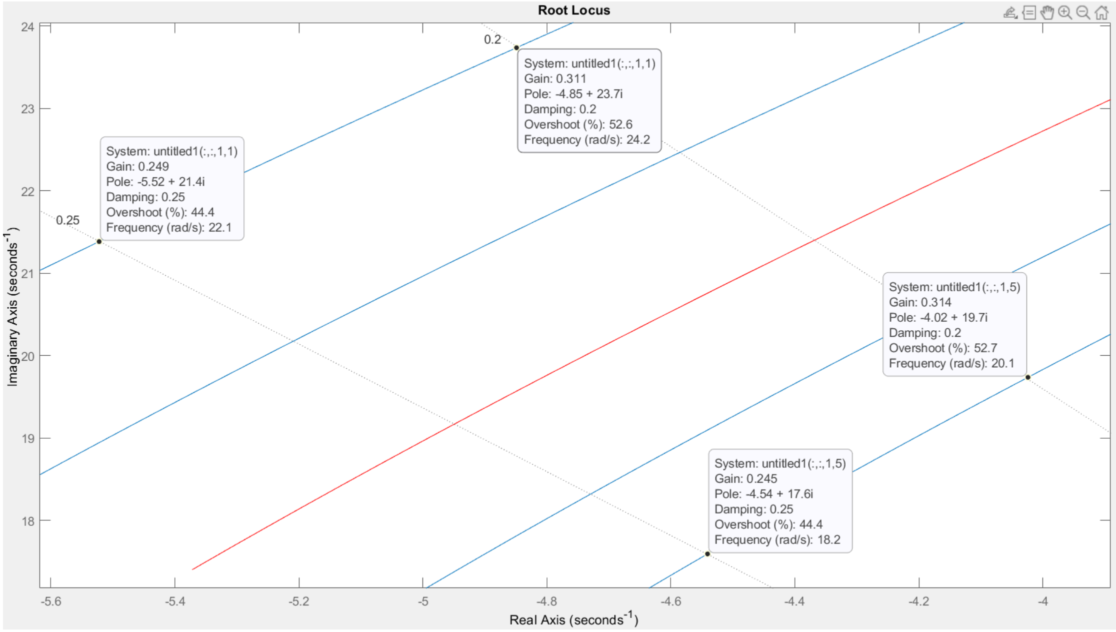
Increasing T beyond 0.04 will cause the system to have two pairs of complex poles. This is unwanted. Decreasing T from 0.01 is not useful either, since the effect of the added pole becomes insignificant as it moves away from the origin.

Proportional gain cannot give us the desired system. We will therefore use a differentiator instead of constant gain.

We add a differentiator to our system, with **T=0.013** (chosen to accommodate the variation in **b** as well, through trial-and-error).



We are now able to find a suitable range of gains that will satisfy the condition for 20% variation in **a**.



Value of a	K for ζ=0.2	K for ζ=0.225	K for ζ=0.25
0.08	0.312	0.278	0.249
0.09	0.311	0.277	0.246
0.1	0.311	0.275	0.245
0.11	0.311	0.275	0.245
0.12	0.313	0.277	0.245

