A picture containing fireworks, dark, water, flying

Description automatically generated

College of Engineering

School of Aeronautics and Astronautics

AAE 421

Flight Dynamics and Controls

EXAM 2

*Author:*

Tomoki Koike

*Supervisor:*

Ran Dai

November 20th, 2020

Purdue University

West Lafayette, Indiana

I certify that I have neither given help to, nor received help from, any individual in matters relating to this examination.

Diagram

Description automatically generated

Signature:

Problem 1 (25 pts)

Given the following set of nonlinear differential equations:

Where

The parameters are set as

1. Find the trim condition for a leveled flight at altitude of flying at a speed of , where the state and the control is denoted by . (Hint: to maintain a trim condition at level flight, we must have and )

For the trim conditions we solve the following two equations by plugging in all the given parameters as well as trim conditions.

That gives the following equations

Solving the second equation gives us

Text, letter

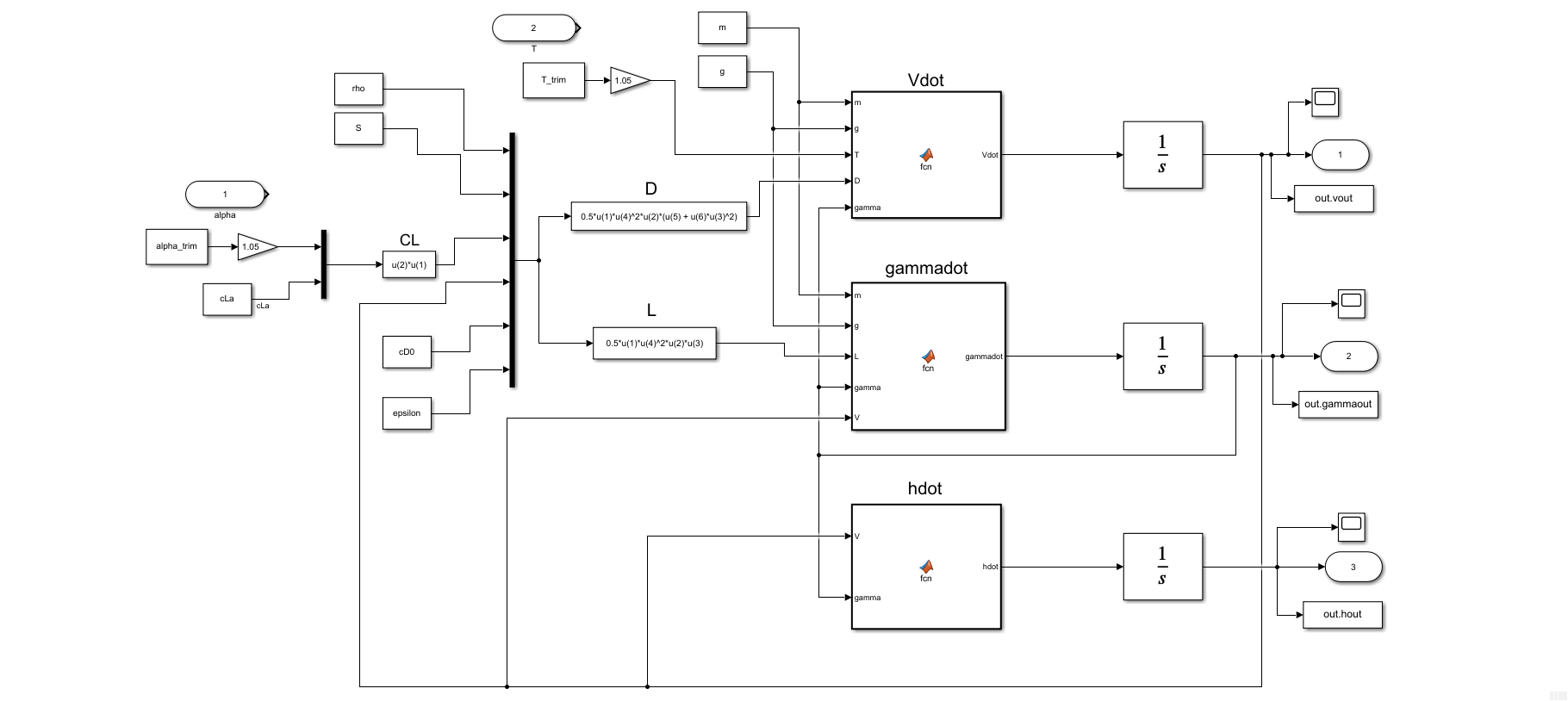
Description automatically generated

And plugging this into equation (1) gives the trim value of *T*

Also,

Develop a Simulink model to simulate the system state response for t = 10 sec with the initial condition set as the trim condition in Part (a) and input .

