Project: Discrete–Time Controller Design for a Two–DOF Helicopter

Project No: 2

Course: ELEC 6061 (M.Eng)

Name: Rishabh Singh Thakur

Student id: 40106439

Number of students in group: 1

Date: 22 April 2019

Department: Electrical and Computer Engineering

Name of Instructor: Prof Shahin Hashtrudi Zad

Contents

Appendix D

Table of Figures

1. Introduction

As discussed in Project 1, the Two– Degree–of–Freedom Helicopter by Quansar is equipped with two propellers driven by DC motors. The front propeller (pitch propeller) is used to control the pitch angle θ and the back propeller (yaw propeller) is used to control the yaw angle ψ. The objective in this project is to design two (single–input–single–output) discrete–time controllers for the pitch and yaw angles.

In this follow up project we will be doing the following

I. Designing a controller for the yaw channel
II. Combining the pitch, yaw controllers and the plant in state space form into a MIMO system.
III. Assessing the effects of cross coupling between the two channels.

2. Yaw channel controller

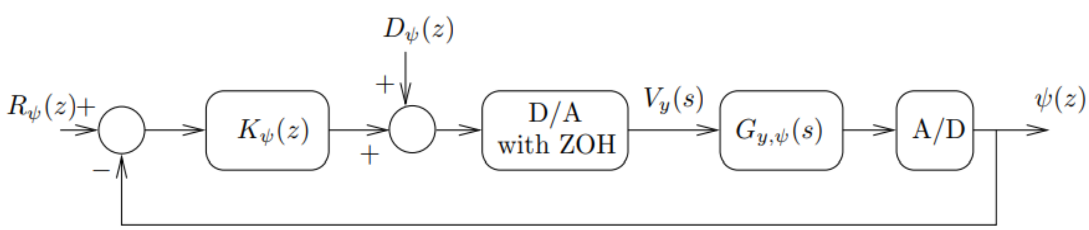


Figure 1: Yaw channel

2.1 Required specifications

2.1.1 Percentage of overshoot for step reference input

$$Mp \leq 0.2 \;=>\; e^{-\frac{pi*zeta}{sqrt(1-(zeta)^2)}}$$
$$=>\; zeta \geq 0.456$$

2.1.2 Settling time of step response

$$ts \leq 16\, sec \;=>\; (zeta * wn) \geq 0.2875$$

2.1.3 Rise Time of step response

$$tr \leq 2\, sec \;=>\; wn \geq 0.9\, rad/sec$$

2.1.4 Steady state error for step reference input

$$(ess)step = 0$$

$$=> 1/(1 + kp) = 0$$

Where $kp = \lim_{z->1} K(z)\, G(z)$

Therefore, either K(z) or G(z) should have at least one pole at z = 1.

2.1.5 Steady state output in response to step disturbance

For $d[n] = 1[n] and\, r[n] = 0$

$(yss) = 0$

$$=> \lim_{z->1}(z - 1)\, Y(z) = 0$$

$$=> \lim_{z->1}(z - 1) * \frac{G(z)}{1+K(z)G(z)} * \frac{z}{z-1} = 0$$

$$=> \lim_{z->1}(z - 1) * \frac{G(z)}{1+K(z)G(z)} = 0$$

Therefore $K(z)$ should have at least one pole at z=1

2.1.6 Step disturbance must settle within 16 sec

$\frac{Y(z)}{D(z)} = \frac{G(z)}{1+K(z)G(z)}$. Since the characteristic equation $1 + G(z)K(z)$ is the same as in the case of $\frac{Y(z)}{R(z)}$ the settling time criteria obtained in 2.12 applies here as well.

2.1.7 Determining sampling period Ts

Using the bandwidth equation $Bw = \big((-1.196 * zeta) + 1.85\big) * wn$ with $zeta = 0.456$ and $wn = 0.9$ we get $Bw = 1.174\ rad/sec$.

$=> Ts = \frac{2*pi}{25*Bw} = 0.214\ sec$ and $Ts \leq \frac{tr}{6} => wn \leq \frac{tr}{6*Ts}$

$=> wn \leq 1.557$

2.2 Design using root locus

To satisfy the requirements from the previous section we choose the following design parameters:

$Zeta = 0.5$

$Wn = 1$

$Ts = 0.214\ sec$

Using these we proceed with the design of the yaw channel controller.

$e^{-zeta*wn*Ts} = 0.8985$

$wn * \sqrt{1-(zeta)^2} * Ts = 0.185$

$Zp = 0.8831 + 0.1652i$

Where Zp is the desired point on root locus according to our specifications.

The given plant transfer function for yaw channel is

$$G_{p,\psi}(s) = \frac{\psi(s)}{V_y(s)} = \frac{7.461}{s(s+0.2701)}$$

The ZOH discrete equivalent of the above plant is

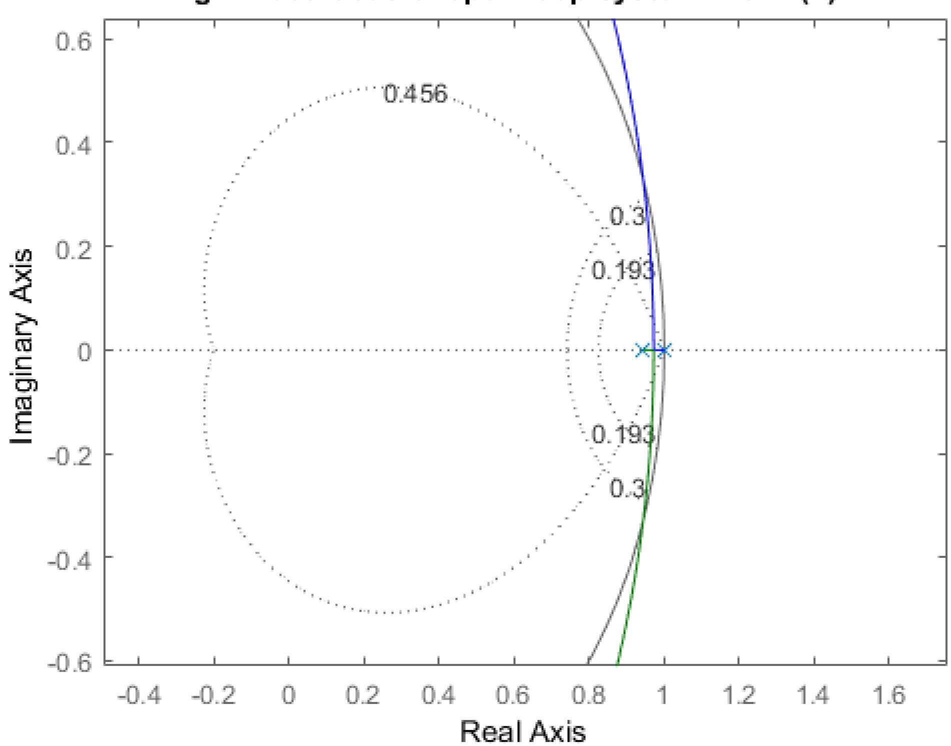
$$gz_{psi}(z) = \frac{0.1676\,z\ +\ 0.1644}{z^2 -\ 1.944\,z\ +\ 0.9438}$$

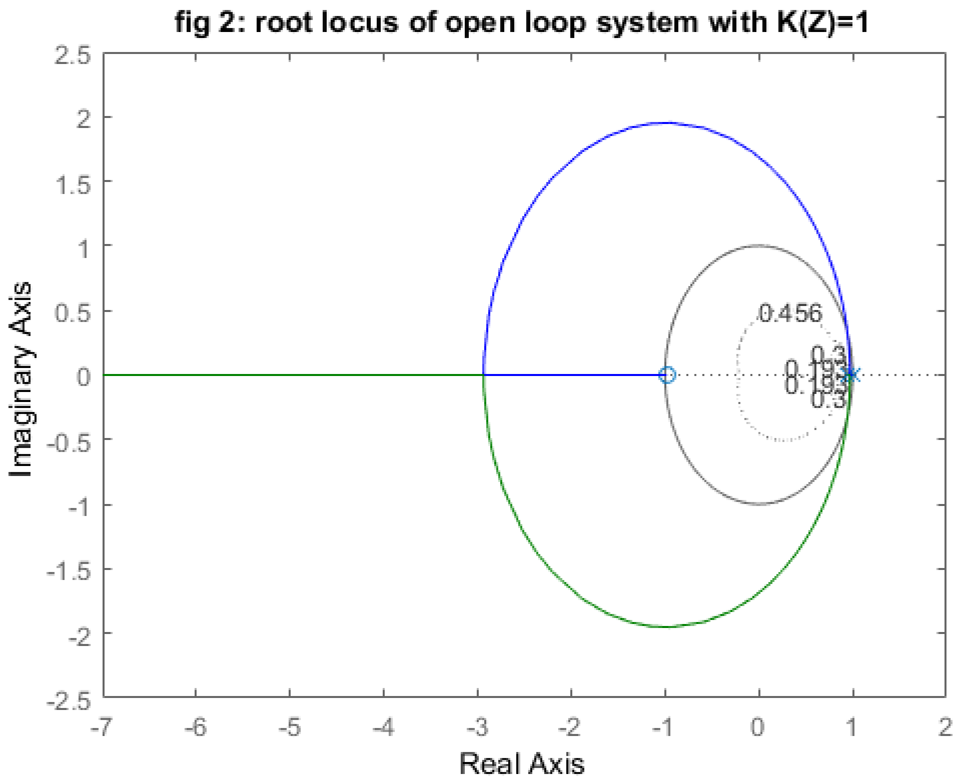
$$Poles\ =\ 1.0000, 0.9438$$

$$Zeros\ =\ -0.9809$$

The root locus plot for the above discrete plant for k=1 is shown in figure 2 below.

fig 2: root locus of open loop system with K(Z)=1





2.2.1 Controller with undesired overshoot (m file in Appendix A)

We can see from the above figures that since we want the steady state output in response to step disturbance to be zero we have to first design a lag compensator as per the condition in section 2.1.5.

$$K_{lag}(z) = \frac{z - z1}{z - p1} \quad (|z1| < |p1|)$$

Where $p1 = 1$ as per our requirement

To obtain z1 we use angle criterion:

$$\angle(Zp - z1) + \angle(Zp - zs1) - \angle(Zp - p1) - \angle(Zp - ps1) - \angle(Zp - ps2) = -\pi$$

Where z1, p1 are controller zeros and poles and $zs1 = -0.9809$, $ps1 = 0.9438$, $ps2 = 1$ are plant zeros and poles.

$$\angle(Zp - z1) + \tan^{-1}\left(\frac{0.1652}{0.9809+0.8831}\right) - (\pi - \tan^{-1}\left(\frac{0.1652}{1-0.8831}\right)) - (\pi - \tan^{-1}\left(\frac{0.1652}{0.9438-0.8831}\right)) - (\pi - \tan^{-1}\left(\frac{0.1652}{1-0.8831}\right)) = -\pi$$

Therefore $\angle(Zp - z1) = 174.39°$

It is clear that just with the zero above the required phase angle can't be achieved and hence we use a Lead compensator in series.

$$K_{lead}(z) = k * \frac{z - z2}{z - p2} \quad (|z2| > |p2|)$$

Using the above controller the angle criterion becomes

$$\angle(Zp - z2) + \angle(Zp - z1) - \angle(Zp - p2) = 174.39°$$

Let the zeros z1,z2 contribute an angle of $90°$ each and the pole an angle of $5.61°$

Therefore $z1 = 0.8831, z2 = 0.8831$ and $\tan^{-1}\left(\frac{0.1652}{0.8831-p2}\right) = 5.61°$ which gives $p2 = -0.7991$

