GROUP 18

Aman Saini (2020EEB1155)

Ansaf Ahmad (2020EEB1160)

Anirudh Sharma (2020EEB1158)

CONTROLLER DESIGN ON MATLAB PLATFORM USING DISCRETE ROOT LOCI.

OBJECTIVE

- To design a cascade transfer function for a given digital transfer function according to desired specifications
- To perform sensitivity analysis for variation of key parameters.

DISCRETE OPEN LOOP TRANSFER FUNCTION TO BE OPERATED IN CLOSED LOOP

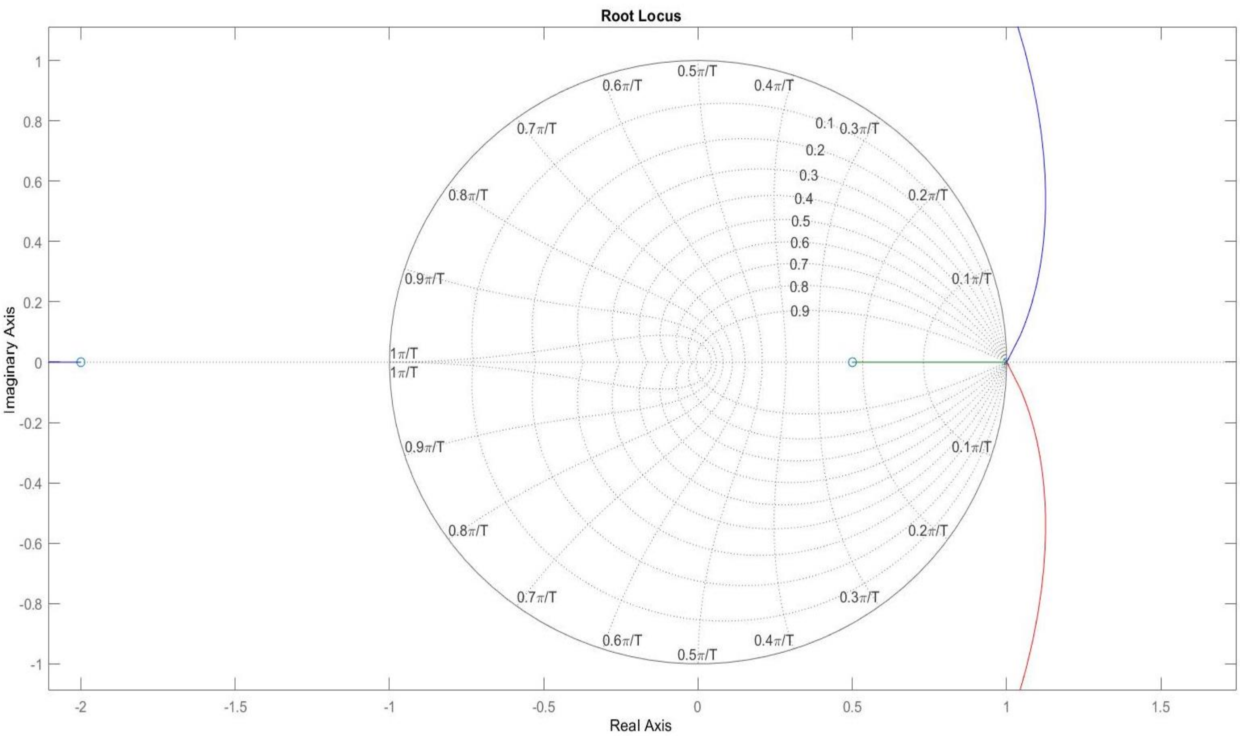
The OLTF of a digital system under study has three marginally stable poles, that is, certain frequencies at which sustained oscillations are expected.

$$\mathbf{G(z) = \frac{z^2+1.5z-1}{(z-1)^3}}$$

The given OLTF has all its poles at z = 1, (which is a point of marginal stability). However, because of presence of three poles, the given system is unstable.

Closing the system with feedback gain K, does not make the system stable as there is always at least one pole outside the unit circle.

To confirm the above statement, we plot root loci and step response of the given system G(z) with negative unity feedback.