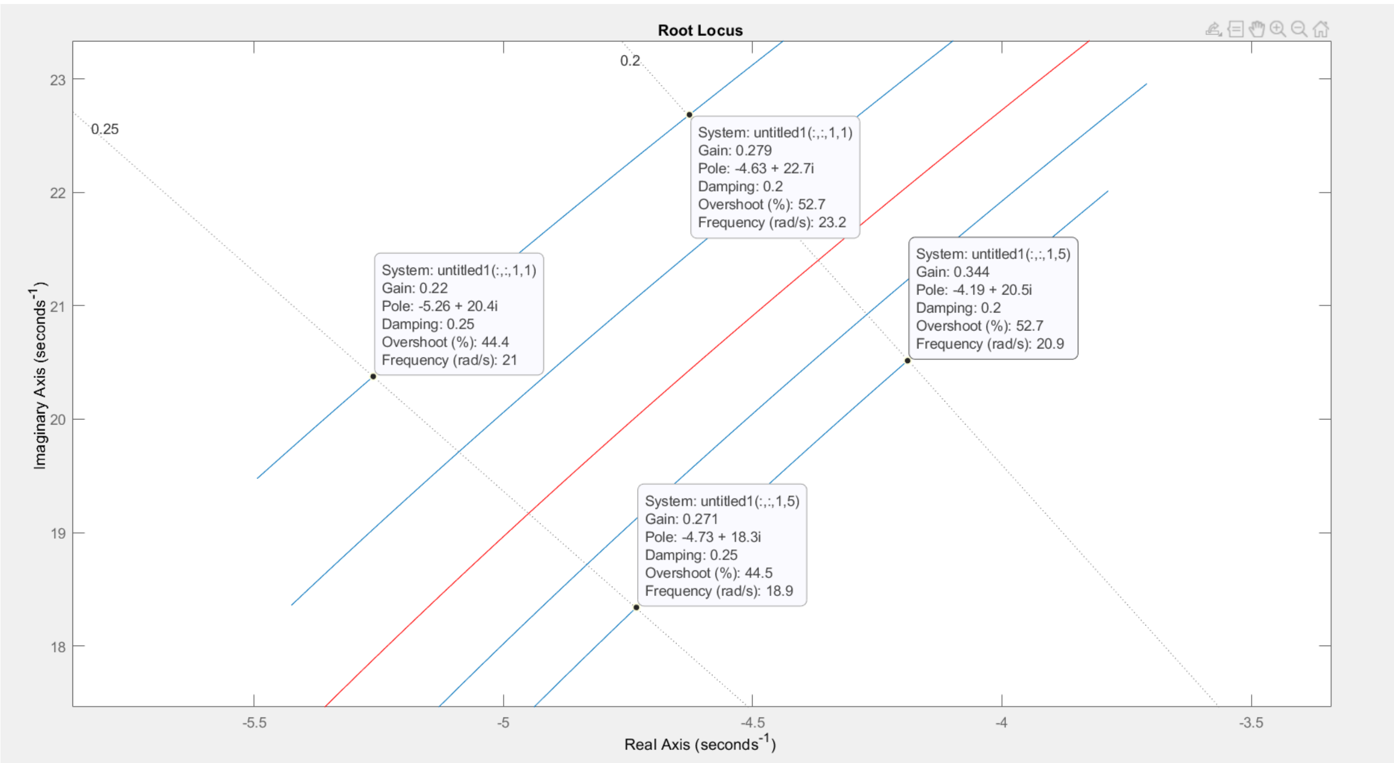
Due to the addition of the differentiator, there is little variation in the gains corresponding to different values of a.

The required range of gains is thus, **0.249 to 0.311**.

VARYING “B”

Having found a suitable range for **a**, we shall now find the range for **b**.

Continuing with the same differentiator and T=0.013 system, we have –



Value of b	K for ζ=0.2	K for ζ=0.225	K for ζ=0.25
0.16	0.279	0.247	0.220
0.18	0.294	0.261	0.233
0.2	0.311	0.276	0.245
0.22	0.328	0.291	0.258
0.24	0.344	0.305	0.271