The Simulink model is the following



The Embedded MATLAB Blocks have the following functions defined

function Vdot = fcn(m, g, T, D, gamma)

Vdot = (T - D - m \* g \* sin(gamma)) / m;

end

function gammadot = fcn(m, g, L, gamma, V)

gammadot = (L - m \* g \* cos(gamma)) / m / V;

end

function hdot = fcn(V, gamma)

hdot = V \* sin(gamma);

end

The Simulation results is the following

Chart

Description automatically generated

Running the following commands give us the simulation

% Problem 1

% Parameters

epsilon = 0.9;

AR = 0.86;

m = 0.003;

S = 0.017;

cD0 = 0.02;

rho = 0.41405;

h = 10000;

cLa = 1.2936;

g = 9.81;

% (a)

% Equilibrium conditions

he = h;

Ve = 3.7;

gamma = 0;

% Find trim conditions

syms alpha T

cL = cLa \* alpha;

L = 0.5\*rho\*Ve^2\*S\*cL;

D = 0.5\*rho\*Ve^2\*S\*(cD0 + epsilon\*cL^2);

eqn1 = 0 == (T - D - m\*g\*sin(gamma)) / m;

eqn2 = 0 == (L - m\*g\*cos(gamma)) / m / Ve;

res = solve([eqn1, eqn2], [alpha, T]);

% (b)

alpha\_trim = double(res.alpha)

T\_trim = double(res.T)

simout = sim("e2\_p1\_model.slx");

t = simout.tout;

Vsim = simout.vout.signals.values;

gammasim = simout.gammaout.signals.values;

hsim = simout.hout.signals.values;

% Plotting

fig1 = figure('Renderer',"painters", 'Position', [10 10 900 1000]);

subplot(3,1,1)

plot(t,Vsim)

grid on; grid minor; box on;

ylabel('V [m/s]')

subplot(3,1,2)

plot(t,gammasim)

grid on; grid minor; box on;

ylabel('$\gamma$ [rad]')

subplot(3,1,3)

plot(t,hsim)

grid on; grid minor; box on;

ylabel('h [m]')

xlabel('time, [sec]')

sgtitle('Problem 1 (b) $0\leq t \leq 10$ - T. Koike')

saveas(fig1, fullfile(fdir, "p1\_2.png"));

1. Make use of the MATLAB command ‘linmod.m’ to find the linearized state space model about the trim condition found in part (a), assuming the output .

The following command gives us the A, B, C, and D matrices

[A, B, C, D] = linmod('e2\_p1\_model',[Ve;gamma;he],[alpha\_trim; T\_trim])

**Problem 2 (25 pts)**

Using Dutch roll approximation, determine the state feedback gains so that the damping ratio and frequency of the Dutch roll are 0.3 and 1.0 rad/s, respectively. Assume the airplane has the following characteristics:

Text, letter

Description automatically generated

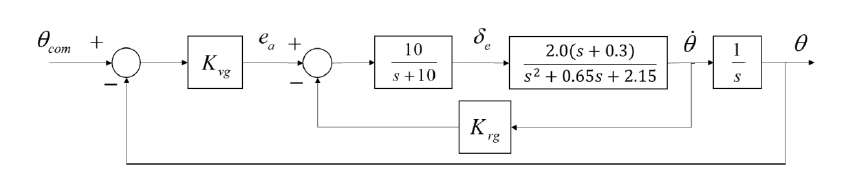
Text, letter

Description automatically generated

If

**Problem 3 (25 pts)**

For the pitch displacement autopilot system shown below



1. Determine the gain necessary to improve the system characteristic so that the control system has the following performance: and rad/s. Verify your solution by providing the root locus plot of the overall system and plot of system response to a step change in the commanded pitch attitude.