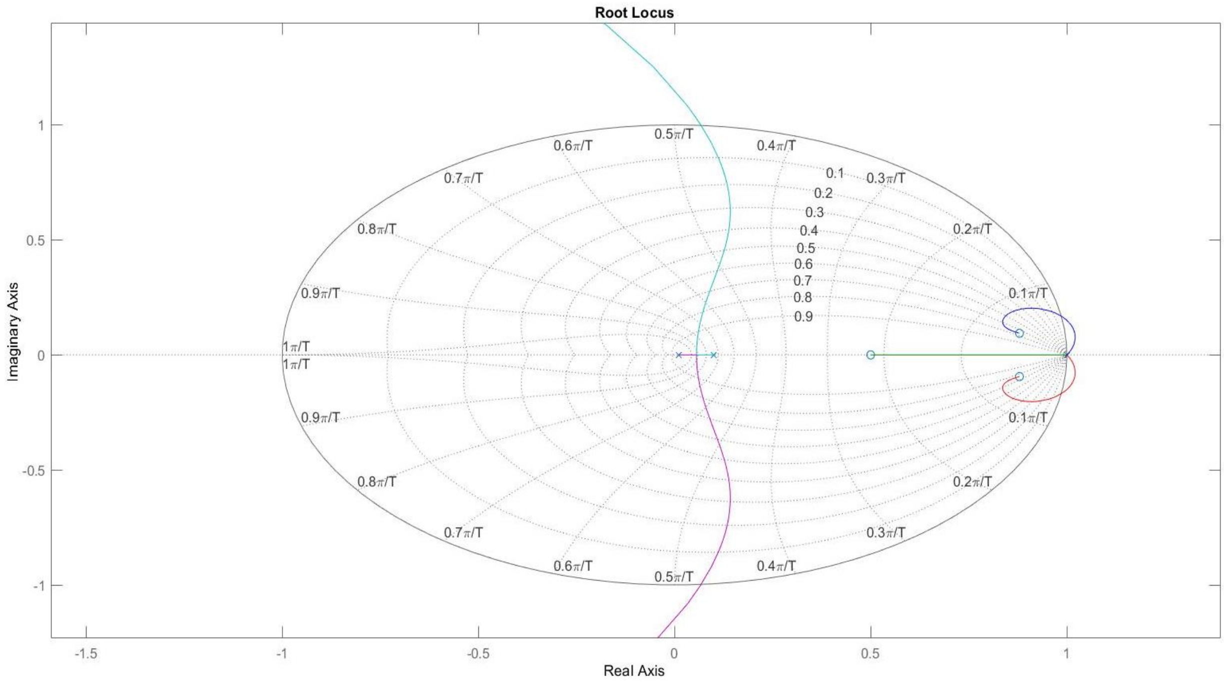

Root Locus
Imaginary Axis
Real Axis

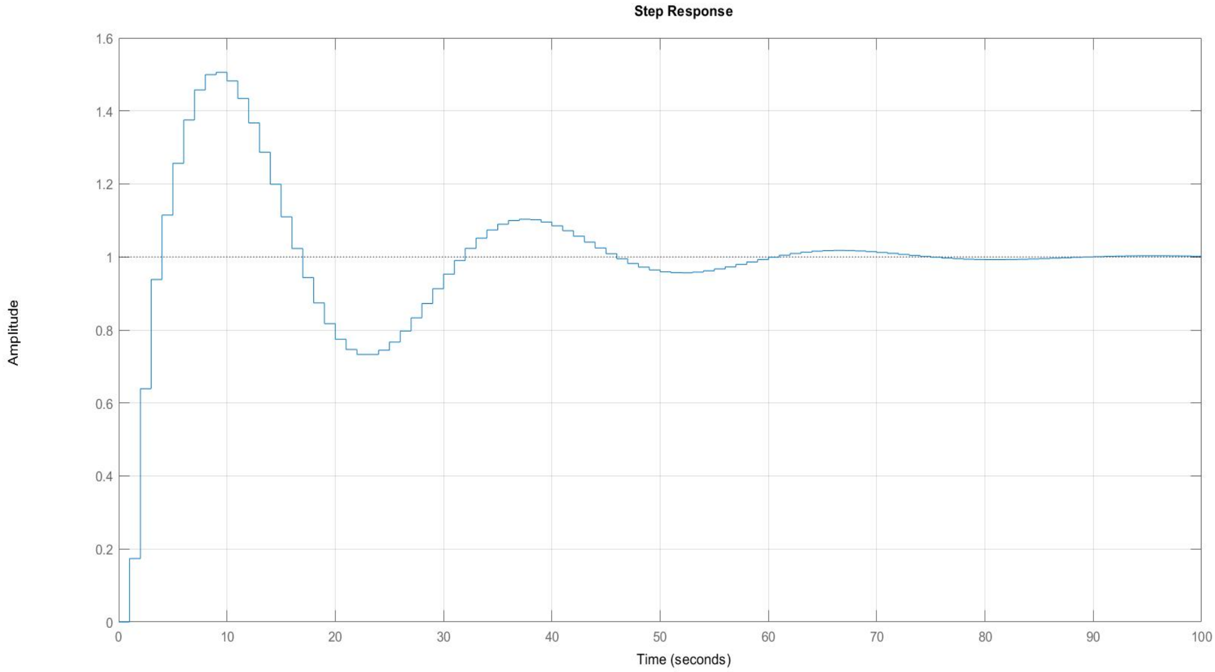

Step Response
$\times 10^{28}$
Amplitude
Time (seconds)

As evident from above root locus plot, we need to design a controller to make overall CLTF stable.

We obtain the following root loci and step response after cascading our controller C(z)

$$\boldsymbol{C(z) = 0.17371 * \frac{z^2 - 1.758z + 0.7815}{(z - 0.1)(z - 0.01)}}$$