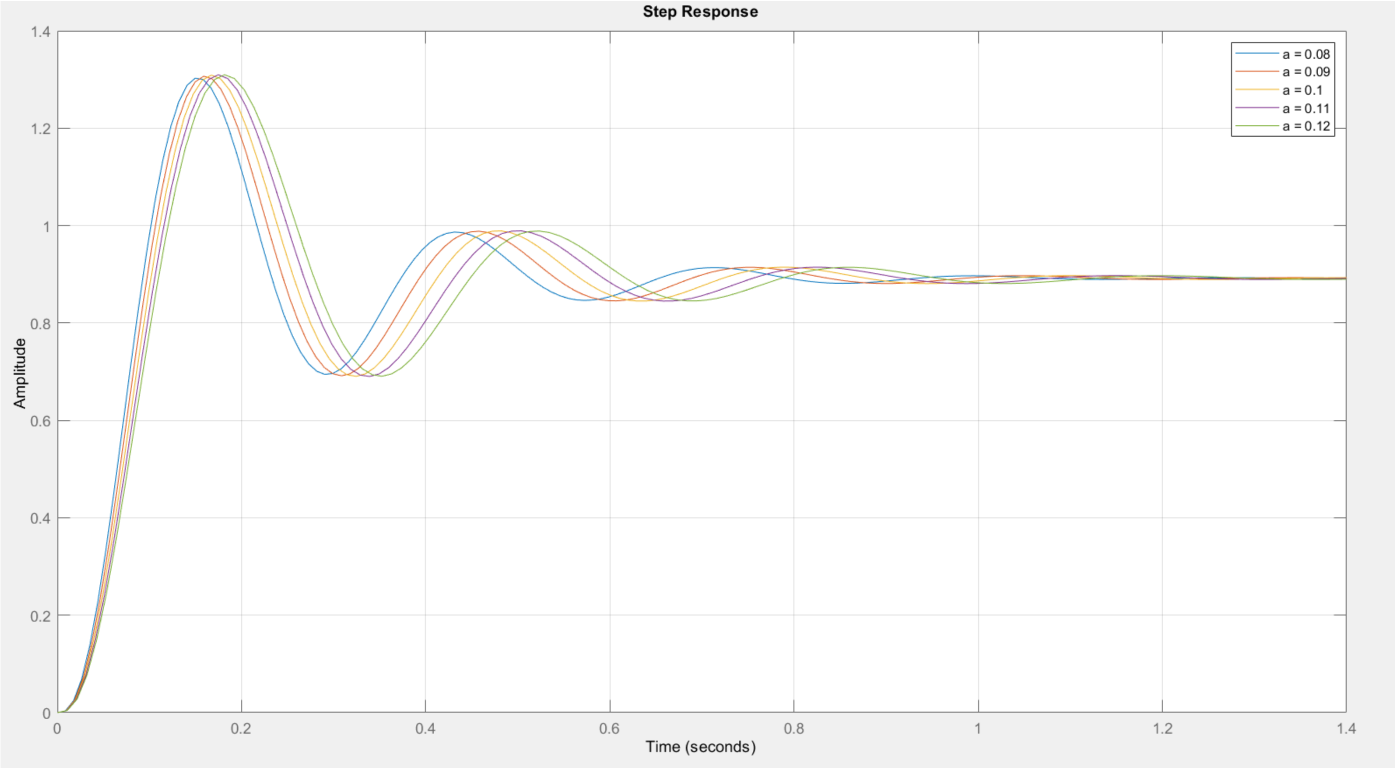
We can ascertain the suitable range to be **0.271 to 0.279**.

OVERALL RANGE

Thus, the overall range is – (0.249 to 0.311) ∩ 0.271 to 0.279 **= 0.271 to 0.279**, with **T=0.013**.

Let us plot the step responses for **K=0.275**.

First with variation in **a**:

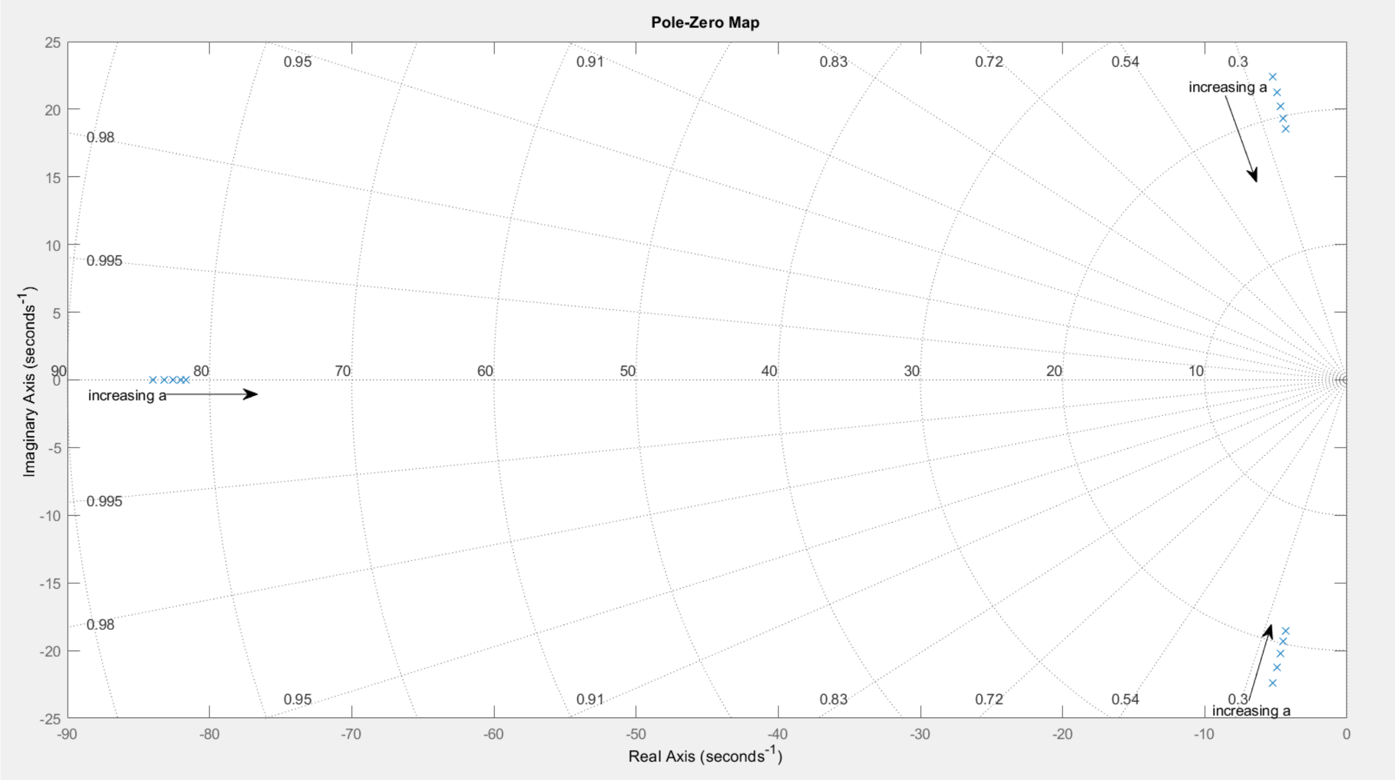


- The dynamics are oscillatory-type.
- A table of performance measures is given below:

Value of a	Rise Time	Settling Time	Peak Time	Peak Overshoot
0.08	0.0571	0.7412	0.1496	46%
0.09	0.0599	0.7837	0.1589	46.516%
0.1	0.0627	0.8231	0.1672	46.75%
0.11	0.0653	0.8601	0.1747	46.8%
0.12	0.0679	0.8951	0.1815	46.84%

All the parameters tend to increase with increasing **a**.

We can see the pzmap to understand this.



Here, the poles move towards the origin, but the complex conjugate poles move roughly along the line of constant damping ratio, so the damping ratio is not decreased by a large amount. Instead, since the poles are moving towards the origin, their natural frequency reduces and hence the increase in performance measures.