Using magnitude criterion we can get the value of k

$$\left|K(Zp)gz_{psi}(Zp)\right| = 1$$

$$=> \left|k * \frac{Zp - z2}{Zp - p2} * \frac{Zp - z1}{Zp - p1} * \frac{0.1676 * (Zp + 0.9809)}{Zp^2 - (1.944 * Zp) + 0.9438}\right| = 1$$

$$=> k = 1.426$$

Therefore $K(z) = 1.426 * \frac{z-0.8831}{z-1} * \frac{z-0.8831}{z+0.7991}$

Poles $p1 = 1$, $p2 = -0.7991$

Zeros $z1 = 0.8831$, $z2 = 0.8831$

Open loop transfer function $K(z)gz_{psi}(z)$ is

$$\frac{0.239\,z^3 - 0.1877\,z^2 - 0.2277\,z + 0.1828}{z^4 - 2.145\,z^3 + 0.5353\,z^2 + 1.364\,z - 0.7542}$$

$$Poles = -0.7991, 1.0000, 1.0000, 0.9438$$

$$Zeros = -0.9809, 0.8831, 0.8831$$

With the above controller we get the following root locus plot as shown in figure 3 below

fig 3: root locus of open loop system with Lead-Lag compensation (controller with undesired overshoot)

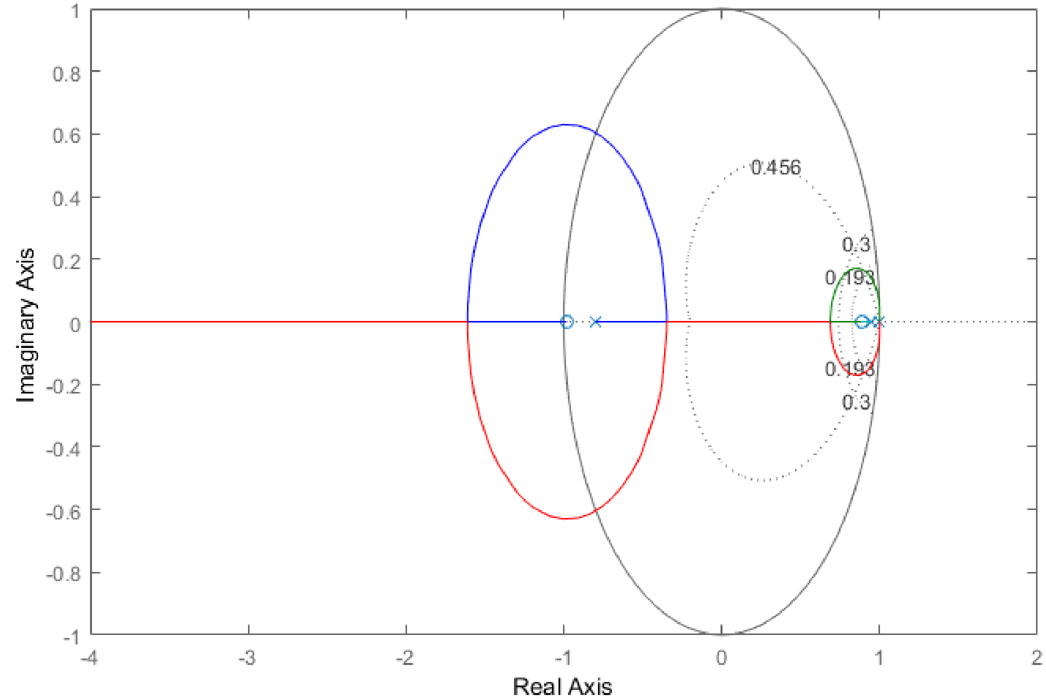
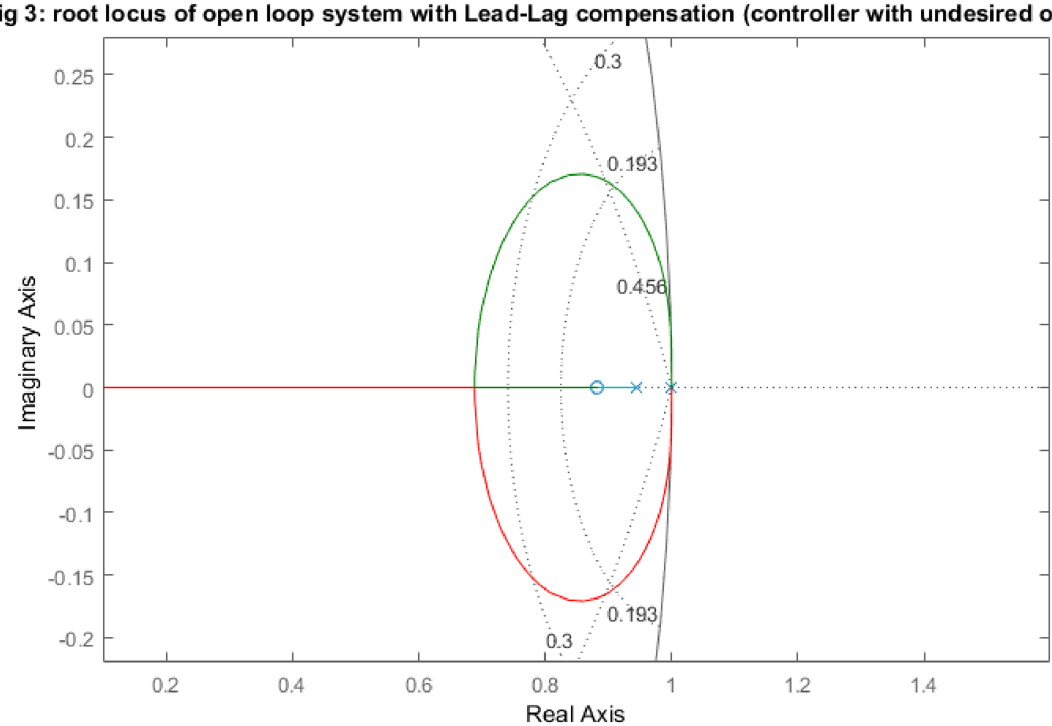


fig 3: root locus of open loop system with Lead-Lag compensation (controller with undesired overshoot)



The closed loop transfer function $\frac{K(z)*gz_{psi}(z)}{1+K(z)*gz_{psi}(z)}$ is as follows

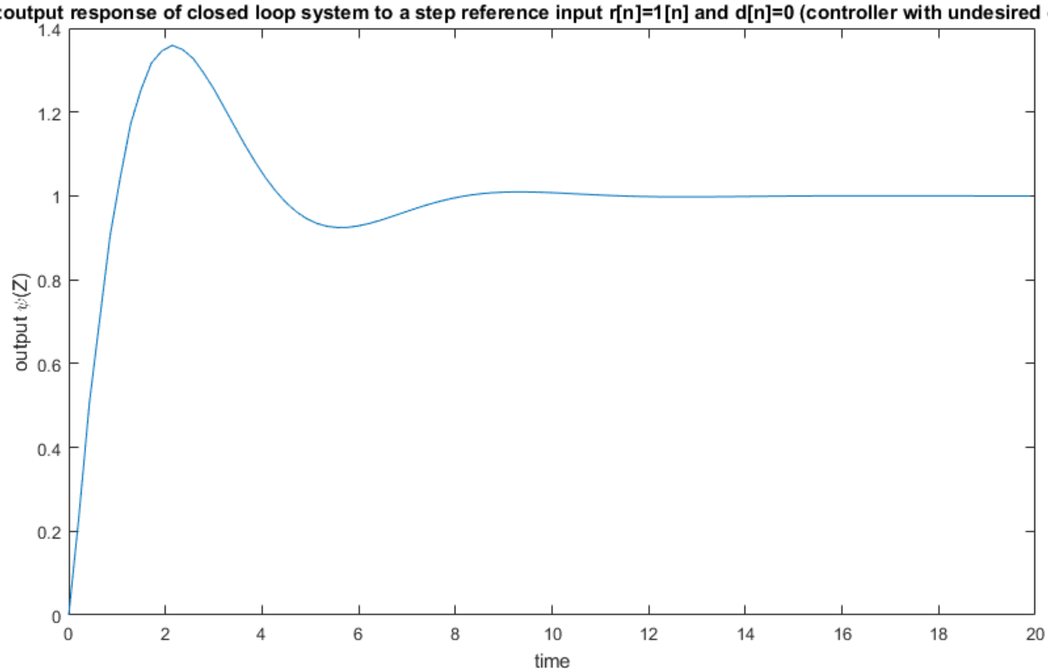
$$\frac{0.239\,z^3 - 0.1877\,z^2 - 0.2277\,z + 0.1828}{z^4 - 1.906\,z^3 + 0.3476\,z^2 + 1.136\,z - 0.5714}$$

$$Poles = -0.7740 + 0.0000i,\ 0.8834 + 0.1682i, 0.8834 - 0.1682i, 0.9129 + 0.0000i$$

$$zeros = -0.9809, 0.8831, 0.8831$$

The step response is shown in figure 4 below

fig 4:output response of closed loop system to a step reference input r[n]=1[n] and d[n]=0 (controller with undesired overshoot)

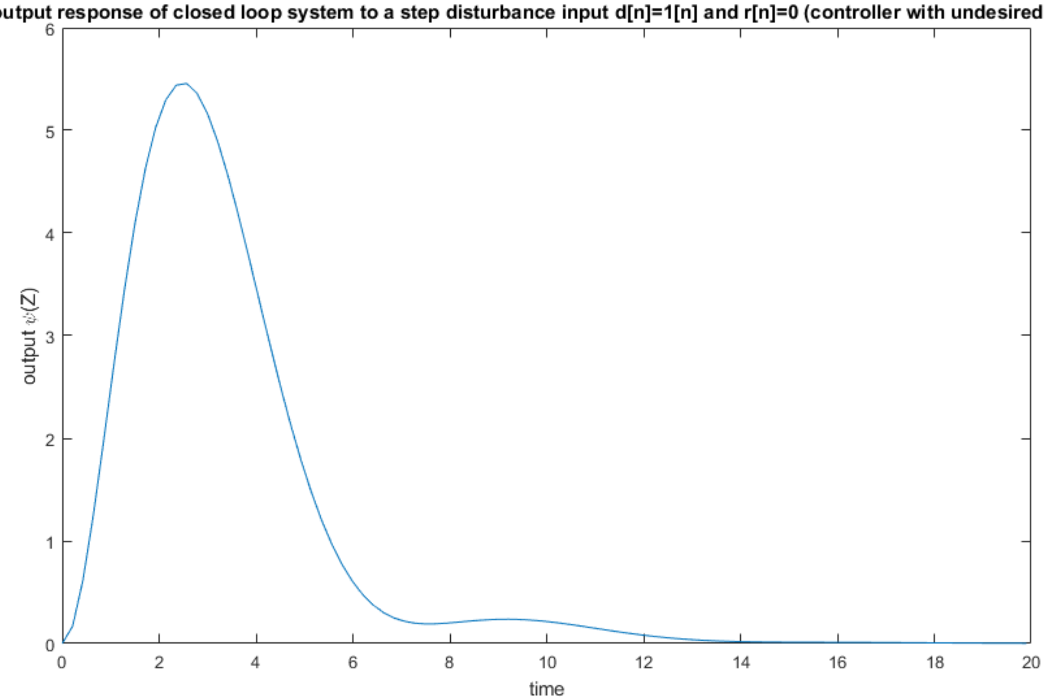


The step info for the above graph is as below

$$\begin{aligned} ref\_output\_resp = \ & RiseTime: 0.6420 \\ & SettlingTime: 7.4900 \\ & SettlingMin: 0.9050 \\ & SettlingMax: 1.3592 \\ & Overshoot: 35.9200 \\ & Undershoot: 0 \\ & Peak: 1.3592 \\ & PeakTime: 2.1400 \end{aligned}$$

The output response to step disturbance is shown in figure 5 below

fig 5:output response of closed loop system to a step disturbance input d[n]=1[n] and r[n]=0 (controller with undesired overshoot)



The step info for the above graph is as below

$$disturbance\_output\_resp \ = \ RiseTime: NaN$$

$$SettlingTime: 11.7700$$

$$SettlingMin: NaN$$

$$SettlingMax: NaN$$

$$Overshoot: 0$$

$$Undershoot: 1.2721e + 17$$

$$Peak: 5.4576$$

$$PeakTime: 2.5680$$

We can see that the rise time is too low (ie too fast) and overshoot is 35.92% which is not acceptable. This behavior can be because of the dominant poles of the plant which are very close to the unit circle. Since the rise time is too low the desired point on root locus can be moved farther to the right with decreasing wn values so as to increase the rise time and the damping ratio can also be increased by taking the desired point on root locus close to real axis. Therefore, the controller poles and zeros and hence the root locus needs to be brought closer to the unit circle for our second order system approximation to work.