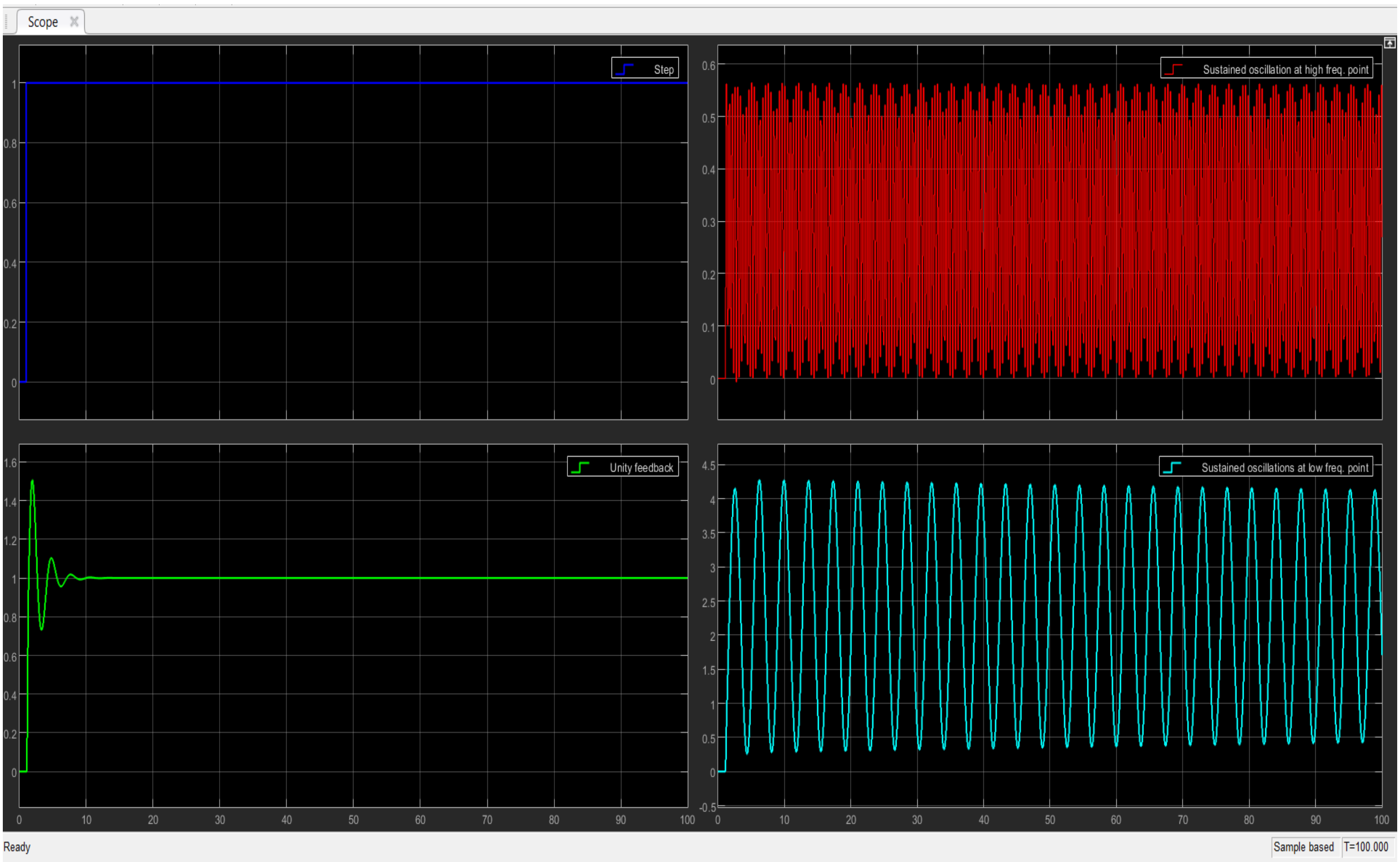




The present graph uses sampling time = 1 sec (as a default value), however changing the settling time only changes the scale of time axis and have no other impact on step response

From the above two plots we observe the overall system becomes stable for a considerable range of **feedback Gain K**.

We notice that the root locus (when we add the controller) crosses the unity circle at 4 different positions, which corresponds to 2 different frequencies (one comparatively lower that the other) of sustained oscillations, hence we need to conduct the study separately for both the cases.



LEGEND

Step Input

Step Response (unity feedback)

Sustained Oscillation at high frequency

Sustained Oscillation at low frequency

Thus, we studied the various aspects of the system, and looked at the frequencies at which sustained oscillations are obtained, keeping the zeroes constant, the different responses to step input are obtained only by varying feedback gain 'K'.

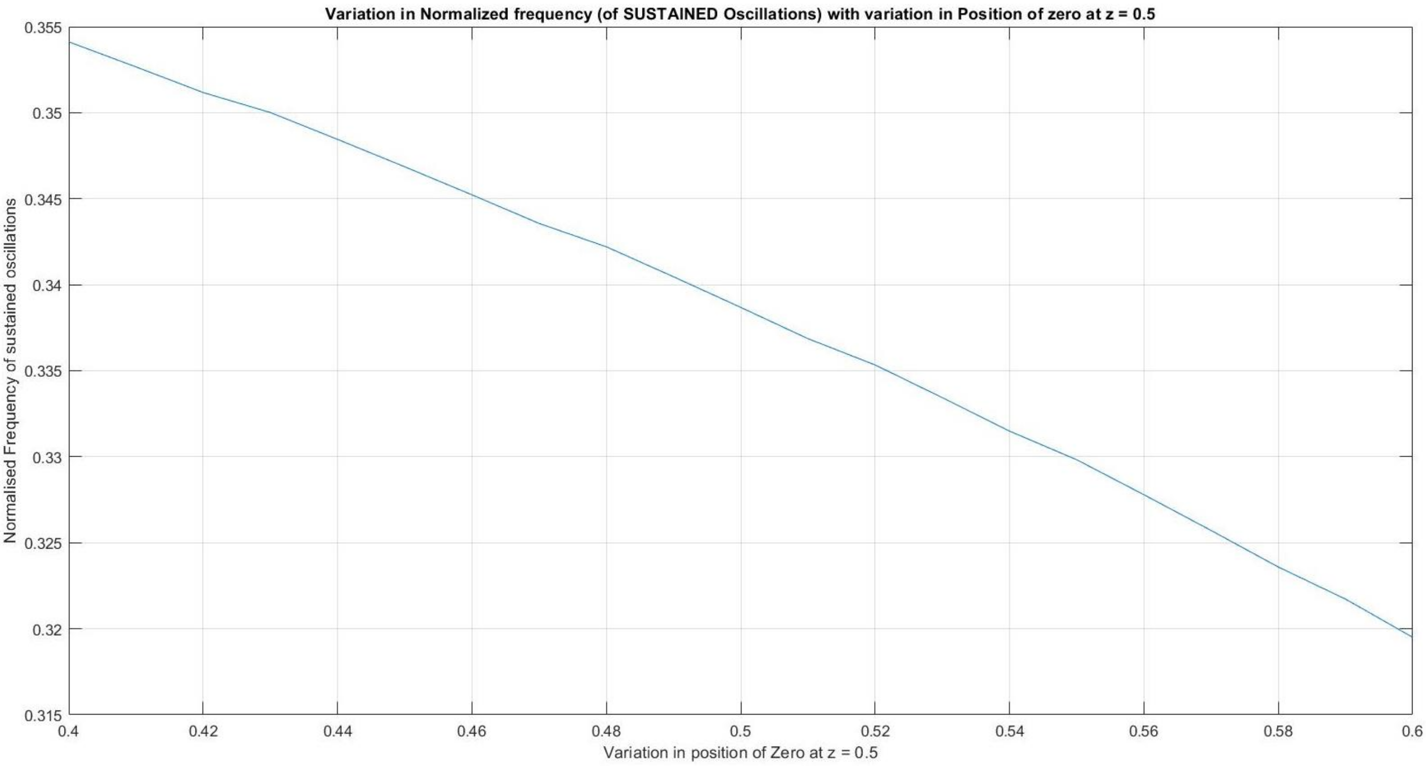
We now try to study the characteristics of the system when zeroes are varied.