Use the Control Systems Designer to tune the gyro gains for the given requirements

num = conv([0, 10], 2.0\*[1, 0.3]);

den = conv([1, 10], [1, 0.65, 2.15]);

sys = tf(num, den);

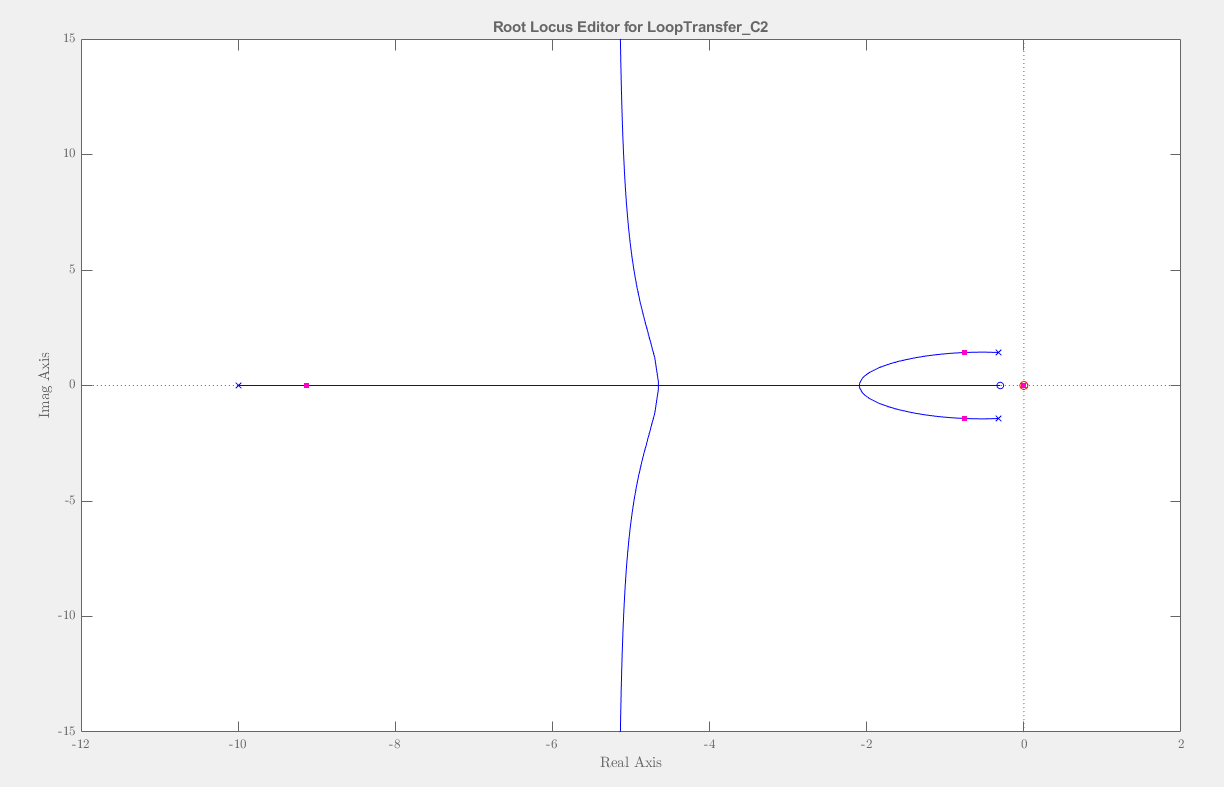
s = tf("s");

H = 1\*s;