2.2.2 Controller with undesired disturbance (m file in Appendix B)

Through multiple iterations, we see that by moving the zeros to $z1 = 0.9382, z2 = 0.9382$ and $p2 = 0.92$ with a gain of 1.2109 we get a similar root locus as in figure 3 but with a smaller circle close to unit circle on the right as shown below in figure 6

Therefore $K(z) = 1.2109 * \frac{z-0.9382}{z-1} * \frac{z-0.9382}{z-0.92}$

The open loop transfer function $K(z)gz_{psi}(z)$ is

$$\frac{0.2029\,z^3 - 0.1817\,z^2 - 0.1949\,z + 0.1752}{z^4 - 2.024\,z^3 + 0.1793\,z^2 + 1.713\,z - 0.8683}$$

$$poles = -0.9200, 1.0000, 1.0000, 0.9438$$

$$Zeros = -0.9809 + 0.0000i, 0.9382 + 0.0000i, 0.9382 - 0.0000i$$

The closed loop transfer function $\frac{K(z)*gz_{psi}(z)}{1+K(z)*gz_{psi}(z)}$ is as follows

$$\frac{0.2029\,z^3 - 0.1817\,z^2 - 0.1949\,z + 0.1752}{z^4 - 1.821\,z^3 - 0.002388\,z^2 + 1.518\,z - 0.6931}$$

$$Poles = -0.9131 + 0.0000i, 0.9426 + 0.0000i, 0.8957 + 0.0557i, 0.8957 - 0.0557i$$

$$zeros = -0.9809, 0.9382, 0.9382$$

The step response is shown in figure 7 below fig 6: root locus of open loop system with Lead-Lag compensation (controller with undesired disturbance)

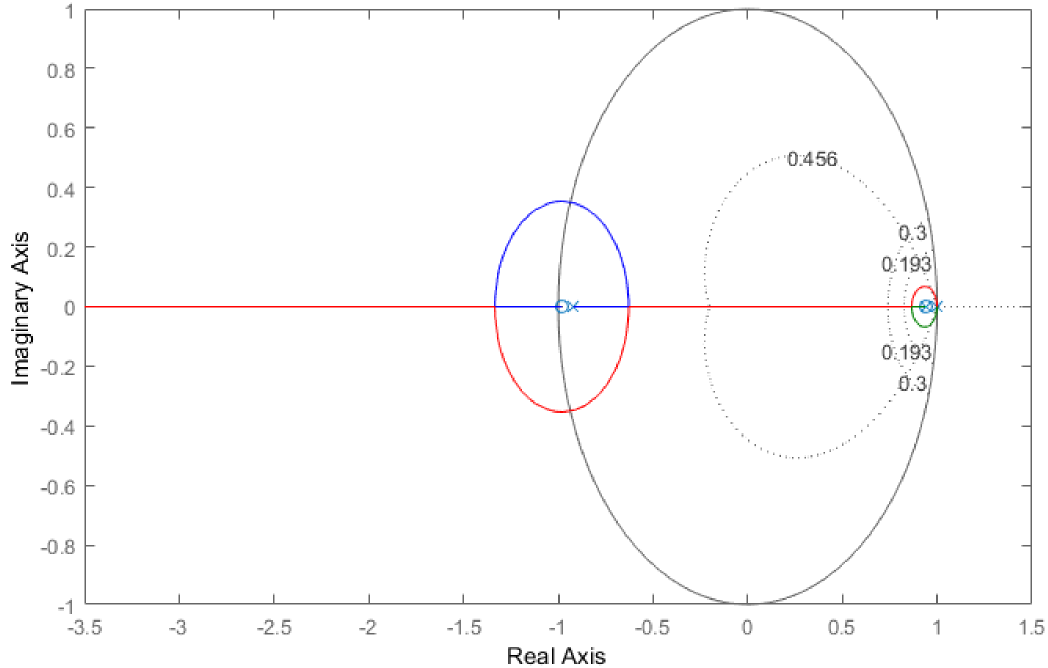


fig 6: root locus of open loop system with Lead-Lag compensation (controller with undesired disturbance)

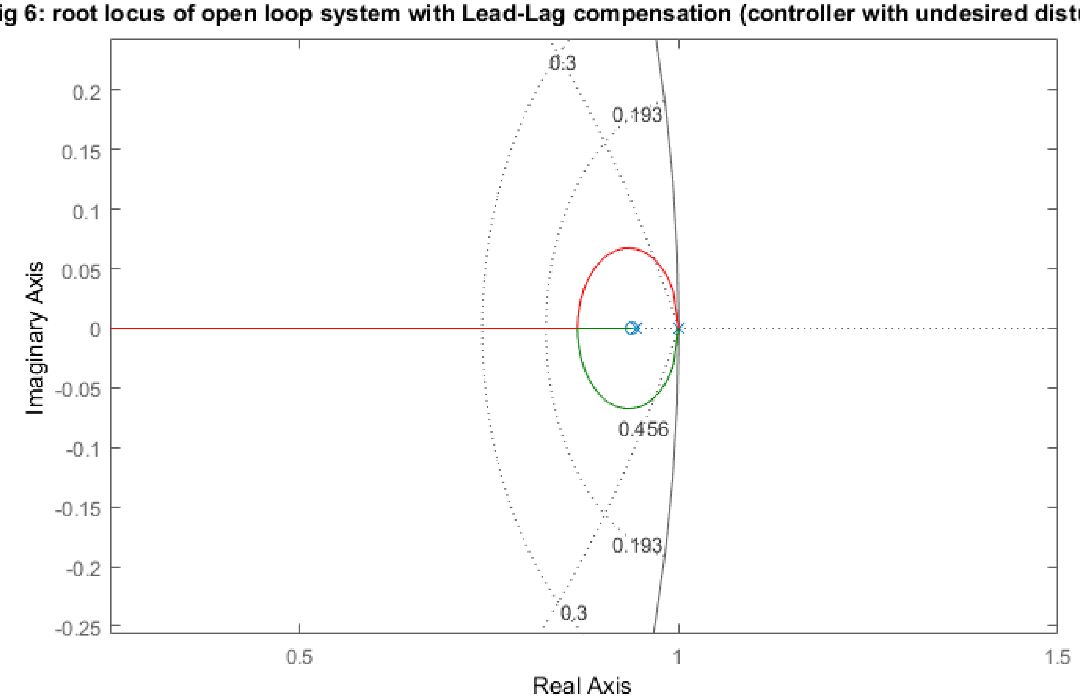
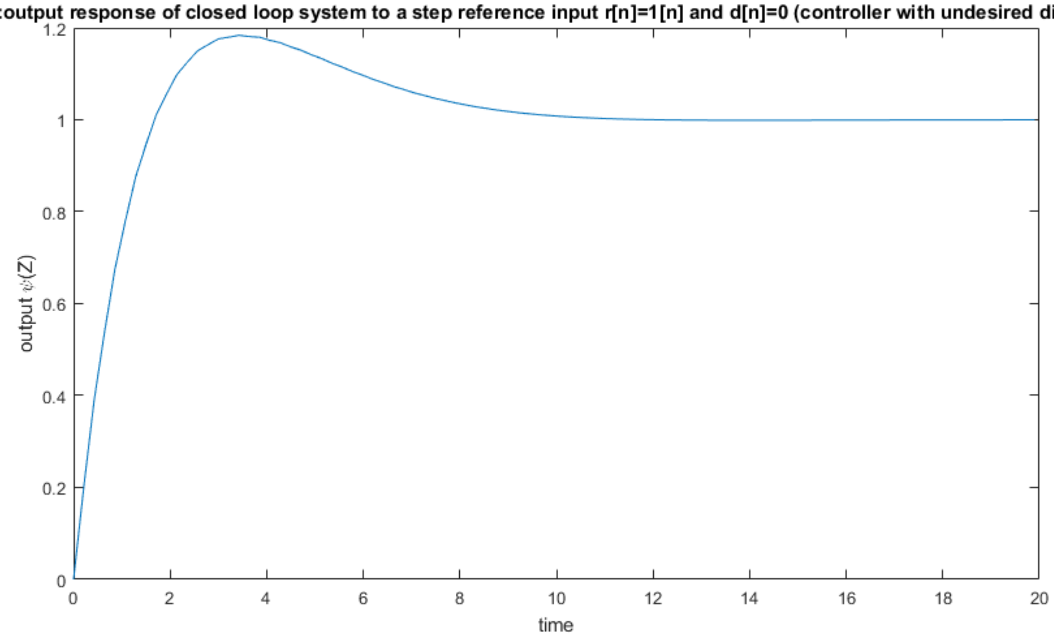


fig 7:output response of closed loop system to a step reference input r[n]=1[n] and d[n]=0 (controller with undesired disturbance)



The step info for the figure 7 is as follows

$$ref\_output\_resp \;=\; RiseTime: 1.2840$$

$$SettlingTime: 8.9880$$

$$SettlingMin: 0.9456$$

$$SettlingMax: 1.1838$$

$$Overshoot: 18.3831$$

$$Undershoot: 0$$

$$Peak: 1.1838$$

$$PeakTime: 3.4240$$

The output response to step disturbance is shown in fig 8 below

The step info for the figure 8 is as follows

$$disturbance\_output\_resp \;=\; RiseTime: NaN$$

$$SettlingTime: 20.3300$$

$$SettlingMin: NaN$$

$$SettlingMax: NaN$$

$$Overshoot: 0$$

$$Undershoot: 6.1895e+16$$

$$Peak: 11.1889$$

$$PeakTime: 4.2800$$

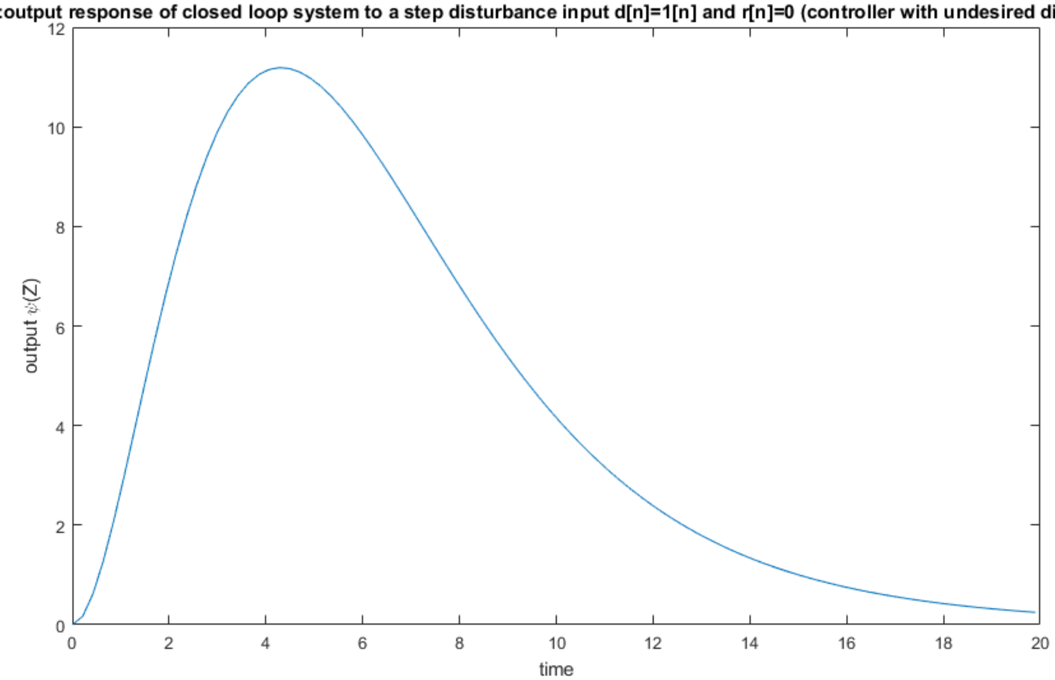


fig 8:output response of closed loop system to a step disturbance input d[n]=1[n] and r[n]=0 (controller with undesired disturbance)

2.2.3 Desired yaw channel controller (m file in Appendix C)

Now we see that the output settling time to step disturbance is 20.33 sec which doesn't meet our specification. It is caused because the controller zeros cancel the effect of poles at z=1 as they were moved closer. To mitigate this the zeros have to be moved away from point z=1 which can be done by adding an imaginary part to zeros by taking complex conjugate zeros. Hence the new controller zeros are $z1 = 0.9382 + 0.025i, z2 = 0.9382 - 0.025i$. This is the final design.