EFFECT OF VARIATION OF ZEROES ON FREQUENCY OF SUSTAINED OSCILLATIONS

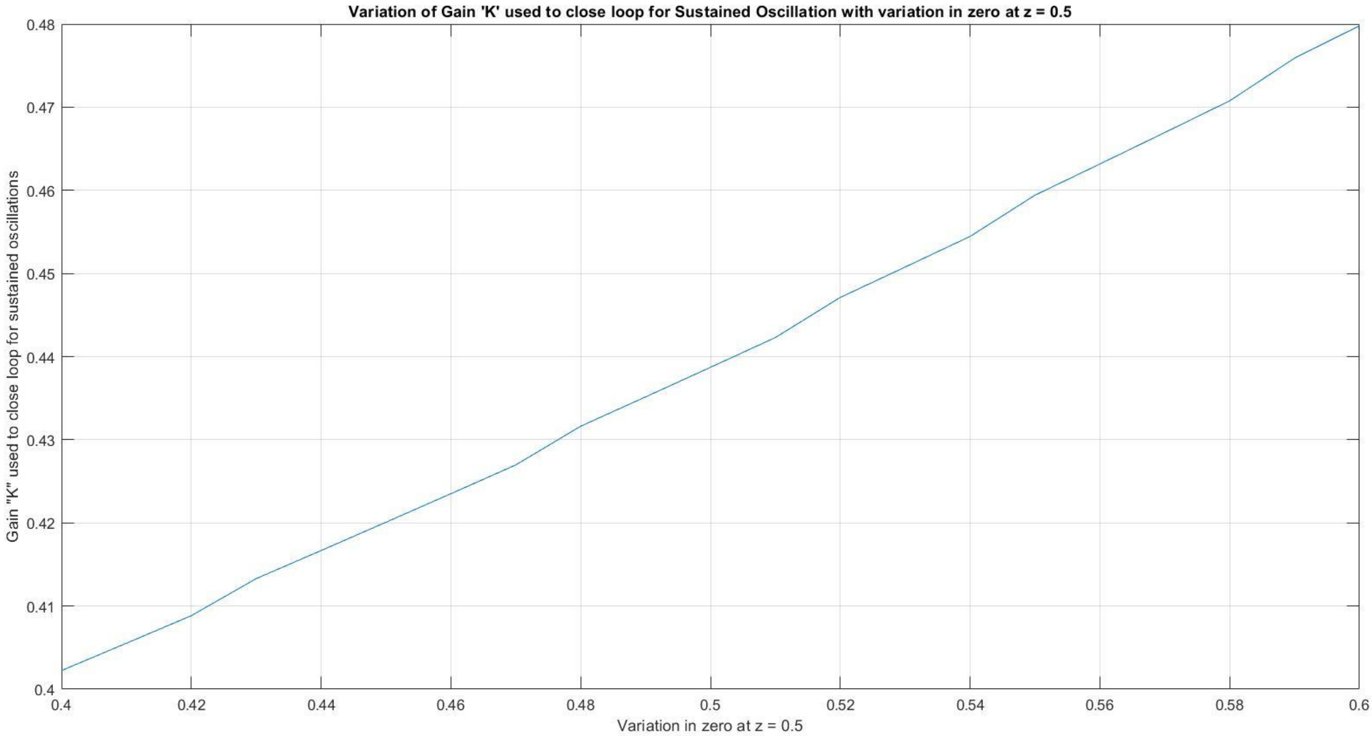
As we know that the sustained oscillations are obtained at 2 different frequencies, therefore we study them separately.

1. **<u>Low frequency point (Right Hand Z plane)</u>**

   We vary the zero at $z = 0.5$ by an amount of $\pm 20\%$ that is $z \in [0.4, 0.6]$ and note down the frequency of sustained oscillations (Normalised) and keep a track of the feedback gain 'K' by which we need to close the loop for marginal stability. We plot these variations and obtain the following trends:

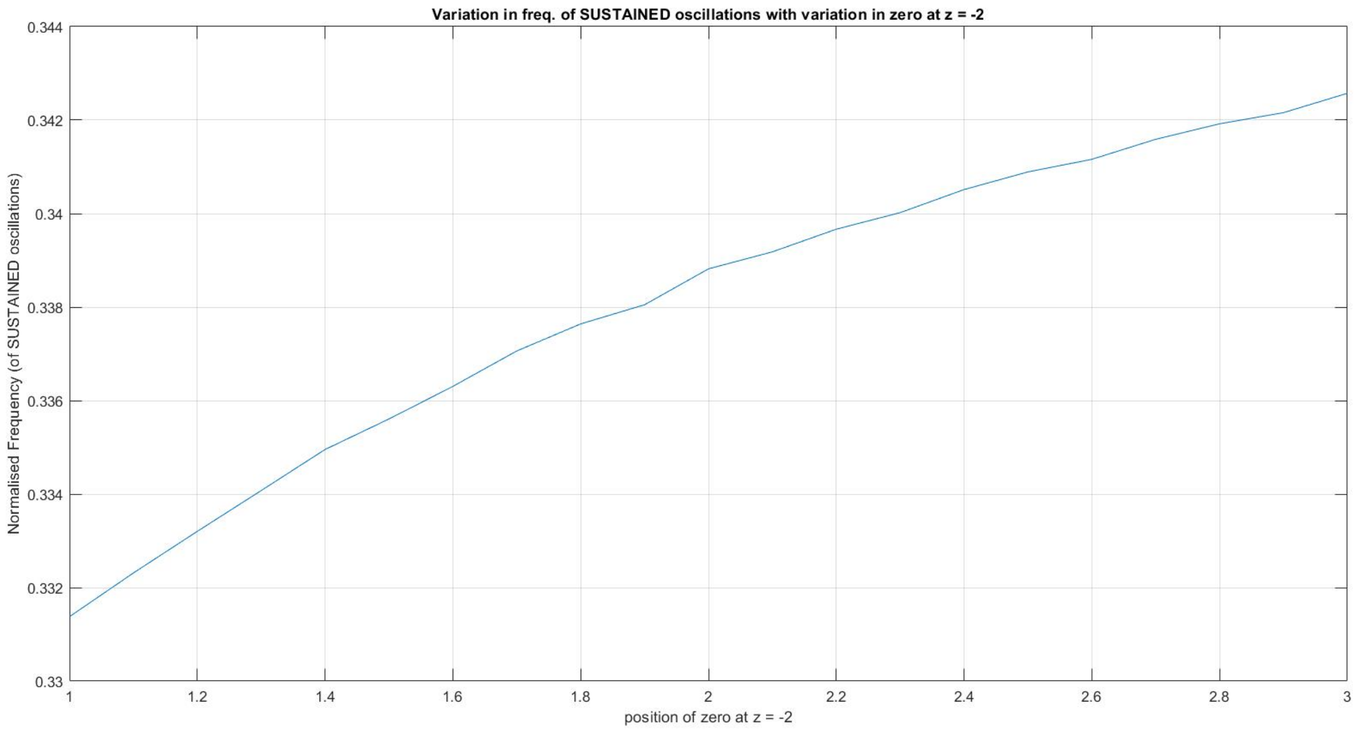


We see a decreasing trend (linearly) of normalized frequency when the zero at 0.5 is varied from **0.4 to 0.6**. The normalised frequency decreases approximately from **0.354 to 0.319.** We further plot the variation of gain "K" used to close the loop to obtain the sustained oscillations