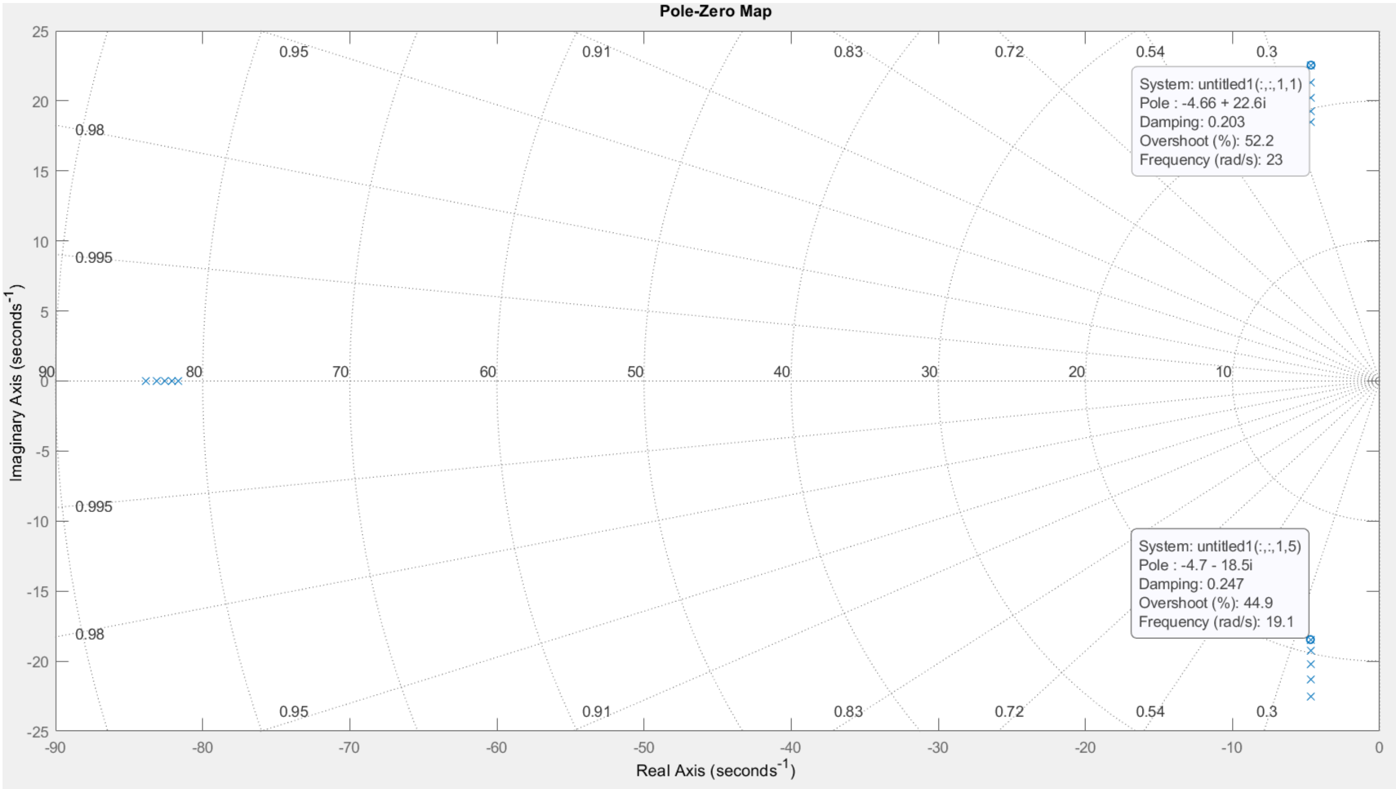
Now for the variations in **b**:

- Again, we have oscillatory-type dynamics.
- A table of performance measures is shown below:

Value of b	Rise Time	Settling Time	Peak Time	Peak Overshoot
0.16	0.0556	0.7558	0.1481	50%
0.18	0.0591	0.7922	0.1577	48.38%
0.2	0.0627	0.8231	0.1672	46.75%
0.22	0.066	0.8447	0.1767	45.15%
0.24	0.0693	0.7536	0.186	43.5%

In this case, Peak Overshoot is decreased while other parameters are increased with increasing **b**.

Looking at the pzmap -



From the DataTips that are highlighted, we can see that the poles move vertically as we increase **b**, thereby *increasing* the **damping**, causing an increase in our performance measures, but decreasing the peak % overshoot.