Therefore $K(z) = 1.2109 * \frac{z-(0.9382+0.025i)}{z-1} * \frac{z-(0.9382-0.025i)}{z-0.92}$

With the above controller we get the following root locus plot as shown in figure 9 below

The open loop transfer function $K(z)gz_{psi}(z)$ is

$$\frac{0.2029\, z^3 - 0.1817\, z^2 - 0.1948\, z + 0.1754}{z^4 - 2.024\, z^3 + 0.1793\, z^2 + 1.713\, z - 0.8683}$$

$$poles = -0.9200, 1.0000, 1.0000, 0.9438 Zeros = -0.9809 + 0.0000i, 0.9382 + 0.0250i, 0.9382 - 0.0250i$$

$$Zeros = -0.9809 + 0.0000i, 0.9382 + 0.0250i, 0.9382 - 0.0250i$$

fig 9: root locus of open loop system with Lead-Lag compensation (desired controller)

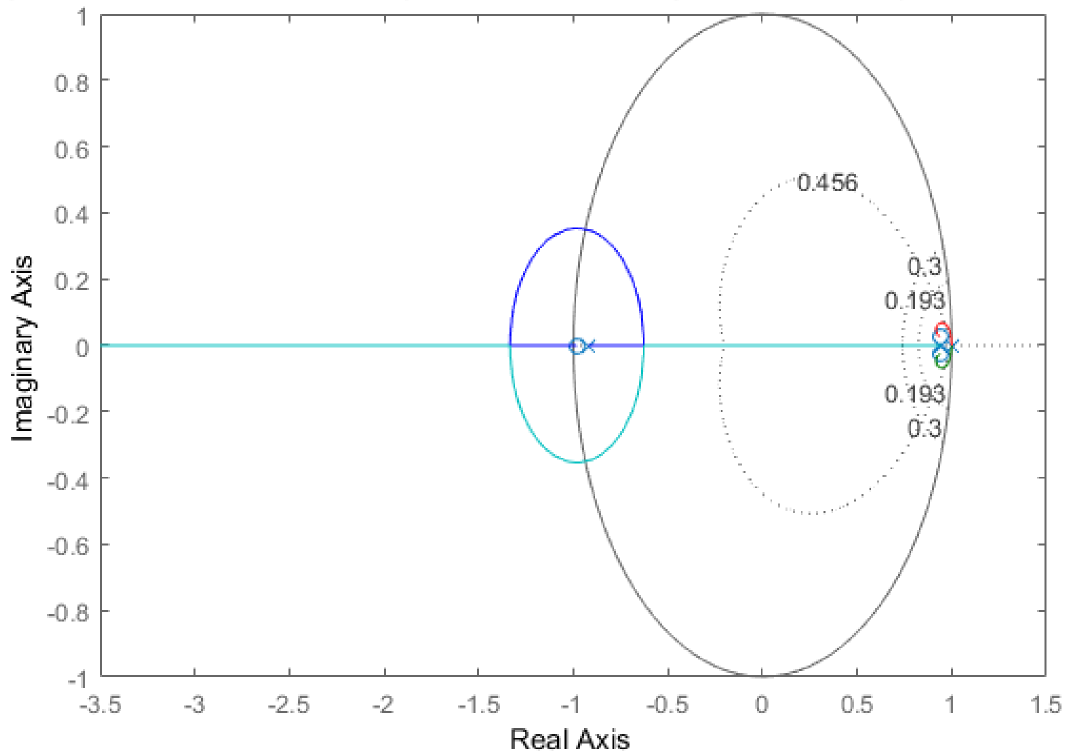
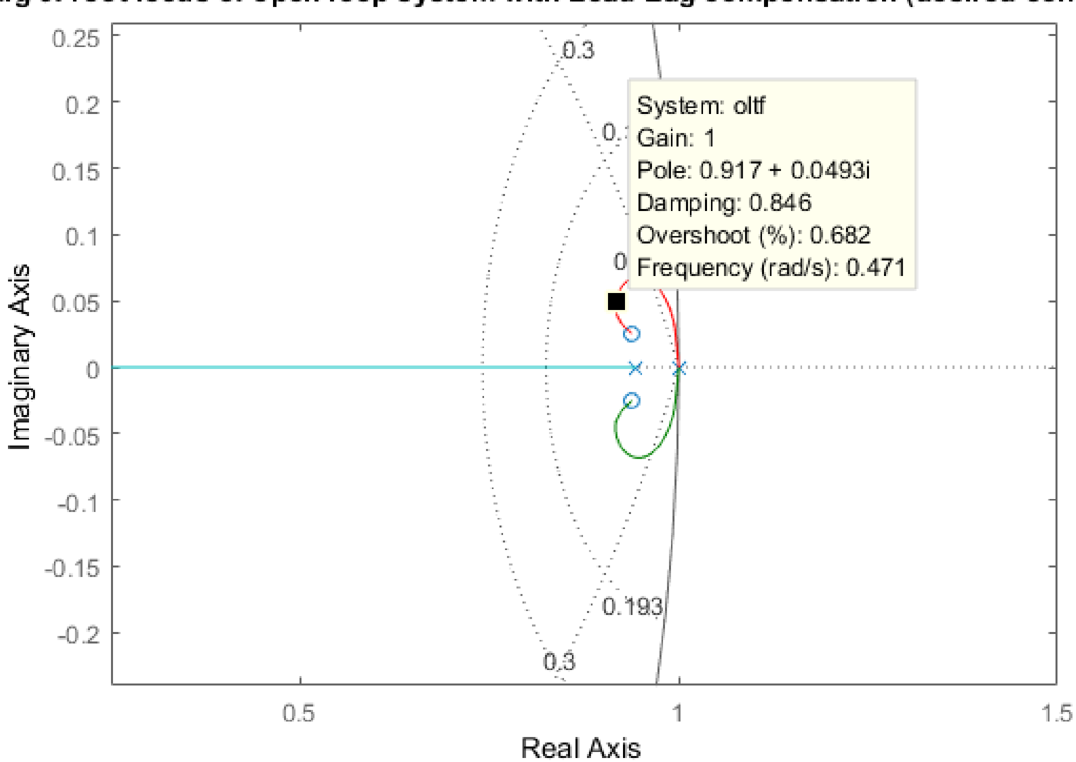


fig 9: root locus of open loop system with Lead-Lag compensation (desired controller)



The closed loop transfer function $\frac{K(z)*gz_{psi}(z)}{1+K(z)*gz_{psi}(z)}$ is as follows

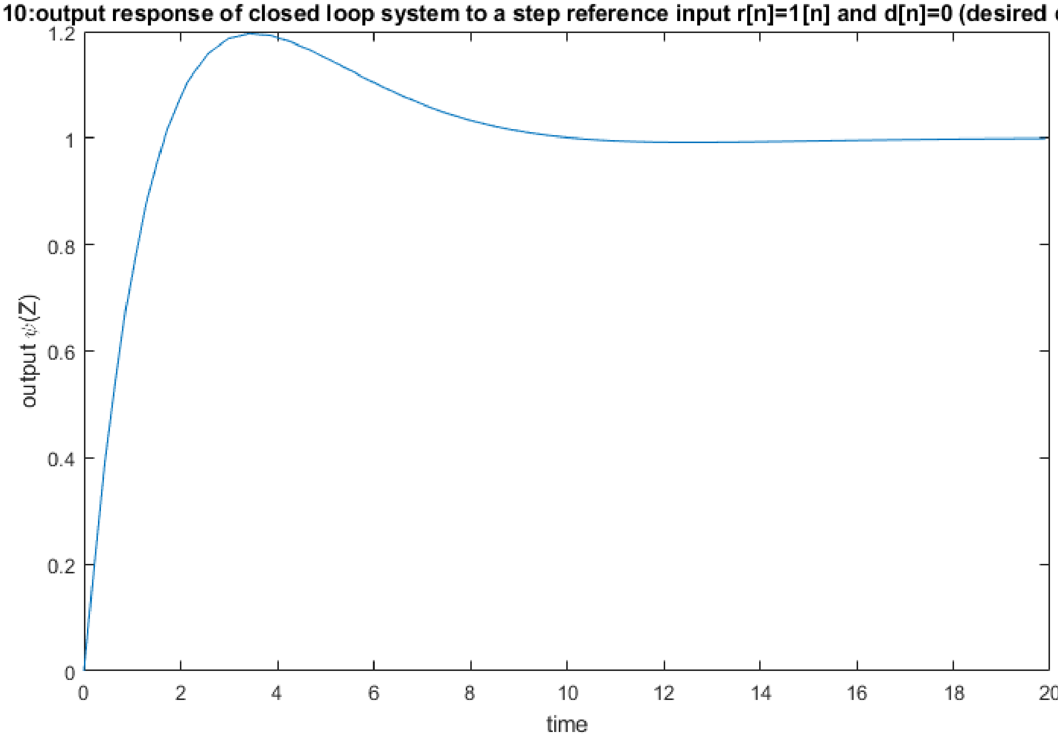
$$\frac{0.2029\,z^3 - 0.1817\,z^2 - 0.1948\,z + 0.1754}{z^4 - 1.821\,z^3 - 0.002388\,z^2 + 1.518\,z - 0.693}$$

$$Poles = -0.9131 + 0.0000i, 0.9167 + 0.0495i, 0.9167 - 0.0495i, 0.9005 + 0.0000i$$

$$zeros = -0.9809 + 0.0000i, 0.9382 + 0.0250i, 0.9382 - 0.0250i$$

The step response is shown in figure 10 below

fig 10:output response of closed loop system to a step reference input r[n]=1[n] and d[n]=0 (desired controller)

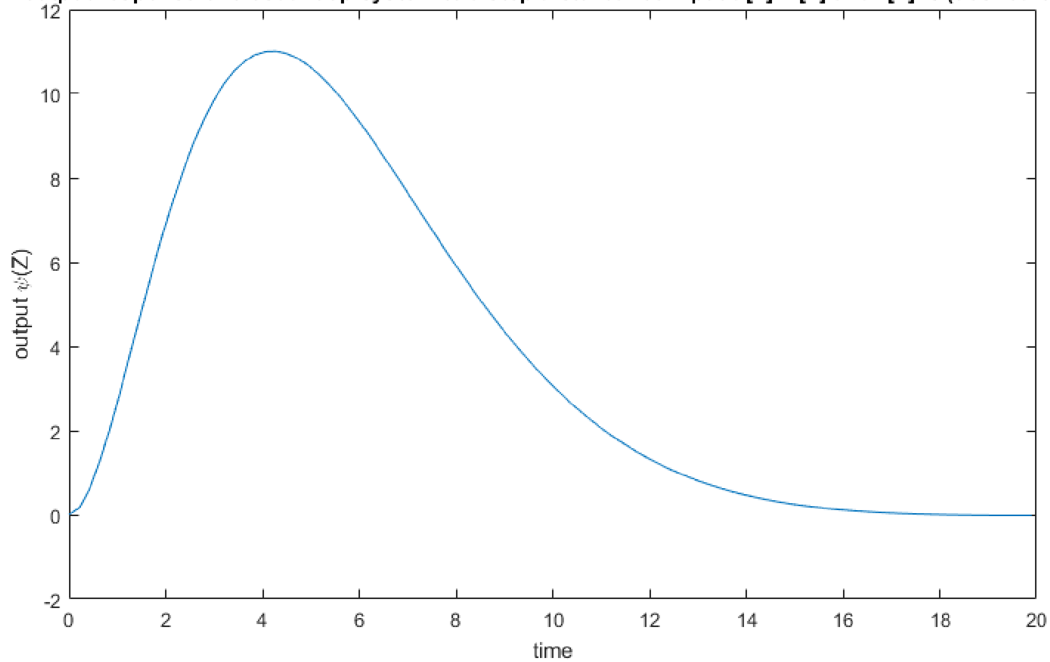


The step info for the figure 10 is as follows

$$ref\_output\_resp = RiseTime: 1.2840$$
$$SettlingTime: 8.7740$$
$$SettlingMin: 0.9483$$
$$SettlingMax: 1.1962$$
$$Overshoot: 19.6234$$
$$Undershoot: 0$$
$$Peak: 1.1962$$
$$PeakTime: 3.4240$$

The output response to step disturbance is shown in fig 11 below

fig 11:output response of closed loop system to a step disturbance input d[n]=1[n] and r[n]=0 (desired controller)



$$disturbance\_output\_resp\ =\ RiseTime\!: 0$$

$$SettlingTime\!: 15.1940$$

$$SettlingMin\!:\ -0.0198$$

$$SettlingMax\!:\ -8.9929e-04$$

$$Overshoot\!: 1.2765e+14$$

$$Undershoot\!: 7.0871e+16$$

$$Peak\!: 11.0098$$

$$PeakTime\!: 4.2800$$

The controller output is shown in figure 10_1 below

With a unit step r[n]=1[n] the controller output reaches a peak voltage of 1.2109. Therefore if 8 volts is the max controller output voltage to avoid dc motor saturation then the max step input that doesn't result in motor saturation for the system is $\frac{8}{1.2109} = 6.606$.