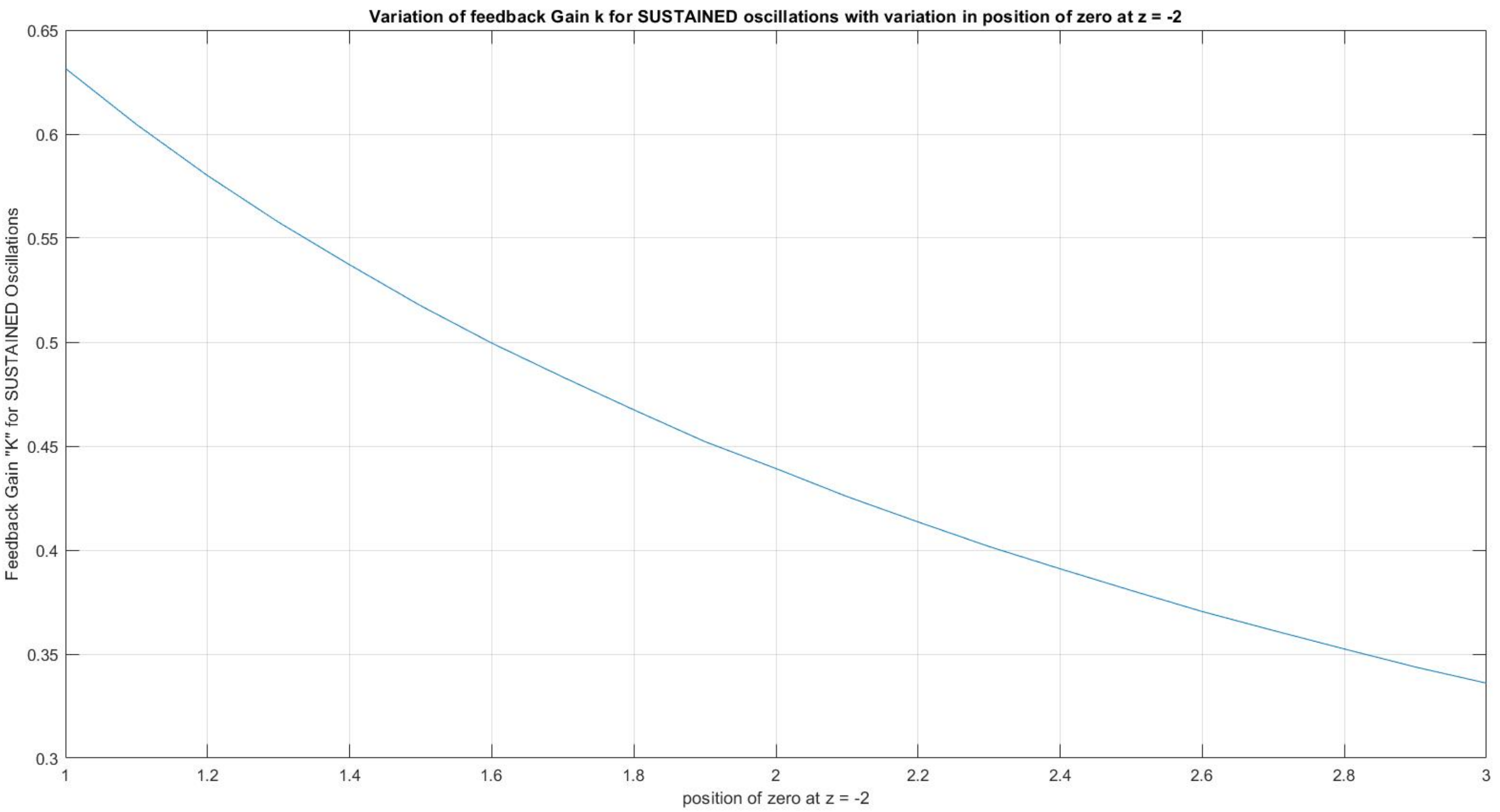


From the above plot, we observe that the gain "K" used to close the loop increases linearly approximately from **0.403 to 0.480** when the zero at **z = 0.5** is varied from **0.4 to 0.6**.

Now we study the effect of variation of zero at **z = -2** on the frequency of sustained oscillations **(Low frequency point)**.