SENSITIVITY OF PARAMETERS

Our overall CLTF is now $\frac{30^*(\mathrm{Ks})}{s(1+\mathrm{as})(1+\mathrm{bs})(1+Ts)+30^*(\mathrm{Ks})}$, with K from 0.271 to 0.279, and T=0.013.

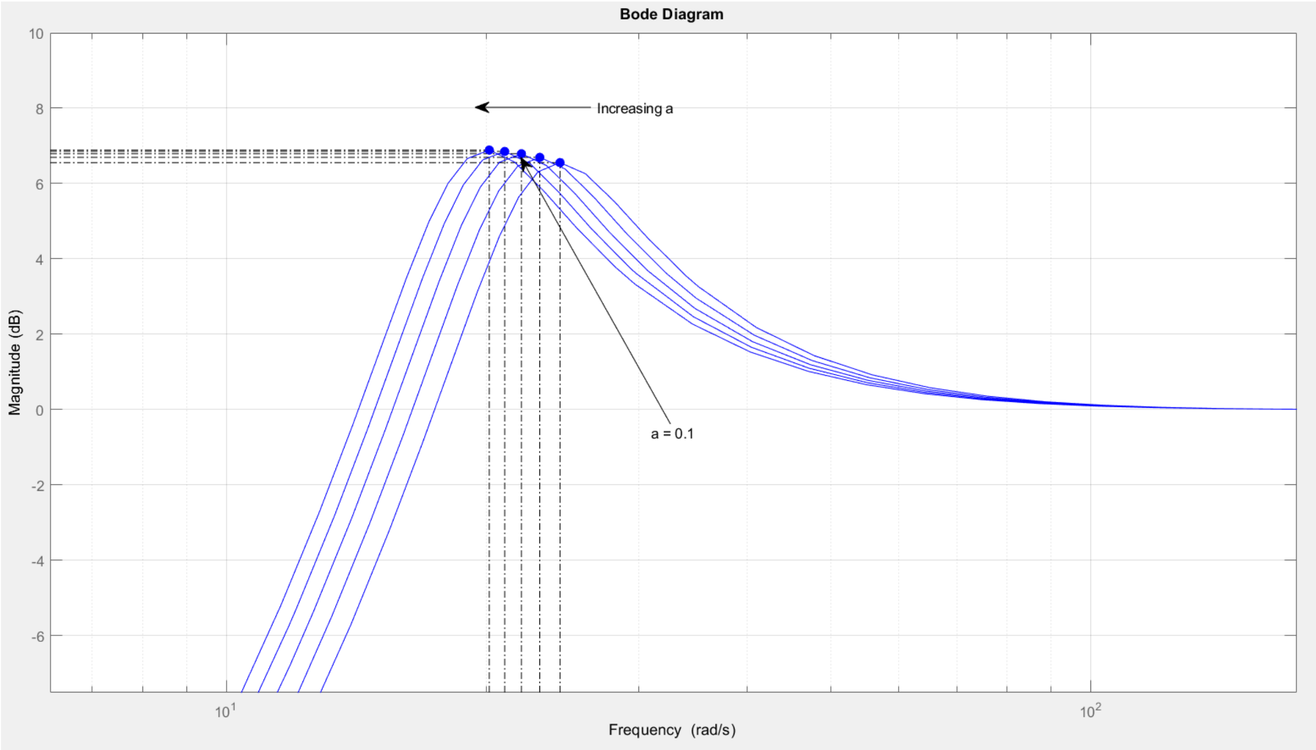
We can find the System Sensitivity Function with respect to **a** and **b** as by the following formula – $S_{\mathrm{a}}^{T} = \frac{dT}{da}\frac{a}{T}$.

We obtain, $S_{\mathrm{a}}^{T} = \frac{-as^2(1+bs)(1+Ts)}{s(1+\mathrm{as})(1+\mathrm{bs})(1+Ts)+30*(Ks)}$.