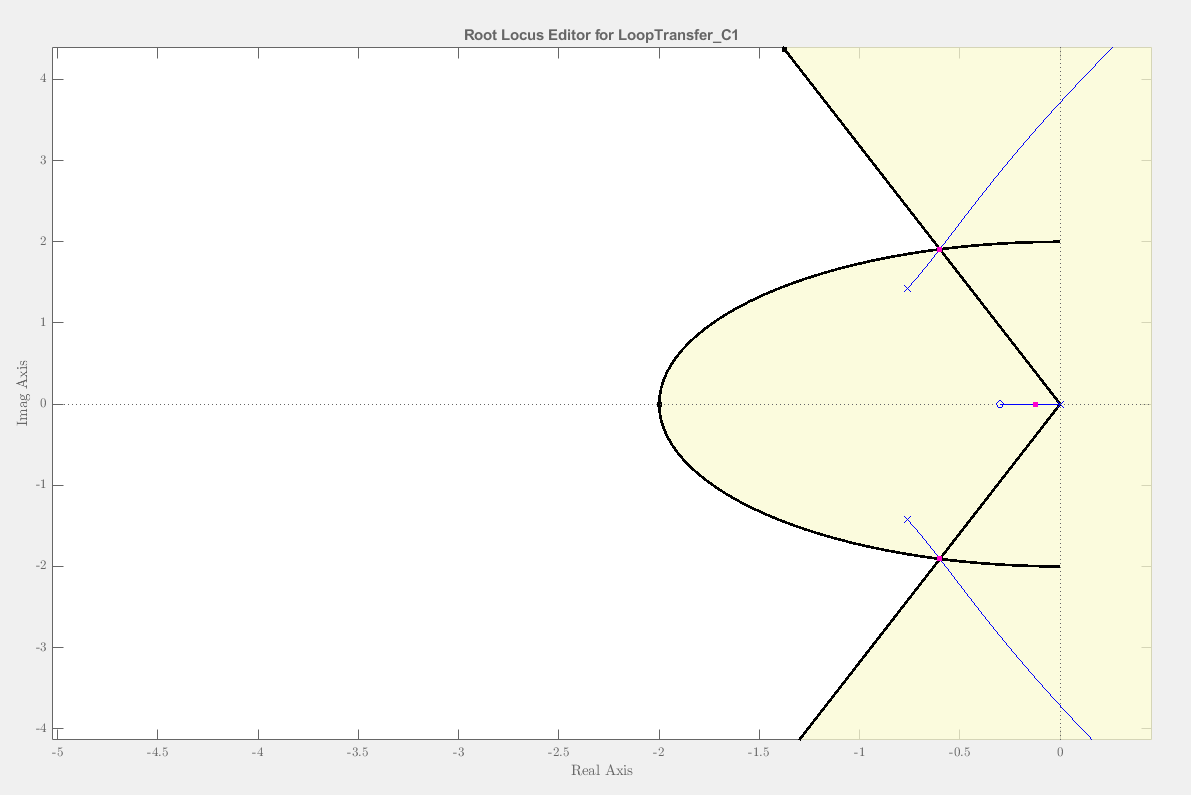
controlSystemDesigner(sys);

The control system tuner gives us the following gains

The Root locus of the inner loop system becomes



The Root Locus of the outer loop system becomes



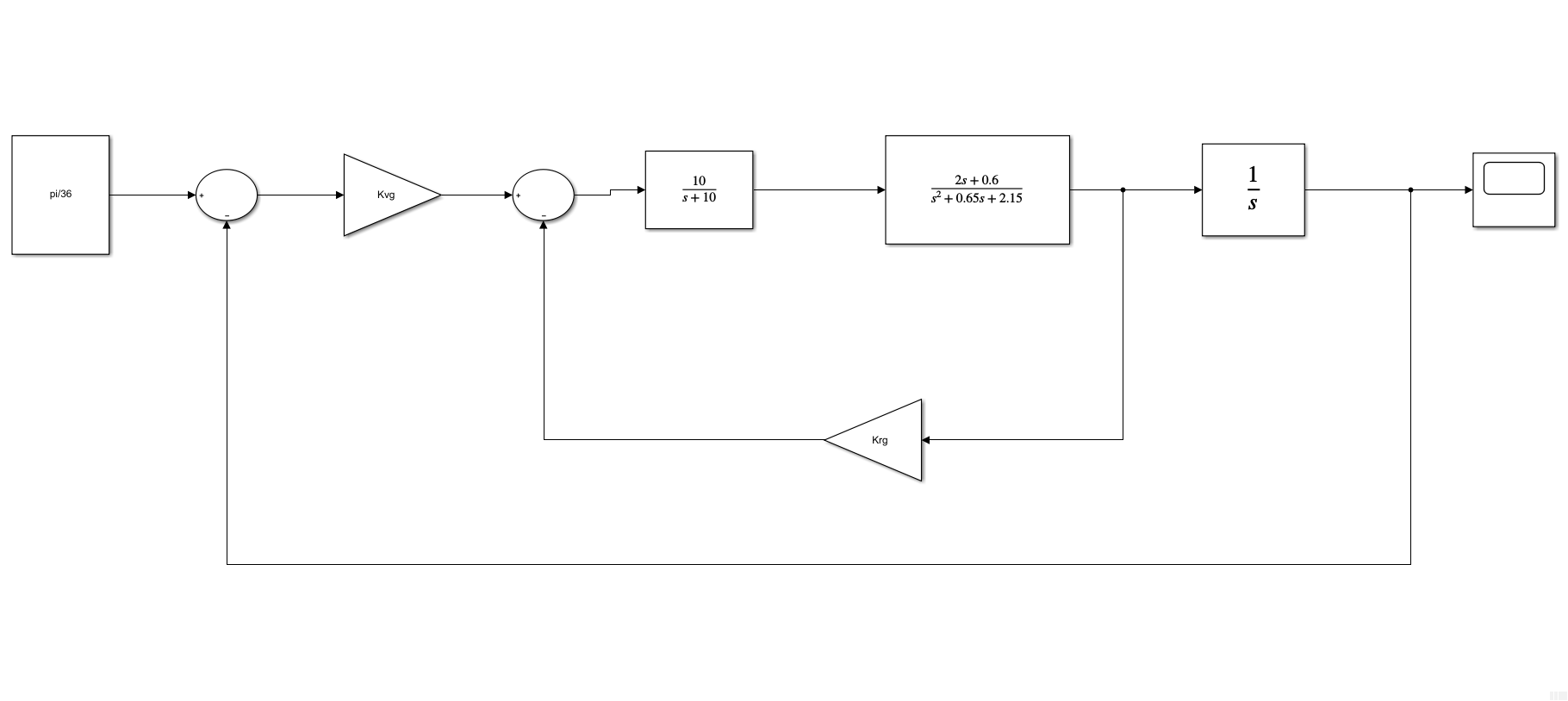
Then we get the response for 5-degree pitch attitude change