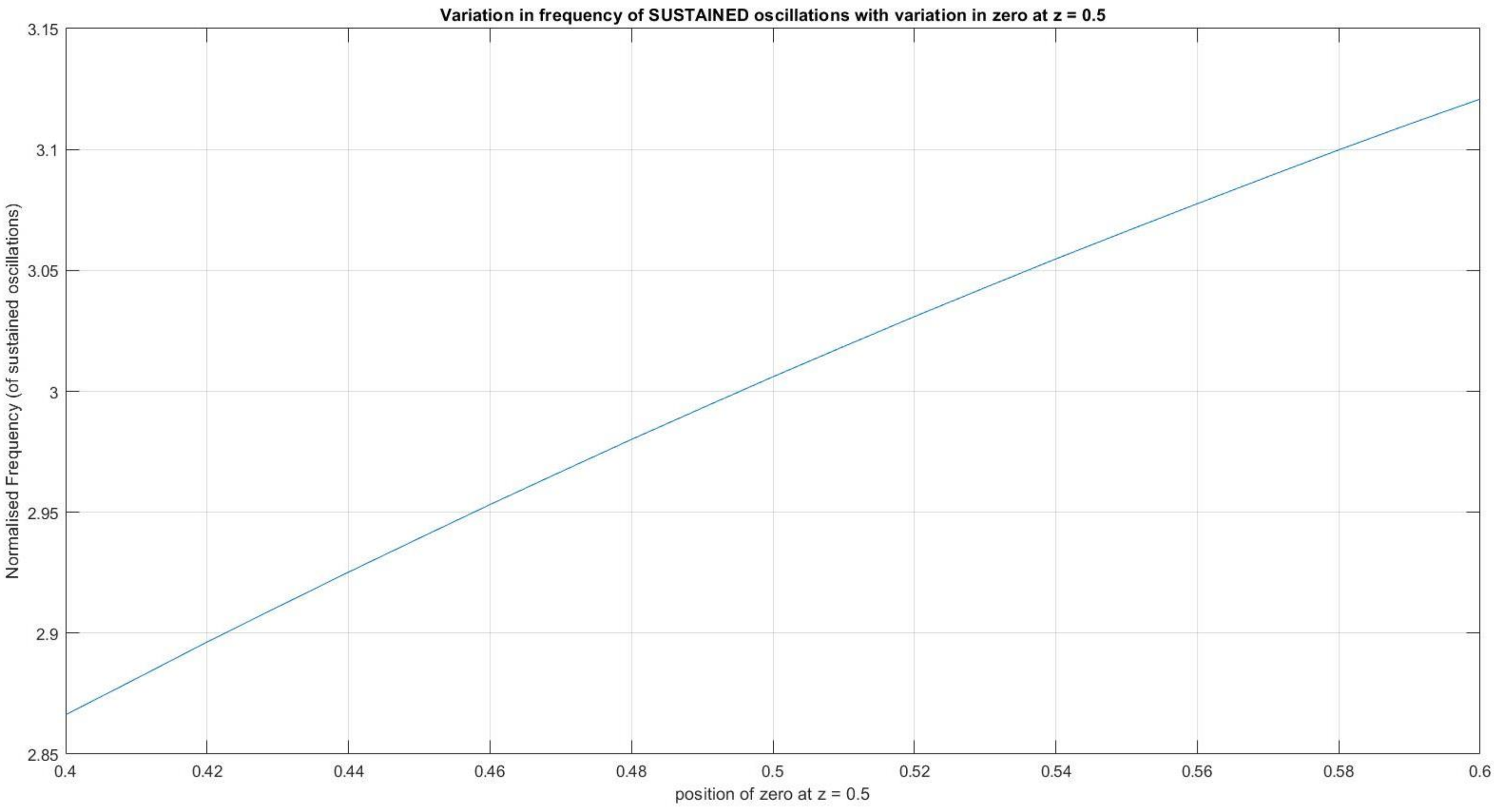
We see an increasing trend (almost linearly) of normalized frequency when the zero at **z = -2** is varied from **-1 to -3 (±50%)**. The normalised frequency increases approximately from **0.3315 to 0.3425.** we further plot the variation of gain "K" used to close the loop to obtain the sustained oscillations.



From the above plot, we observe that the gain "K" used to close the loop decreases exponentially approximately from **0.63 to 0.34** when the zero at **z = -2** is varied from **-1 to -3**.

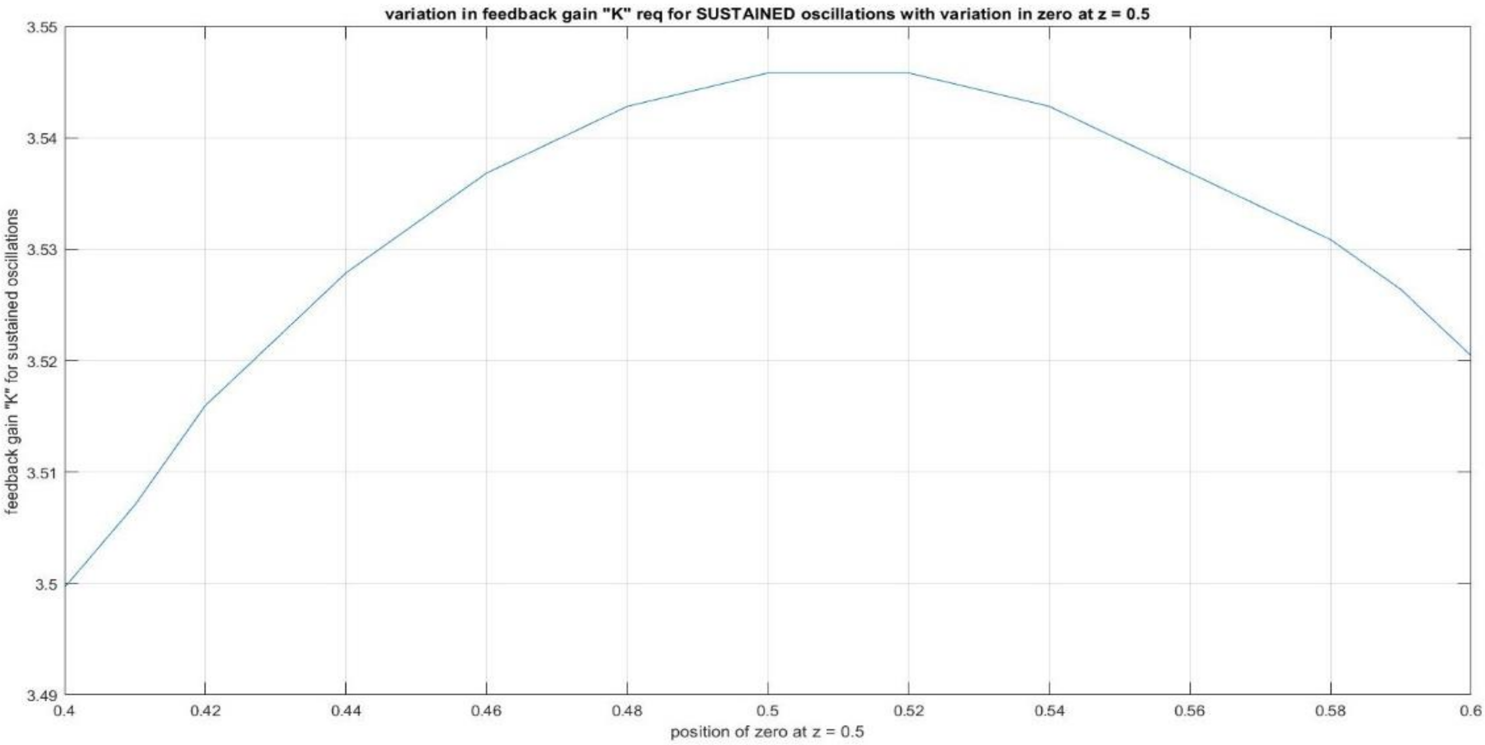
2. High frequency point (Left Hand Z plane)