Similarly, $S_{\mathrm{b}}^{T} = \frac{-\mathrm{b}s^2(1+as)(1+Ts)}{s(1+\mathrm{as})(1+\mathrm{bs})(1+Ts)+30*(Ks)}$.

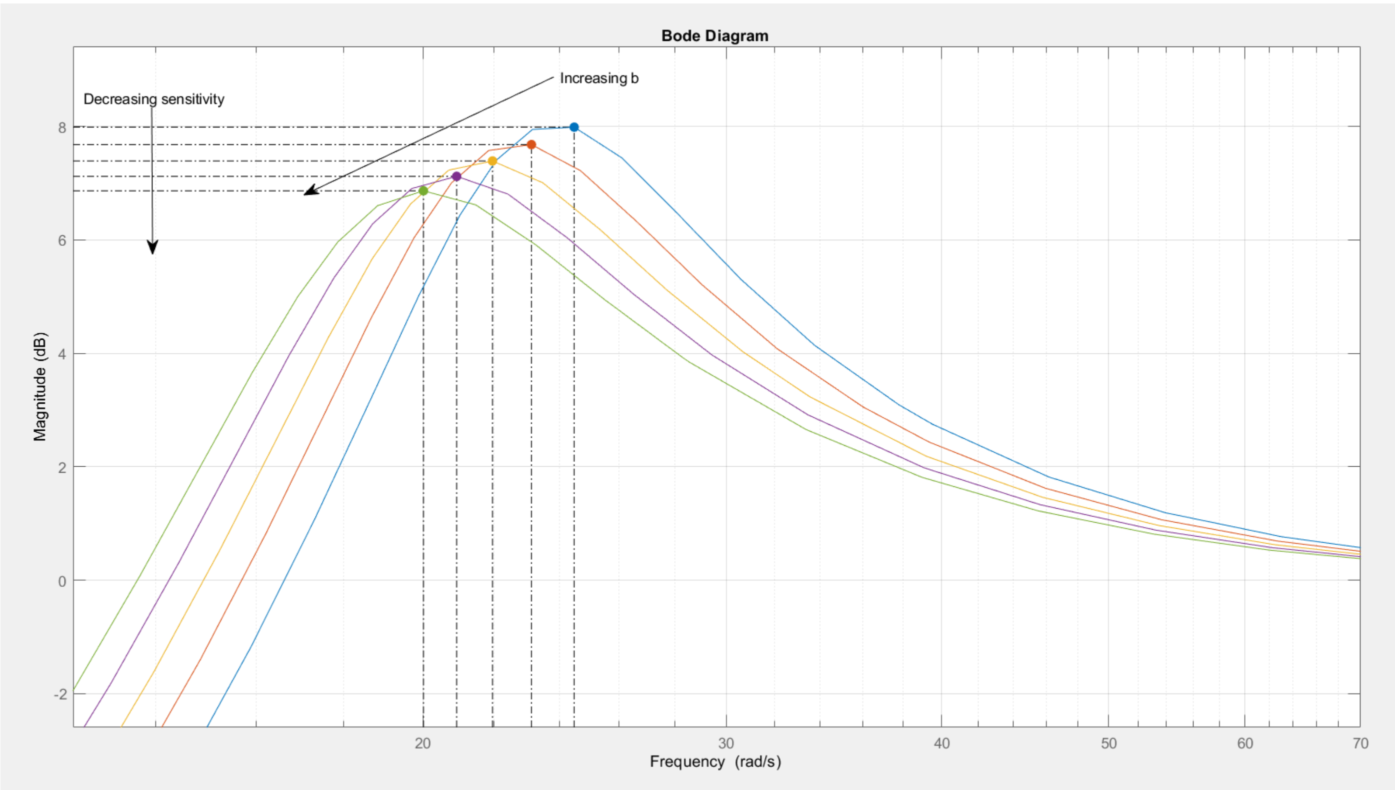
We can see that a large **K** will reduce system sensitivity, however we cannot simply increase K beyond the range shown above since we need to follow the specifications.