The step info for controller is as follows

$$controller\_output \;=\; RiseTime: 0$$

$$SettlingTime: 9.4160$$

$$SettlingMin: \; -1.2101$$

$$SettlingMax: 1.0106$$

$$Overshoot: 4.8717e+14$$

$$Undershoot: 4.8685e+14$$

$$Peak: 1.2109$$

$$PeakTime: 0$$

From the above graphs it's clear that the design specs have been met and this completes the design of our yaw channel controller.

3. Entire System Simulation(m file in Appendix D)

Now that we have controllers for both the channels (Using the pitch design from project 1) they can be applied simultaneously to the plant as a whole in state space form and can be arranged in a feedback loop as shown in figure 12 below

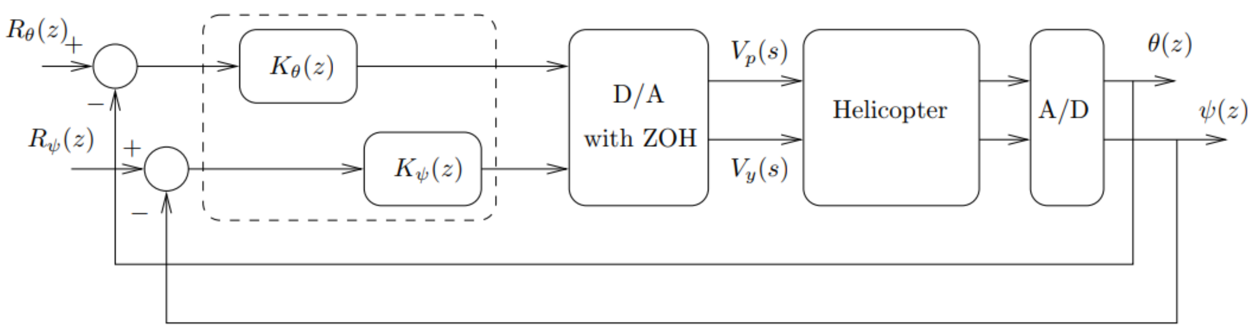


Figure 12: Closed Loop System

The following are the state space representations of the controllers, plant, open loop system and closed loop system. (Refer Appendix for code)

$kz\_theta\_ss \;=$

$a\ =$

$$\begin{array}{ccc} & x1 & x2 \\ x1 & 1.55 & -0.55 \\ x2 & 1 & 0 \end{array}$$

$b\ =$

$$\begin{array}{cc} & u1 \\ x1 & 0.25 \\ x2 & 0 \end{array}$$

$c\ =$

$$\begin{array}{ccc} & x1 & x2 \\ y1 & -0.1075 & 0.1403 \end{array}$$

$d\ =$

$$\begin{array}{cc} & u1 \\ y1 & 0.08 \end{array}$$

$kz\_psi\_ss\ =$

$a\ =$

$$\begin{array}{ccc} & x1 & x2 \\ x1 & 0.08 & 0.92 \\ x2 & 1 & 0 \end{array}$$

$b\ =$

$$\begin{array}{cc} & u1 \\ x1 & 2 \\ x2 & 0 \end{array}$$

$c\ =$

$$\begin{array}{ccc} & x1 & x2 \\ y1 & -1.088 & 1.09 \end{array}$$

$d\ =$

$$\begin{array}{c} u1 \end{array}$$

$y1\ \ 1.211$

$kz\_combined\ =$

$a\ =$

	$x1$	$x2$	$x3$	$x4$
$x1$	1.55	-0.55	0	0
$x2$	1	0	0	0
$x3$	0	0	0.08	0.92
$x4$	0	0	1	0

$b\ =$

	$u1$	$u2$
$x1$	0.25	0
$x2$	0	0
$x3$	0	2
$x4$	0	0

$c\ =$

	$x1$	$x2$	$x3$	$x4$
$y1$	-0.1075	0.1403	0	0
$y2$	0	0	-1.088	1.09

$d\ =$

	$u1$	$u2$
$y1$	0.08	0
$y2$	0	1.211

$gs\_combined\ =$

$a\ =$

	$x1$	$x2$	$x3$	$x4$
$x1$	0	1	0	0
$x2$	-2.745	-0.2829	0	0
$x3$	0	0	0	1

$x4$	0	0	0	-0.2701

$b =$

	$u1$	$u2$
$x1$	0	0
$x2$	37.2	3.531
$x3$	0	0
$x4$	2.389	7.461

$c =$

	$x1$	$x2$	$x3$	$x4$
$y1$	1	0	0	0
$y2$	0	0	1	0

$d =$

	$u1$	$u2$
$y1$	0	0
$y2$	0	0

$gz\_combined =$

$a =$

	$x1$	$x2$	$x3$	$x4$
$x1$	0.939	0.2033	0	0
$x2$	-0.5582	0.8815	0	0
$x3$	0	0	1	0.2079
$x4$	0	0	0	0.9438

$b =$

	$u1$	$u2$
$x1$	0.8262	0.07841
$x2$	7.564	0.7179
$x3$	0.05367	0.1676
$x4$	0.4968	1.551

$c =$

	$x1$	$x2$	$x3$	$x4$
$y1$	1	0	0	0
$y2$	0	0	1	0

$d =$

	$u1$	$u2$
$y1$	0	0
$y2$	0	0

$oltf\_combined =$

$a =$

	$x1$	$x2$	$x3$	$x4$	$x5$	$x6$	$x7$	$x8$
$x1$	0.939	0.2033	0	0	−0.08884	0.1159	−0.08528	0.0855
$x2$	−0.5582	0.8815	0	0	−0.8133	1.061	−0.7808	0.7827
$x3$	0	0	1	0.2079	−0.00577	0.00753	−0.1823	0.1827
$x4$	0	0	0	0.9438	−0.05342	0.06971	−1.687	1.692
$x5$	0	0	0	0	1.55	−0.55	0	0
$x6$	0	0	0	0	1	0	0	0
$x7$	0	0	0	0	0	0	0.08	0.92
$x8$	0	0	0	0	0	0	1	0

$b =$

	$u1$	$u2$
$x1$	0.0661	0.09495
$x2$	0.6051	0.8693
$x3$	0.004294	0.2029
$x4$	0.03974	1.879
$x5$	0.25	0
$x6$	0	0
$x7$	0	2
$x8$	0	0

c =

	$x1$	$x2$	$x3$	$x4$	$x5$	$x6$	$x7$	$x8$
$y1$	1	0	0	0	0	0	0	0
$y2$	0	0	1	0	0	0	0	0

d =

	$u1$	$u2$
$y1$	0	0
$y2$	0	0

$cltf\_combined$ =

a =

	$x1$	$x2$	$x3$	$x4$	$x5$	$x6$	$x7$	$x8$
$x1$	0.8729	0.2033	− 0.09495	0	− 0.08884	0.1159	− 0.08528	0.0855
$x2$	− 1.163	0.8815	− 0.8693	0	− 0.8133	1.061	− 0.7808	0.7827
$x3$	− 0.004294	0	0.7971	0.2079	− 0.00577	0.00753	− 0.1823	0.1827
$x4$	− 0.03974	0	− 1.879	0.9438	− 0.05342	0.06971	− 1.687	1.692
$x5$	− 0.25	0	0	0	1.55	− 0.55	0	0
$x6$	0	0	0	0	1	0	0	0
$x7$	0	0	− 2	0	0	0	0.08	0.92
$x8$	0	0	0	0	0	0	1	0

b =

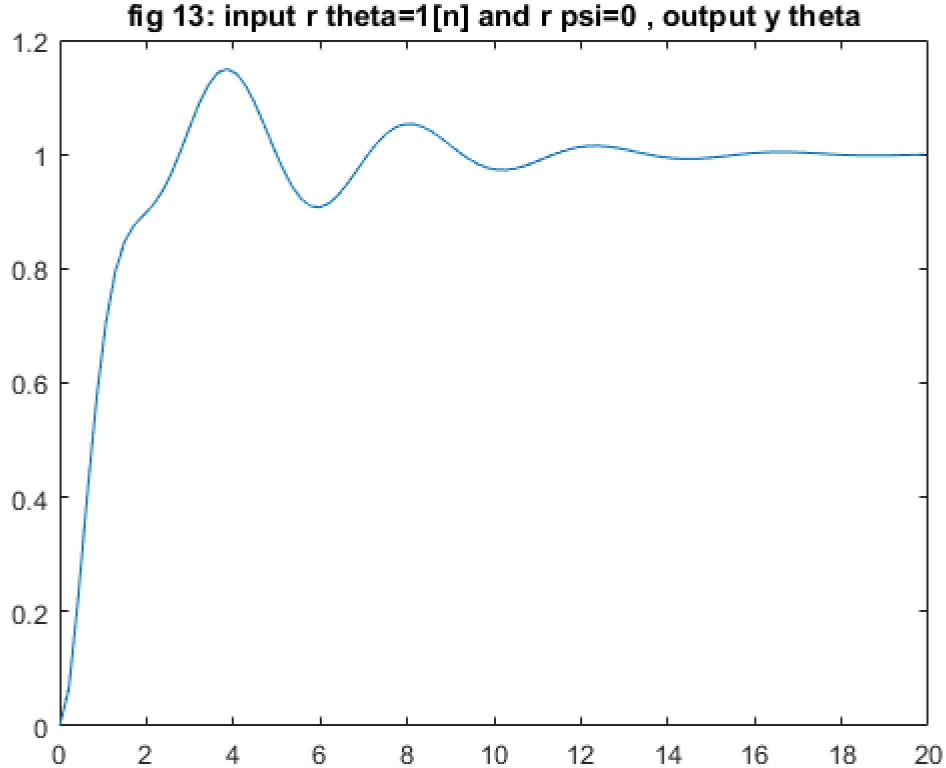
	$u1$	$u2$
$x1$	0.0661	0.09495
$x2$	0.6051	0.8693
$x3$	0.004294	0.2029
$x4$	0.03974	1.879
$x5$	0.25	0
$x6$	0	0
$x7$	0	2
$x8$	0	0

$$
c =
\begin{array}{c|cccccccc}
 & x1 & x2 & x3 & x4 & x5 & x6 & x7 & x8 \\
y1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
y2 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0
\end{array}
$$

$$
d =
\begin{array}{c|cc}
 & u1 & u2 \\
y1 & 0 & 0 \\
y2 & 0 & 0
\end{array}
$$

The pitch channel output Ɵ(z) to a step pitch channel input with $D_{\psi}(z)=0$ is shown in figure 13