We vary the zero at z = 0.5 by an amount of $\pm$20% that is z ∈ [0.4, 0.6] and note down the frequency of sustained oscillations (Normalised) and keep a track of the feedback gain 'K' by which we need to close the loop for marginal stability. We plot these variations and obtain the following trends:



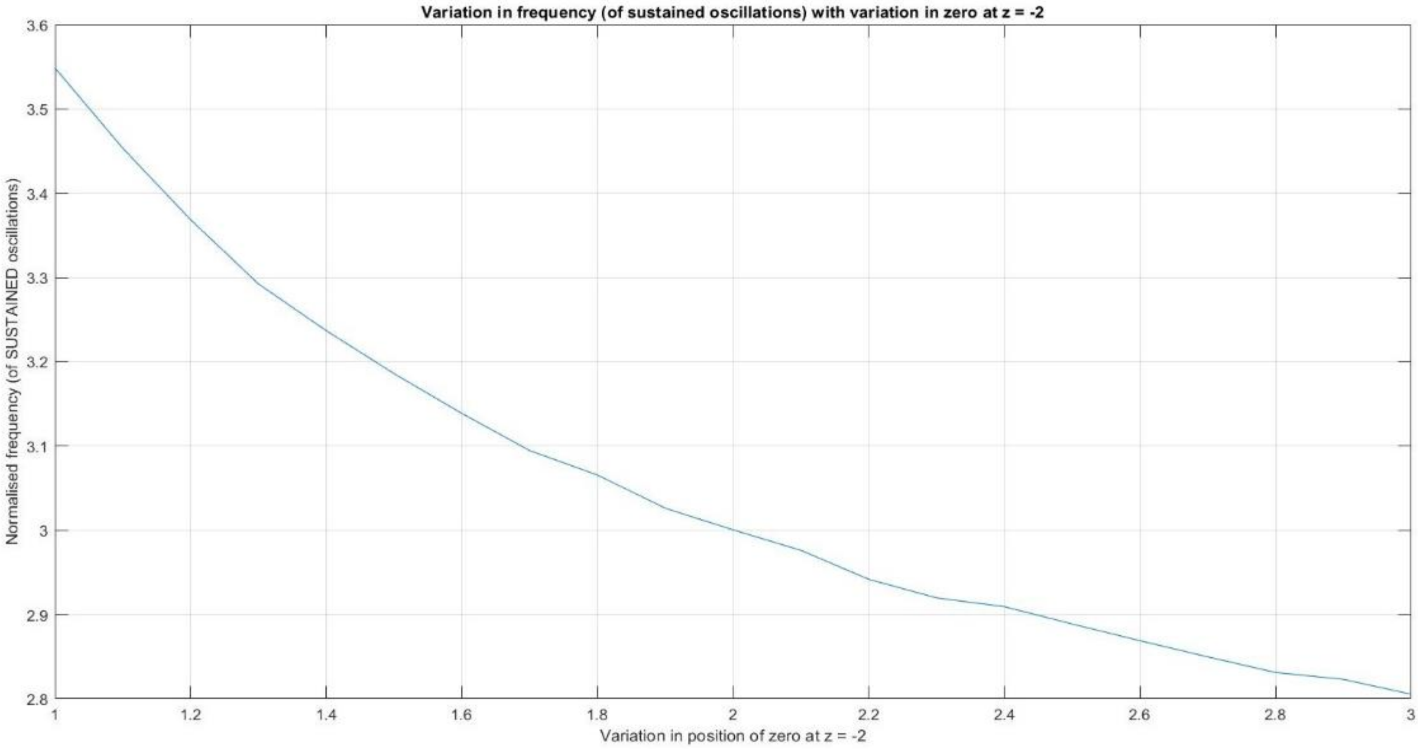
We see an increasing trend (linearly) of normalized frequency when the zero at **z = 0.5** is varied from **0.4 to 0.6**. The normalised frequency increases approximately from **2.865 to 3.123.**

We further plot the variation of gain “K” used to close the loop to obtain the sustained oscillations.