num = conv([0, 10], 2.0\*[1, 0.3]);

den = conv([1, 10], [1, 0.65, 2.15]);

Krg = 0.392;

Kvg = 0.76561;

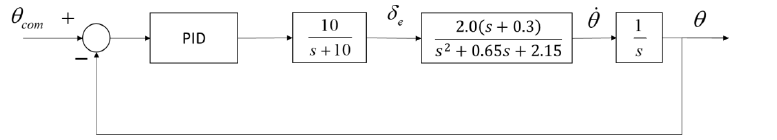


The response to 5-degree () change in pitch is

A picture containing chart

Description automatically generated

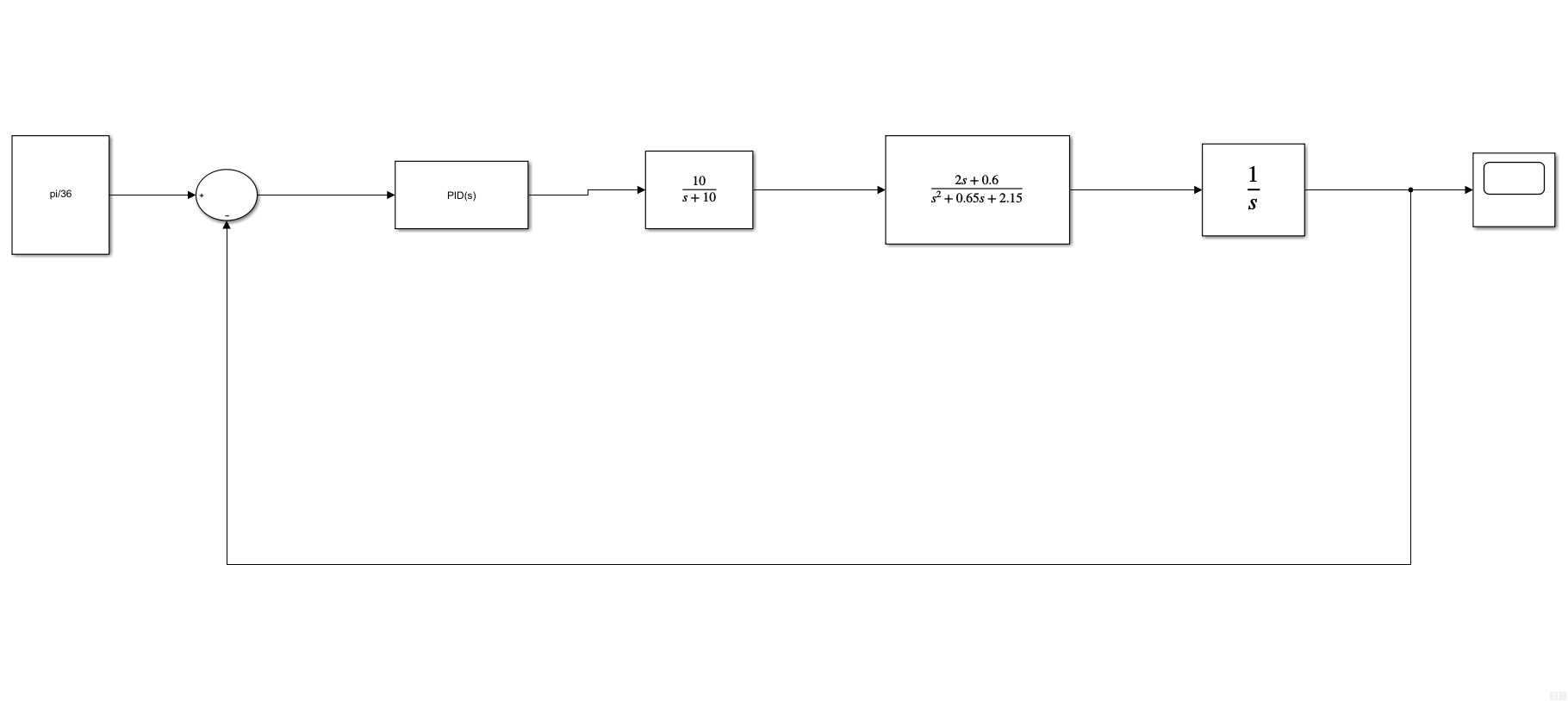
1. Replace the rate gyro and amplifier with a PID controller shown below:



Design the PID gains using MATLAB Control System Tuner. Compare the design results with Part (a) by providing the plot of system response to a 5-degree step change in the commanded pitch altitude.

The requirements are

The model is as follows.



The results are

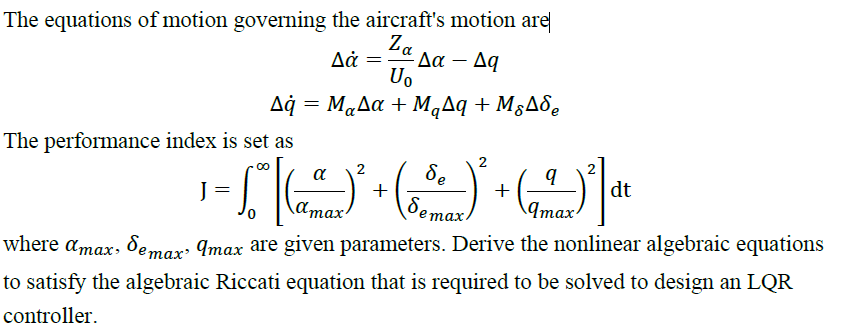
The response to a 5-degree step change is as follows.

Chart

Description automatically generated

The PID tuned results seem to be more stable than the one tuned by 2 proportional controllers inside feedback loops.

**Problem 4 (25 pts)**



A picture containing text

Description automatically generated

Diagram

Description automatically generated

Text, letter

Description automatically generated

**Bonus (5 pts)**

Use state feedback to design an altitude hold control system. Assume the forward speed is held constant and the longitudinal equation can be modeled using the short-period approximation. The short-period equations are

Assume the where . Determine the state feedback gain if the closed-loop eigenvalues are located at

The state space equation can be remodeled by including the altitude change, which becomes as follows.

First, we have to find out if this system is controllable or not.

% Setup

A = [-1.5, 1, 0, 0; -4, -1, 0, 0; 0, 1, 0, 0; -200, 0, 200, 0];

B = [-0.2; -8; 0; 0];

% Check controllability

Qc = ctrb(A, B);

Qc\_rref = rref(Qc);

|  |  |
| --- | --- |
| Qc = 4×4  103 ×  -0.0002 -0.0077 0.0204 -0.0085  -0.0080 0.0088 0.0220 -0.1034  0 -0.0080 0.0088 0.0220  0 0.0400 -0.0600 -2.3100 | Qc\_rref = 4×4  1 0 0 0  0 1 0 0  0 0 1 0  0 0 0 1 |

The controllability matrix has full rank, and therefore, this system is controllable.

Now we find the controller gains using the Brogan’s Algorithm

Step 1:

Find

Step 2:

Compute

Step 3:

Calculate

Where correspond to the columns of .

Step 4:

Find the gains with

Where

Thus,

And

Thus, our gains are correct.

% Brogan's Algorithm with desired poles

lambda = [-1.5+2.5j, -1.5-2.5j, -0.75+1j, -0.75-1j];

syms x

[n, m] = size(B);

phi = inv(x\*eye(n)-A);

psi = phi \* B;

psibar = ([subs(psi(:), lambda(1)), subs(psi(:), lambda(2)),...

subs(psi(:), lambda(3)), subs(psi(:), lambda(4))]);

K = -[1, 1, 1, 1]\*inv(psibar);

Acl = A - B\*K

eig(Acl)