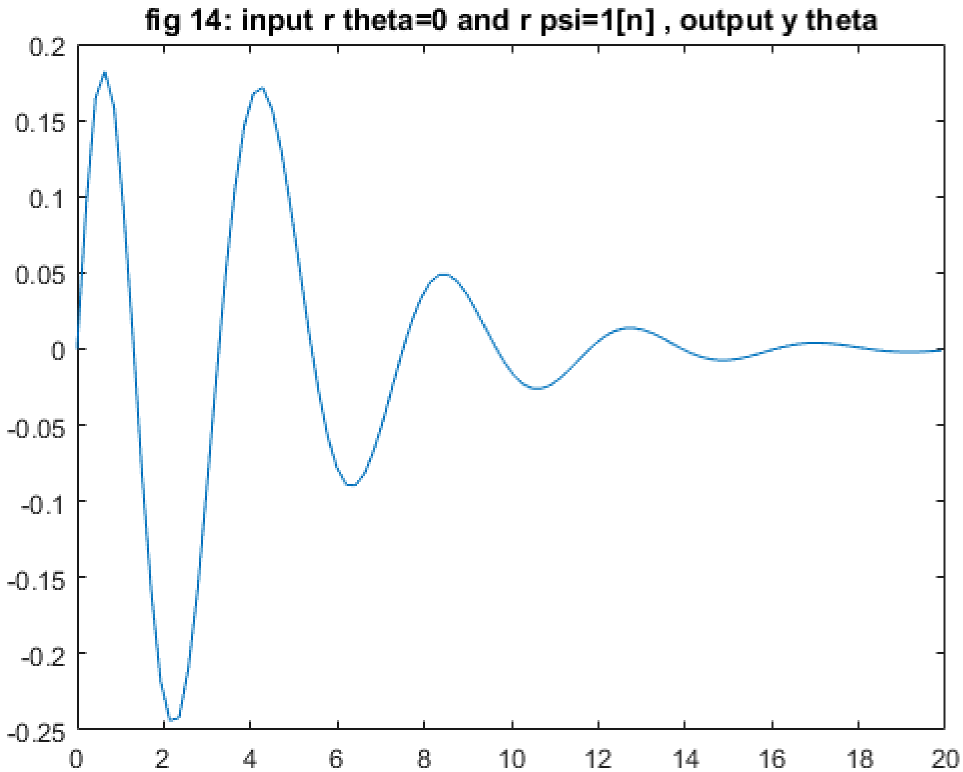
fig 13: input r theta=1[n] and r psi=0 , output y theta



The step info for the above graph is as follows

$$step\_resp\_input\_r\_theta\_output\_r\_theta\ =$$

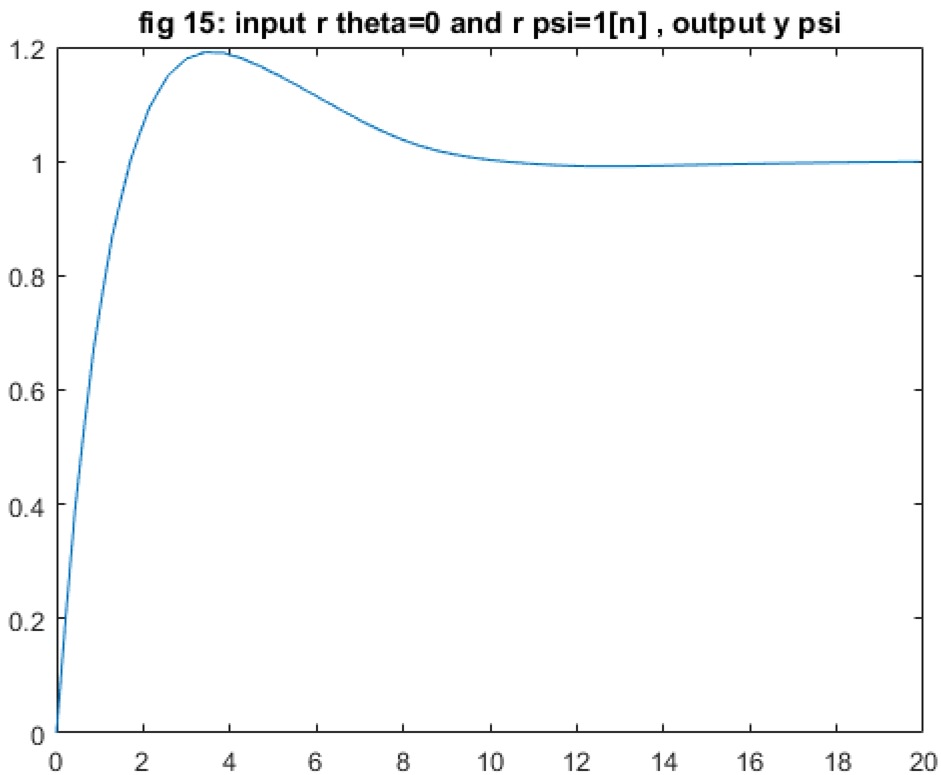
$$RiseTime: 1.7120$$
$$SettlingTime: 10.9140$$
$$SettlingMin: 0.9072$$
$$SettlingMax: 1.1494$$
$$Overshoot: 14.9388$$
$$Undershoot: 0$$
$$Peak: 1.1494$$
$$PeakTime: 3.8520$$



The pitch channel output Θ(z) to a step disturbance from yaw channel with $R_\Theta(z)=0$ is shown in figure 14
The step info for the above graph is as follows

$$step\_resp\_input\_r\_psi\_output\_r\_theta\ =$$

$$RiseTime: 0$$
$$SettlingTime: 15.6220$$
$$SettlingMin: \ -0.2438$$
$$SettlingMax: 0.1714$$
$$Overshoot: 8.5964e+19$$
$$Undershoot: 6.4152e+19$$
$$Peak: 0.2438$$
$$PeakTime: 2.1400$$

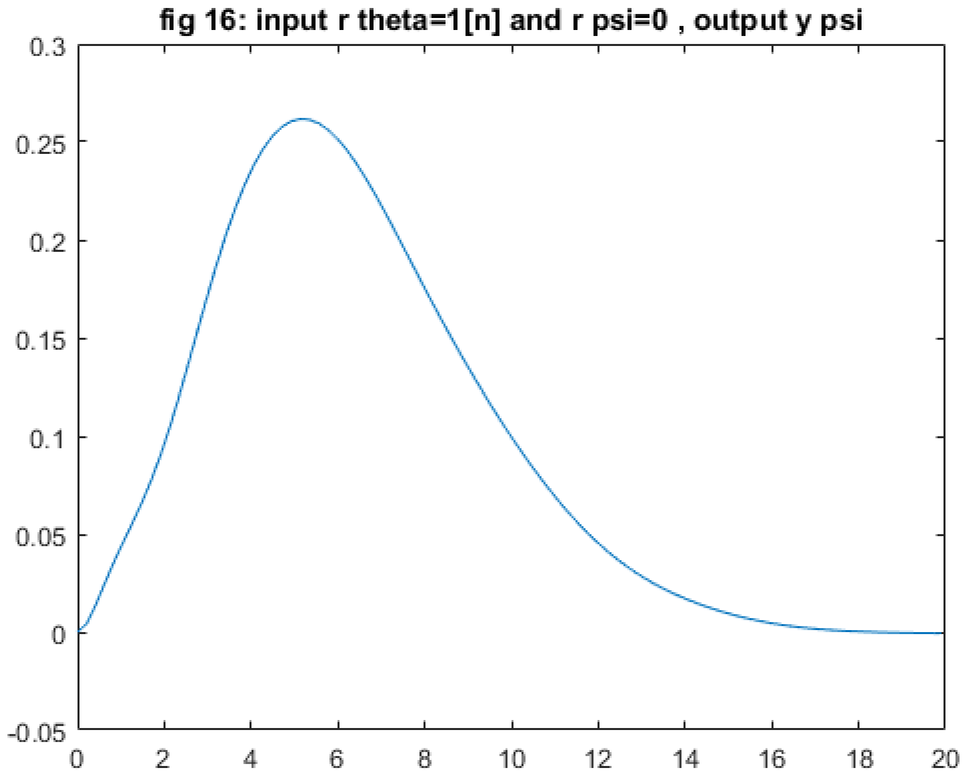


The yaw channel output ψ(z) to a step yaw channel input with D_{Θ}(z)=0 is shown in figure 15

The step info for the above graph is as follows

$$step\_resp\_input\_r\_psi\_output\_r\_psi \ =$$

$$\begin{aligned} RiseTime &: 1.2840 \\ SettlingTime &: 8.7740 \\ SettlingMin &: 0.9385 \\ SettlingMax &: 1.1912 \\ Overshoot &: 19.1152 \\ Undershoot &: 0 \\ Peak &: 1.1912 \\ PeakTime &: 3.4240 \end{aligned}$$



The yaw channel output ψ(z) to a step disturbance from pitch channel with R_ψ(z)=0 is shown in figure 16

The step info for the above graph is as follows

$$step\_resp\_input\_r\_theta\_output\_r\_psi \; =$$

$$\begin{gathered} RiseTime: 0 \\ SettlingTime: 15.8360 \\ SettlingMin: -6.5896e-04 \\ SettlingMax: 8.8147e-06 \\ Overshoot: 2.6860e+14 \\ Undershoot: 1.0668e+17 \\ Peak: 0.2617 \\ PeakTime: 5.1360 \end{gathered}$$

4. Assessing the effects of coupling
   We assess the coupling effects by inspecting the change in transient parameters such as rise time and overshoot. The step responses of the pitch channel before and after coupling are shown below

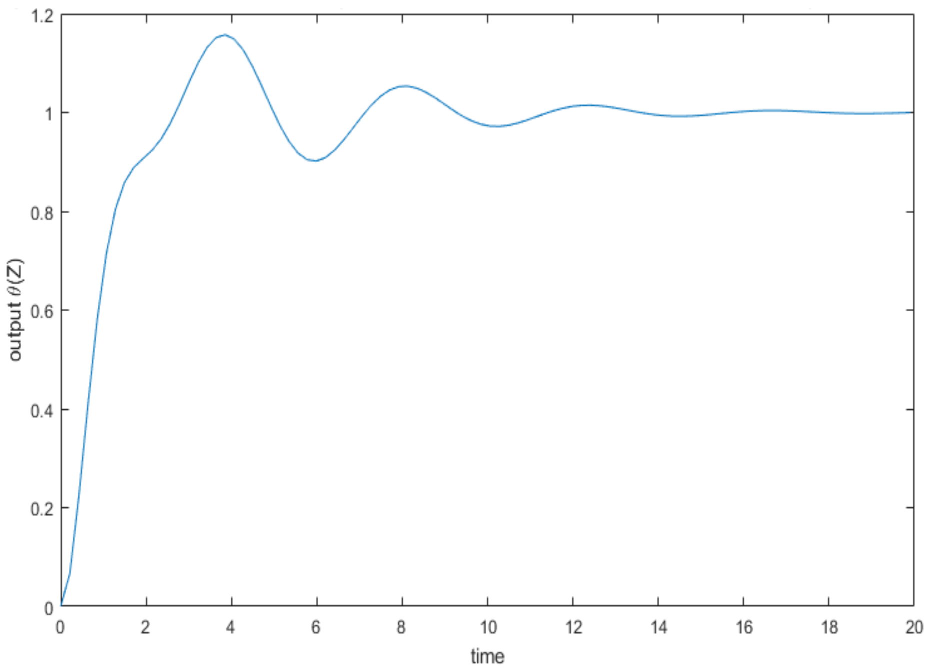


Figure 17: Step response of pitch channel before coupling

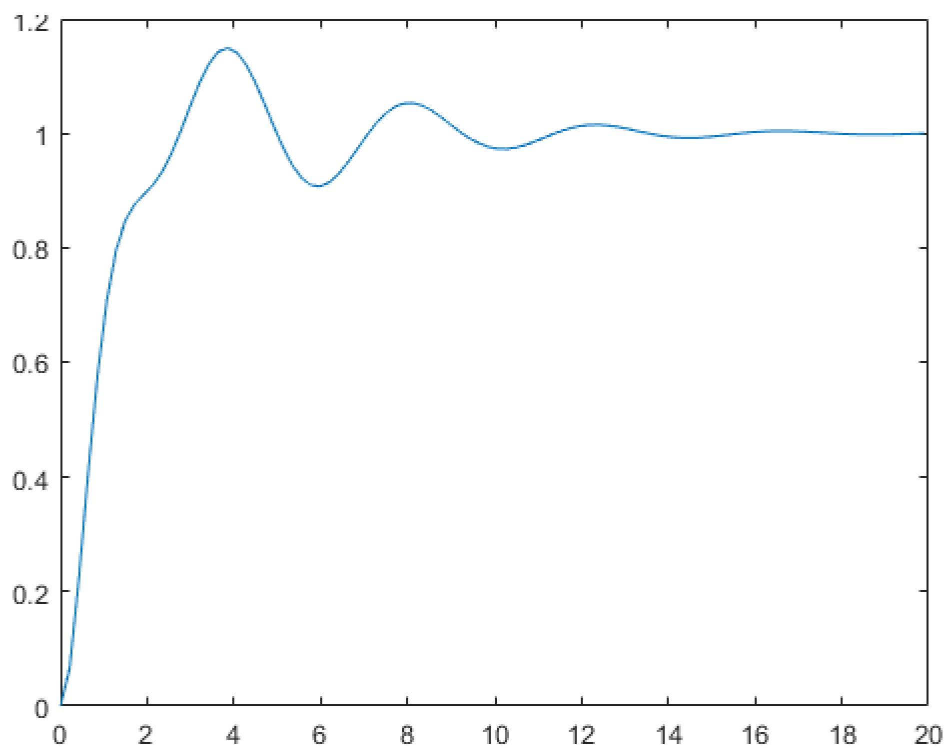


Figure 18: Step response of pitch channel after coupling

From the above two responses it can be observed that due to coupling the rise time has increased from 1.49 before coupling to 1.71 after coupling whereas the overshoot has also remained the same decreasing only by 0.8%. Hence the cross coupling here is acceptable.