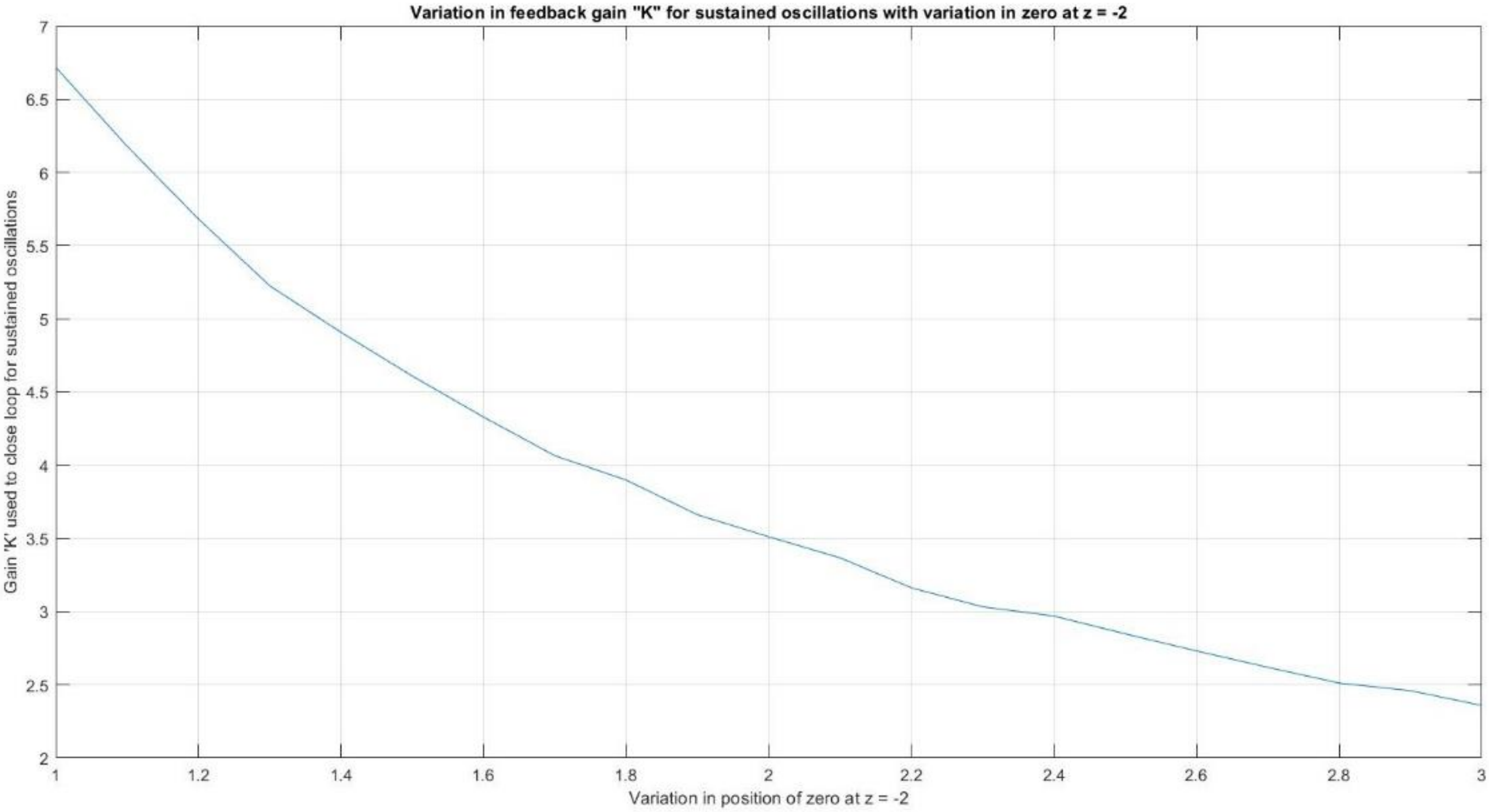
From the above plot, we observe that the gain “K” used to close the loop varies in an inverted parabolic fashion i.e., increases approximately from **3.5 to 3.545** when zero is varied from **0.4 to 0.51** and decreases to **3.52** as the zero reaches **0.6.**

Now we study the effect of variation of zero at **z = -2** on the frequency of sustained oscillations **(High frequency point)**.



We see a decreasing trend (almost exponentially) of normalized frequency when the zero at **z = -2** is varied from **-1 to -3 ($\pm$50%)**. The normalised frequency decreases approximately from **3.55 to 2.8.**

We further plot the variation of gain “K” used to close the loop to obtain the sustained oscillations.



From the above plot, we observe that the gain “K” used to close the loop decreases exponentially approximately from **6.73 to 2.4** when the zero at **z = -2** is varied from **-1 to -3**.

SENSITIVITY ANALYSIS OF NORMALIZED OSCILLATION FREQUENCY

Zero	Characteristics	High Frequency			Low Frequency		
		Max	Min	Bandwidth	Max	Min	Bandwidth
z = 0.5	Normalized frequency	3.123 (at z =0.6)	2.865 (at z =0.4)	0.258	0.354 (at z =0.4)	0.319 (at z =0.6)	0.035
	Feedback Gain (k)	3.545 (at z=0.51)	3.5 (at z =0.4)	0.045	0.480 (at z =0.6)	0.403 (at z = 0.4)	0.077
z = -2	Normalized frequency	3.55 (at z = -1)	2.8 (at z = -3)	0.75	0.3425 (at z =-3)	0.3315 (at z =-1)	0.011
	Feedback Gain (k)	6.73 (at z =-1)	2.4 (at z = -3)	4.33	0.63 (at z = -1)	0.34 (at z = -3)	0.29

The above table shows us the amount of variation occurring in normalized sustained oscillations frequency and feedback gain with variation of key parameters.

CONCLUSION

The initial system given to us was unstable as locus of the two roots lied outside the unit circle |z| = 1. We constructed a controller C(z)

$$\boldsymbol{C(z) = 0.17371 * \frac{z^2 - 1.758z + 0.7815}{(z - 0.1)(z - 0.01)}}$$

This made our system stable for Feedback Gain K ∊ [0.439, 3.5459] (approximately).

To study the variations of sustained oscillation frequency and Feedback Gain K to achieve required marginal stability, we varied the zeroes z = 0.5 by $\pm$ 20% and z = -2 by $\pm$ 50%. We plotted change in Normalized Sustained Oscillation Frequency against change of parameters (zeroes and Feedback Gain K) and noted down the observations.

We find that the high frequency sustained oscillations are more sensitive to the variation of zero at z =-2 and low frequency sustained oscillations are more sensitive to the variation of zero at z =0.5, however the feedback gain k is not affected as much as compared to the case of z =-2

MATLAB CODE USED

```
z = tf('z', -1);
G = (z^2 + 1.5*z - 1)/(z-1)^3;
kk = 1/(z-1)^3;
C = 0.17371*(z+2)*(z-0.5)*(z^2 - 1.758*z + 0.7815)/((z-0.1)*(z-0.01));
step(feedback(kk*C, 1)), grid;
```

```
k = logspace(-0.69, -0.22, 400);
zer = [0.4: 0.01: 0.6];
for idx = 1:21
    for i = 1:400
        [ans1, ans2] = damp(feedback(kk*0.17371*(z+2)*(z-zer(idx))*(z^2 - 1.758*z + 0.7815)/((z-0.1)*(z-0.01)), k(i)));
        for j = 1:5
            if(abs(ans2(j) - 0) < 1e-2)
                idx1 = j;
                break
            end
        end
        if(ans2(idx1) < 1e-2 && ans2(idx1) > 0)
            f(idx) = 0.2*ans1(idx1);
            aa(idx) = k(i);
            break
        end
    end
end
plot(zer, f);
plot(zer, aa);
```