We can plot the Bode Magnitude diagram of the Sensitivity function (with respect to a) –



As we increase a, sensitivity with respect to a (max sensitivity in particular) also increases.

We will also plot the Bode Magnitude diagram of the Sensitivity function (with respect to b).



As we increase **b**, sensitivity with respect to b decreases.

We can tabulate the values –

a	Max Sensitivity in dB
0.08	6.56
0.09	6.7
0.1	6.79
0.11	6.85
0.12	6.89

b	Max Sensitivity in dB
0.16	7.99
0.18	7.68
0.2	7.39
0.22	7.12
0.24	6.86

CONCLUSIONS

We were asked to determine the value of "T" in a given OLTF, to be operated in closed loop, such that the complex poles had a damping ratio between 0.2 and 0.25.

- We found that for positive values of T, said specifications could not be achieved for any value of T.
- We then tried negative values of T, and found a suitable range of T that would provide the desired damping ratio even with parameter variation.
- This method was deemed unsuitable due to the presence of a real pole in the RHP, which caused the system to be unstable.
- We showed how the system could be stabilized by pole-zero cancellation, but this cannot be implemented practically.
- Upon the addition of a constant gain to the system, the system was stabilized and the required damping ratios could be achieved.
- However, this method could not satisfy the conditions imposed by variation of parameters in the denominator polynomial.
- We showed how the addition of a differentiator allowed us to choose a suitable T and loop gain, which would provide the adequate specifications in light of parameter variations.
- We also looked at the step responses of the final CLTF wrt change in parameters.
- Finally, we looked at the Sensitivity function of the CLTF with respect to variations in the parameters, and plotted the Bode Magnitude diagrams of the same.

Based on various analyses, we have arrived at the following controller parameters –

- T = 0.013, corresponding to a pole at -75.
- A cascaded controller with Transfer Function $C(s) = Ks$, where K is between 0.271 to 0.279.

REFERENCES

- *https://eng.libretexts.org/Bookshelves/Industrial_and_Systems_Engineering/Book%3A_Introduction_to_Control_Systems_(Iqbal)/04%3A_Control_System_Design_Objectives/4.05%3A_Sensitivity_and_Robustness*

- *http://www.dii.unimo.it/~zanasi/didattica/Automatic_Controls/Luc_CA_2018_Contour_locus.pdf*
- *Automatic Control Systems, 9th Ed. – 7.4, 7.5*
- *Modern Control Engineering, 5th Ed. – 6.3, 6.5*
- *Relevant MATLAB Documentation for Functions and Control Systems Designer*

MATLAB CODE

Setup code:

```
a0 = 0.1;
b0 = 0.2;
s = tf('s');
G_0 = 30/(s*(1+a0*s)*(1+b0*s)); % OLTF when T=0
```

Root loci:

```
zeta = [0.2 0.25];
w = [];
num = [a0*b0,a0+b0,1,0,0];
den = [a0*b0, a0+b0, 1, 30];
GG = tf(num,den);
rlocus(GG,-GG,'r'), sgrid(zeta,w)
```

Arrays for parameter variation:

```
a = [0.08 0.09 0.1 0.11 0.12];
b = [0.16 0.18 0.2 0.22 0.24];
G_a = tf(zeros(1,1,1,5));
for i=1:5
    G_a(:,:,:,i) = 30/(s*(1+a(i)*s)*(1+b0*s));
end
G_b = tf(zeros(1,1,1,5));
for i=1:5
    G_b(:,:,:,i) = 30/(s*(1+a0*s)*(1+b(i)*s));
end
```

For Sensitivity Bode plots:

```
T = 0.013;
K = 0.275;
for i = 1:length(a)
    sensitivity = -
a(i)*s^2*(1+b0*s)*(1+T*s)/(s*(1+a(i)*s)*(1+b0*s)*(1+T*s)+30*K*s);
    bode(sensitivity);
    hold on
end
```

FUNCTIONS USED

rlocus, tf, residue, pzmap, step, stepinfo, controlSystemDesigner, rlocfind, bode