The step responses of the yaw channel before and after coupling are shown below in fig 19 & 20.

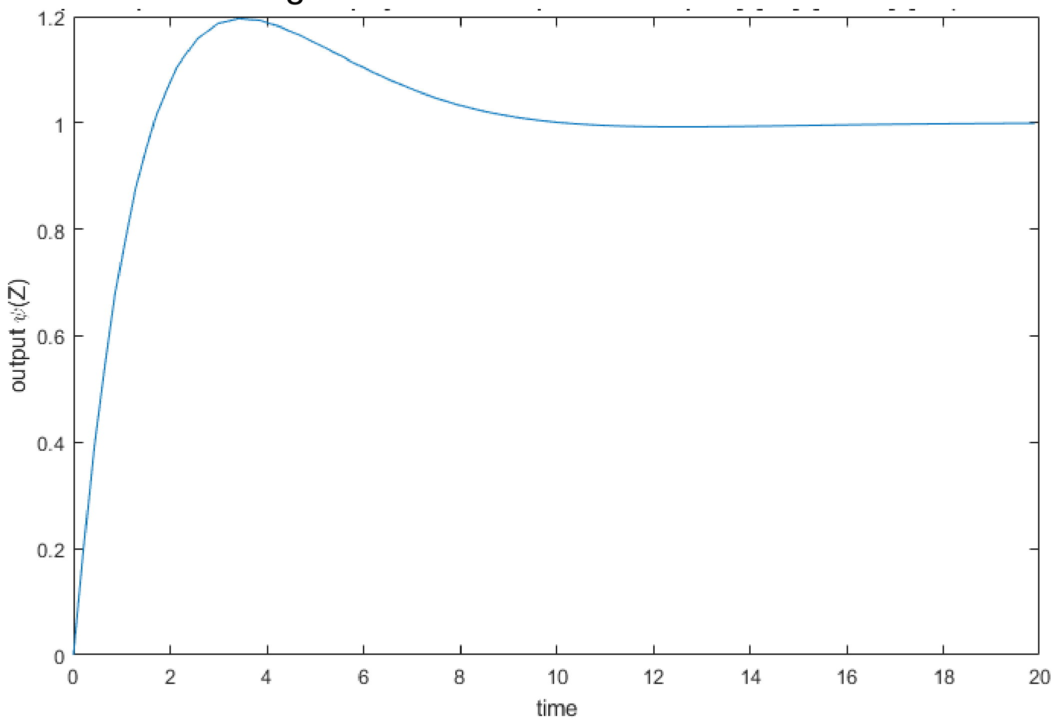


Figure 19: Step response of yaw channel before coupling

From the two responses in fig 19 and 20 it can be observed that even after coupling the rise time has remained the same as well as the overshoot decreasing only by 0.5%. Hence the cross coupling here is acceptable.

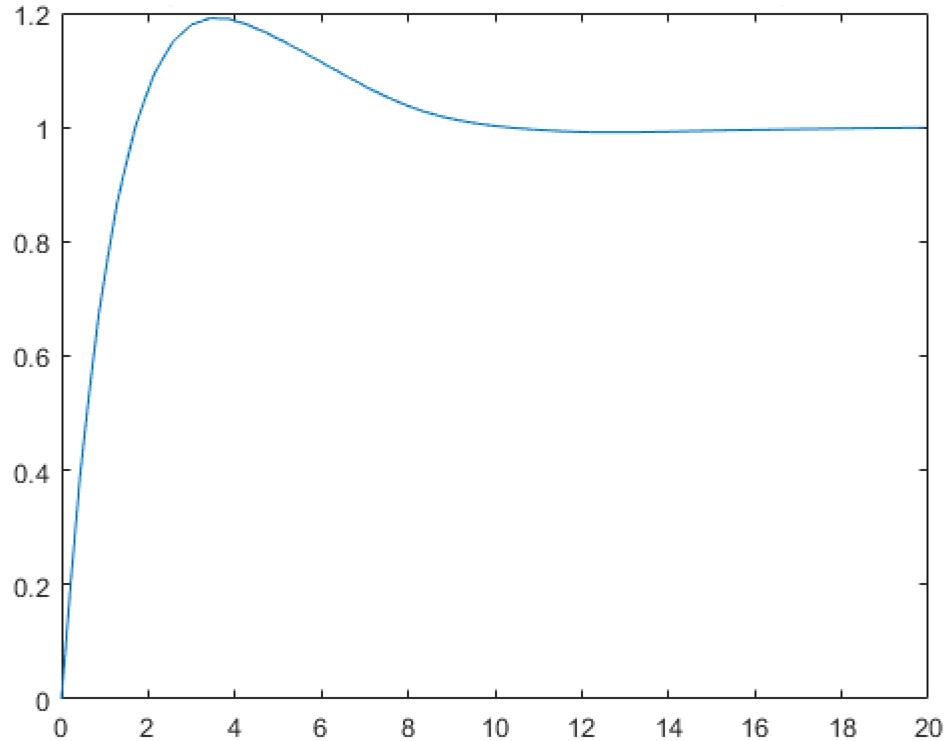


Figure 20: Step response of yaw channel after coupling

5. Conclusion

Hence, In this project we have successfully designed a controller for the yaw channel meeting our specifications, converted the whole system into state space form and simulated it, assessed the effects of cross coupling by comparing the transient parameters before and after coupling.

```
% Real-time computer control systems
% ELEC6061
% Project 2 yaw controller with undesired overshoot spec

%this script contains the design of lead-lag compensator

%21 April 2019

clc
clear all
close all

Ts = 0.214;                                   %sampling period
wnl = 0.9;                                    %natural frequency wn to satisfy rise↙
time spec
wnu = 1.8/(Ts*6);                             %natural frequency wn to satisfy Ts<↙
(Tr/6)

zeta = 0.456;                                 %damping ratio
gs = tf(7.461,[1 0.2701 0])                   %continuous transfer function of pitch↙
channel
                                              %taking yaw coupling as disturbance
gz = c2d(gs,Ts)                               %ZOH discrete equ of gs
p = pole(gz)
z = zero(gz)

kz1 = tf(1.426*[1 -(0.8831)],[1 0.7991],Ts) %Lead compensator
kz = tf([1 -(0.8831)],[1 -1],Ts)          %Lag compensator
kz = series(kz,kz1)                           %Lead-Lag compensator


oltf = series(kz,gz)                          %open loop transfer function
p = pole(oltf)
z = zero(oltf)

cltf = feedback(oltf,1)                       %closed loop transfer function
p = pole(cltf)
z = zero(cltf)

errtf = feedback(1,oltf)                      %error transfer function

figure;
rlocus(gz)                                    %root locus of discrete plant
title('fig 2: root locus of open loop system with K(Z)=1');
zgrid(zeta, [Ts*wnl Ts*wnu])

figure;
rlocus(oltf)                                  %root locus of oltf after compensation
title('fig 3: root locus of open loop system with Lead-Lag compensation (controller with↙
undesired overshoot)');
zgrid(zeta, [Ts*wnl Ts*wnu])
```

```
tfinal = 20;
[yo, to] = step(cltf,tfinal);

figure;
plot(to,yo)                                       %output response to a step reference↙
input r[n]=1[n] and d[n]=0
xlabel('time');
ylabel('output \psi(Z)');
title('fig 4:output response of closed loop system to a step reference input r[n]=1[n]↙
and d[n]=0 (controller with undesired overshoot)');
ref_output_resp=stepinfo(cltf)

controller_cltf = feedback(kz,gz)

tfinal = 20;
[yo, to] = step(controller_cltf,tfinal);

figure;
plot(to,yo)                                       %controller output to a step reference↙
input r[n]=1[n] and d[n]=0
xlabel('time');
ylabel('output vp[n]');
title('fig 4_1:controller output vp[n] of closed loop system to a step reference input r↙
[n]=1[n] and d[n]=0 (controller with undesired overshoot))');
controller_output=stepinfo(controller_cltf)

disturbance_tf = feedback(gz,kz)

tfinal = 20;
[yo, to] = step(disturbance_tf,tfinal);

figure;
plot(to,yo)                                       %output response to a step disturbance↙
input d[n]=1[n] and r[n]=0
xlabel('time');
ylabel('output \psi(Z)');
title('fig 5:output response of closed loop system to a step disturbance input d[n]=1[n]↙
and r[n]=0 (controller with undesired overshoot)');
disturbance_output_resp=stepinfo(disturbance_tf)
```

```
% Real-time computer control systems
% ELEC6061
% Project 2 yaw controller undesired disturbance spec

%this script contains the design of lead-lag compensator

%21 April 2019

clc
clear all
close all

Ts = 0.214;                                    %sampling period
wnl = 0.9;                                     %natural frequency wn to satisfy rise↙
time spec
wnu = 1.8/(Ts*6);                              %natural frequency wn to satisfy Ts<↙
(Tr/6)

zeta = 0.456;                                  %damping ratio
gs = tf(7.461,[1 0.2701 0])                    %continuous transfer function of pitch↙
channel
                                               %taking yaw coupling as disturbance
gz = c2d(gs,Ts)                                %ZOH discrete equ of gs
p = pole(gz)
z = zero(gz)

kz1 = tf(1.2109*[1 -(0.9382)],[1 0.92],Ts) %Lead compensator
kz = tf([1 -(0.9382)],[1 -1],Ts)           %Lag compensator
kz = series(kz,kz1)                            %Lead-Lag compensator


oltf = series(kz,gz)                           %open loop transfer function
p = pole(oltf)
z = zero(oltf)

cltf = feedback(oltf,1)                        %closed loop transfer function
p = pole(cltf)
z = zero(cltf)

errtf = feedback(1,oltf)                       %error transfer function

figure;
rlocus(gz)                                     %root locus of discrete plant
title('fig 2: root locus of open loop system with K(Z)=1');
zgrid(zeta, [Ts*wnl Ts*wnu])

figure;
rlocus(oltf)                                   %root locus of oltf after compensation
title('fig 6: root locus of open loop system with Lead-Lag compensation (controller with↙
undesired disturbance)');
zgrid(zeta, [Ts*wnl Ts*wnu])
```

```
tfinal = 20;
[yo, to] = step(cltf,tfinal);

figure;
plot(to,yo)                                        %output response to a step reference
input r[n]=1[n] and d[n]=0
xlabel('time');
ylabel('output \psi(Z)');
title('fig 7:output response of closed loop system to a step reference input r[n]=1[n]
and d[n]=0 (controller with undesired disturbance)');
ref_output_resp=stepinfo(cltf)

controller_cltf = feedback(kz,gz)

tfinal = 20;
[yo, to] = step(controller_cltf,tfinal);

figure;
plot(to,yo)                                        %controller output to a step reference
input r[n]=1[n] and d[n]=0
xlabel('time');
ylabel('output vp[n]');
title('fig 7_1:controller output vp[n] of closed loop system to a step reference input r
[n]=1[n] and d[n]=0 (controller with undesired disturbance)');
controller_output=stepinfo(controller_cltf)

disturbance_tf = feedback(gz,kz)

tfinal = 20;
[yo, to] = step(disturbance_tf,tfinal);

figure;
plot(to,yo)                                        %output response to a step disturbance
input d[n]=1[n] and r[n]=0
xlabel('time');
ylabel('output \psi(Z)');
title('fig 8:output response of closed loop system to a step disturbance input d[n]=1[n]
and r[n]=0 (controller with undesired disturbance)');
disturbance_output_resp=stepinfo(disturbance_tf)
```

```
% Real-time computer control systems
% ELEC6061
% Project 2 yaw controller desired

%this script contains the design of lead-lag compensator

%21 April 2019

clc
clear all
close all

Ts = 0.214;                                    %sampling period
wnl = 0.9;                                     %natural frequency wn to satisfy rise↙
time spec
wnu = 1.8/(Ts*6);                              %natural frequency wn to satisfy Ts<↙
(Tr/6)

zeta = 0.456;                                  %damping ratio
gs = tf(7.461,[1 0.2701 0])                    %continuous transfer function of pitch↙
channel
                                               %taking yaw coupling as disturbance
gz = c2d(gs,Ts)                                %ZOH discrete equ of gs
p = pole(gz)
z = zero(gz)

kz1 = tf(1.2109*[1 -(0.9382-0.025*1i)],[1 0.92],Ts) %Lead compensator
kz = tf([1 -(0.9382+0.025*1i)],[1 -1],Ts)          %Lag compensator
kz = series(kz,kz1)                            %Lead-Lag compensator


oltf = series(kz,gz)                           %open loop transfer function
p = pole(oltf)
z = zero(oltf)

cltf = feedback(oltf,1)                        %closed loop transfer function
p = pole(cltf)
z = zero(cltf)

errtf = feedback(1,oltf)                       %error transfer function

figure;
rlocus(gz)                                     %root locus of discrete plant
title('fig 2: root locus of open loop system with K(Z)=1');
zgrid(zeta, [Ts*wnl Ts*wnu])

figure;
rlocus(oltf)                                   %root locus of oltf after compensation
title('fig 9: root locus of open loop system with Lead-Lag compensation (desired↙
controller)');
zgrid(zeta, [Ts*wnl Ts*wnu])
```

```
tfinal = 20;
[yo, to] = step(cltf,tfinal);

figure;
plot(to,yo)                                            %output response to a step reference↙
input r[n]=1[n] and d[n]=0
xlabel('time');
ylabel('output \psi(Z)');
title('fig 10:output response of closed loop system to a step reference input r[n]=1[n]↙
and d[n]=0 (desired controller)');
ref_output_resp=stepinfo(cltf)

controller_cltf = feedback(kz,gz)

tfinal = 20;
[yo, to] = step(controller_cltf,tfinal);

figure;
plot(to,yo)                                            %controller output to a step reference↙
input r[n]=1[n] and d[n]=0
xlabel('time');
ylabel('output vp[n]');
title('fig 10_1:controller output vp[n] of closed loop system to a step reference input r↙
[n]=1[n] and d[n]=0 (desired controller)');
controller_output=stepinfo(controller_cltf)

disturbance_tf = feedback(gz,kz)

tfinal = 20;
[yo, to] = step(disturbance_tf,tfinal);

figure;
plot(to,yo)                                            %output response to a step disturbance↙
input d[n]=1[n] and r[n]=0
xlabel('time');
ylabel('output \psi(Z)');
title('fig 11:output response of closed loop system to a step disturbance input d[n]=1[n]↙
and r[n]=0 (desired controller)');
disturbance_output_resp=stepinfo(disturbance_tf)
```

```
% Real-time computer control systems
% ELEC6061
% Project 2 pitch and yaw channels combined

%21 April 2019

clc
clear all
close all

Ts = 0.214;                                       %sampling period
wnl = 0.9;                                        %natural frequency wn to satisfy rise↙
time spec
wnu = 1.8/(Ts*6);                                 %natural frequency wn to satisfy Ts<↙
(Tr/6)

zeta = 0.456;                                     %damping ratio
gs_theta = tf(37.2021,[1 0.2830 2.7452])          %continuous transfer function of pitch↙
channel
                                                  %taking yaw coupling as disturbance
gz_theta = c2d(gs_theta,Ts)                       %ZOH discrete equ of gs_theta
p = pole(gz_theta)
z = zero(gz_theta)

kz = tf(0.08*[1 -(0.943 + 0.315i)],[1 -0.55],Ts)%Lead compensator
kz1 = tf([1 -(0.943 - 0.315i)],[1 -1],Ts)         %Lag compensator
kz_theta = series(kz,kz1)                         %Lead-Lag compensator

oltf = series(kz_theta,gz_theta)                  %open loop transfer function
p = pole(oltf)
z = zero(oltf)

cltf = feedback(oltf,1)                           %closed loop transfer function
p = pole(cltf)
z = zero(cltf)

errtf = feedback(1,oltf)                          %error transfer function

figure;
rlocus(gz_theta)                                  %root locus of discrete plant
title('fig 1 project1: root locus of open loop system with K(Z)=1');
zgrid(zeta, [Ts*wnl Ts*wnu])

figure;
rlocus(oltf)                                      %root locus of oltf after compensation
title('fig 8 project1: root locus of open loop system with Lead-Lag compensation (desired↙
controller)');
zgrid(zeta, [Ts*wnl Ts*wnu])

tfinal = 20;
[yo, to] = step(cltf,tfinal);
```

```
figure;
plot(to,yo)                                    %output response to a step reference↙
input r[n]=1[n] and d[n]=0
xlabel('time');
ylabel('output \theta(Z)');
title('fig 9 project1:output response of closed loop system to a step reference input r↙
[n]=1[n] and d[n]=0 (desired controller)');
ref_output_resp=stepinfo(cltf)

controller_cltf = feedback(kz_theta,gz_theta)

tfinal = 20;
[yo, to] = step(controller_cltf,tfinal);

figure;
plot(to,yo)                                    %controller output to a step reference↙
input r[n]=1[n] and d[n]=0
xlabel('time');
ylabel('output vp[n]');
title('fig 10 project1:controller output vp[n] of closed loop system to a step reference↙
input r[n]=1[n] and d[n]=0 (desired controller)');
controller_output=stepinfo(controller_cltf)

disturbance_tf = feedback(gz_theta,kz_theta)

tfinal = 20;
[yo, to] = step(disturbance_tf,tfinal);

figure;
plot(to,yo)                                    %output response to a step disturbance↙
input d[n]=1[n] and r[n]=0
xlabel('time');
ylabel('output \theta(Z)');
title('fig 11 project1:output response of closed loop system to a step disturbance input↙
d[n]=1[n] and r[n]=0 (desired controller)');
disturbance_output_resp=stepinfo(disturbance_tf)

Ts = 0.214;                                    %sampling period
wnl = 0.9;                                     %natural frequency wn to satisfy rise↙
time spec
wnu = 1.8/(Ts*6);                              %natural frequency wn to satisfy Ts<↙
(Tr/6)

zeta = 0.456;                                  %damping ratio
gs = tf(7.461,[1 0.2701 0])                    %continuous transfer function of pitch↙
channel
                                               %taking yaw coupling as disturbance
gz = c2d(gs,Ts)                                %ZOH discrete equ of gs
p = pole(gz)
z = zero(gz)
```

```
kz1 = tf(1.2109*[1 -(0.9382-0.025*1i)],[1 0.92],Ts) %Lead compensator
kz = tf([1 -(0.9382+0.025*1i)],[1 -1],Ts)            %Lag compensator
kz = series(kz,kz1)                                 %Lead-Lag compensator
kz_psi=kz;

oltf = series(kz,gz)                                %open loop transfer function
p = pole(oltf)
z = zero(oltf)

cltf = feedback(oltf,1)                             %closed loop transfer function
p = pole(cltf)
z = zero(cltf)

errtf = feedback(1,oltf)                            %error transfer function

figure;
rlocus(gz)                                          %root locus of discrete plant
title('fig 2: root locus of open loop system with K(Z)=1');
zgrid(zeta, [Ts*wnl Ts*wnu])

figure;
rlocus(oltf)                                        %root locus of oltf after compensation
title('fig 9: root locus of open loop system with Lead-Lag compensation (desired↙
controller)');
zgrid(zeta, [Ts*wnl Ts*wnu])

tfinal = 20;
[yo, to] = step(cltf,tfinal);

figure;
plot(to,yo)                                         %output response to a step reference↙
input r[n]=1[n] and d[n]=0
xlabel('time');
ylabel('output \psi(Z)');
title('fig 10:output response of closed loop system to a step reference input r[n]=1[n]↙
and d[n]=0 (desired controller)');
ref_output_resp=stepinfo(cltf)

controller_cltf = feedback(kz,gz)

tfinal = 20;
[yo, to] = step(controller_cltf,tfinal);

figure;
plot(to,yo)                                         %controller output to a step reference↙
input r[n]=1[n] and d[n]=0
xlabel('time');
ylabel('output vp[n]');
title('fig 10_1:controller output vp[n] of closed loop system to a step reference input r↙
[n]=1[n] and d[n]=0 (desired controller)');
```

```
controller_output=stepinfo(controller_cltf)

disturbance_tf = feedback(gz,kz)

tfinal = 20;
[yo, to] = step(disturbance_tf,tfinal);

figure;
plot(to,yo)                              %output response to a step disturbance↙
input d[n]=1[n] and r[n]=0
xlabel('time');
ylabel('output \psi(Z)');
title('fig 11:output response of closed loop system to a step disturbance input d[n]=1[n]↙
and r[n]=0 (desired controller)');
disturbance_output_resp=stepinfo(disturbance_tf)

kz_theta_ss=ss(kz_theta)                 %state space form of pitch controller
kz_psi_ss=ss(kz_psi)                     %state space form of yaw controller

kz_combined=append(kz_theta_ss,kz_psi_ss)       %state space form of combined controller

a_ss=[0,1,0,0;-2.7451,-0.2829,0,0;0,0,0,1;0,0,0,-0.2701];
b_ss=[0,0;37.2021,3.5306;0,0;2.3892,7.461];
c_ss=[1,0,0,0;0,0,1,0];
d_ss=0;

gs_combined=ss(a_ss,b_ss,c_ss,d_ss)
gz_combined=c2d(gs_combined,Ts)          %ZOH equivalent of the plant

oltf_combined=series(kz_combined,gz_combined)
cltf_combined=feedback(oltf_combined,[1,0;0,1]) %closed loop transfer function of the↙
MIMO system

tfinal=20;
[yz,tz]=step(cltf_combined,tfinal);
final_output=stepinfo(cltf_combined);
step_resp_input_r_theta_output_r_theta=final_output(1,1)
step_resp_input_r_theta_output_r_psi=final_output(2,1)
step_resp_input_r_psi_output_r_theta=final_output(1,2)
step_resp_input_r_psi_output_r_psi=final_output(2,2)
figure;
plot(tz,yz(:,1,1));                      %step response with r_theta=1[n] and↙
r_psi=0, output y_theta
title('fig 13: input r theta=1[n] and r psi=0 , output y theta');

figure;
plot(tz,yz(:,2,1));                      %step response with r_theta=1[n] and↙
r_psi=0, output y_psi (disturbance for yaw channel)
title('fig 16: input r theta=1[n] and r psi=0 , output y psi');

figure;
```

```
plot(tz,yz(:,1,2));                                     %step response with r_theta=0 and r_psi=1
[n] , output y_theta (disturbance for pitch channel)
title('fig 14: input r theta=0 and r psi=1[n] , output y theta');

figure;
plot(tz,yz(:,2,2));                                     %step response with r_theta=0 and r_psi=1
[n] , output y_psi
title('fig 15: input r theta=0 and r psi=1[n] , output y psi');
```