EEE 444 Robust Feedback Theory
Oğuz Altan – 21600966
25.02.2020

EEE 444 Robust Feedback Theory HW1 Report

Oğuz Altan – 21600966

Part A)

We are to find the maximum value of δ such that there exists a robustly stabilizing controller, and determine the corresponding optimal value of K. We are given intervals for coefficients. I constructed 4 Kharitonov polynomials and used 16-plant theorem to find the wanted values. I used 100,001 values of K values, starting from 0 and ending at 100, with step size of 0.001. For each of these K values, I checked stability of constructed polynoms, checking whether all of the roots of polynomes are on the left hand side of the complex plane, i.e $\mathbb{C}^-$. The 16-plant theory suggests if the 16 polynoms are all stable, then the system is stable. For a fixed value of δ, I looked for the number of K values suitable for stability. As we are to find the optimal value of K, we need to find a value of δ so that there is only one K value. Using iteratative algorithms, experimentally, I found

$$\delta = 3.151626793$$

and the corresponding optimal value of K is

$$K = 30.6190$$

Part B)

We are to draw the root locus of this system with respect to K, using the optimal value of δ found in the first part. Given the plant, to draw the root locus of the system, we first try to find the transfer function F, using the equation:

$$X(s) = \ 1 + F(s)K = N_0N_c + D_0Dc$$

where N_0, N_c, D_0 and D_c are defined as

$$P_0 = \frac{N_0}{D_0} \qquad C = \frac{N_c}{D_C}$$

We obtain the characteristic polynomial X(s) as

$$X(s) = 1 + \frac{K(s + 0.35)}{s^7 + 10s^6 + 60s^5 + 140s^4 + 200s^3 + 121s^2 + (-\frac{\delta}{2} - 1.4)}$$

for the max value of δ found in part A, and we use this polynomial to find closed loop tranfer function of the system $F(s)$, which is

$$F(s) = \frac{s + 0.35}{s^7 + 10s^6 + 60s^5 + 140s^4 + 200s^3 + 121s^2 + (-\frac{\delta}{2} - 1.4)}$$

Also, the location of the closed loop system poles for the optimal value of K, determined above are given below, in the root locus diagram corresponding to our system:

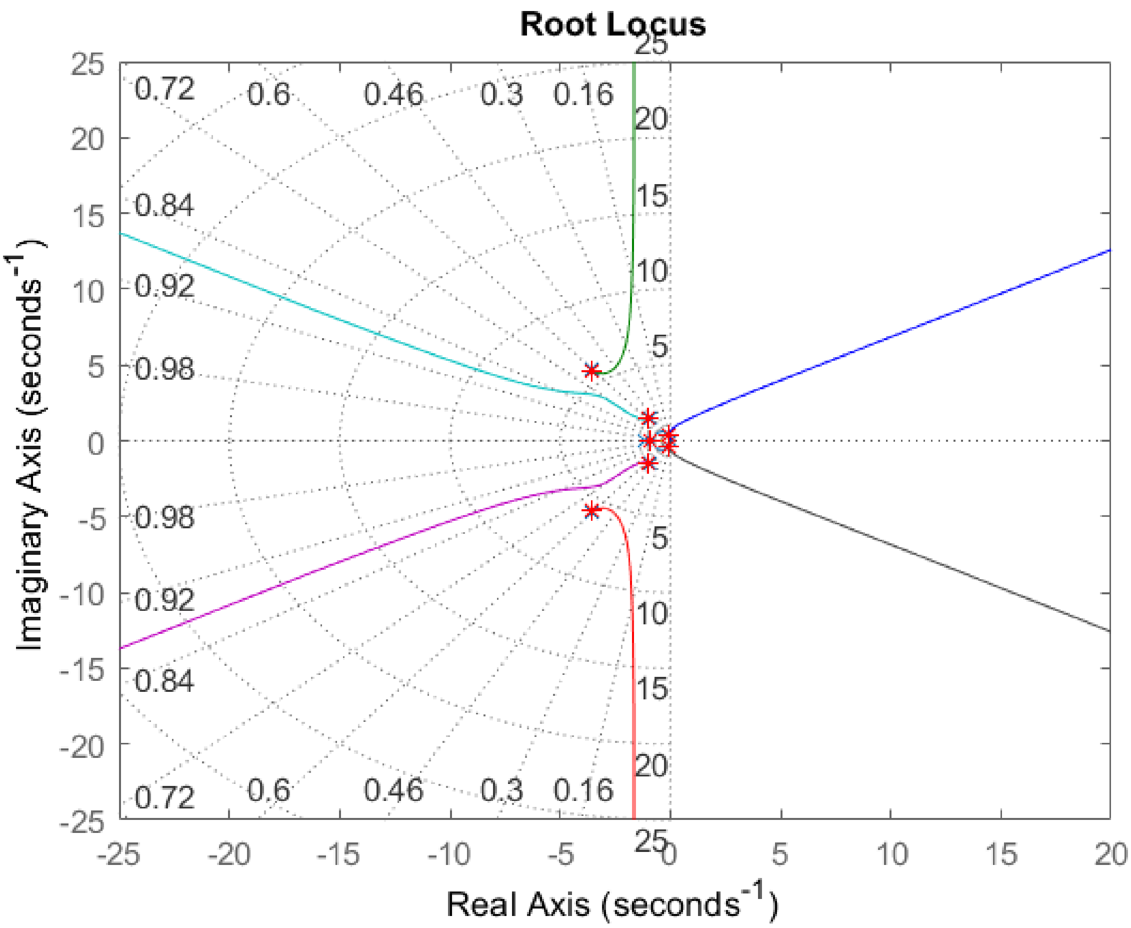

Figure 1: Root Locus of the System

The circles in the figure represent zeros, whereas crosses represent poles of the system. Besides, the stars are for the roots for the closed loop system having the maximum K value,

$$K = 30.6190$$

The only issue about the figure is the part of the root locus on the right hand side of the plane. To make this figure, I use "rlocus" command of the MATLAB. As this command does not take the optimal K value found above, but rather a value close to this K value, it supposes that one of the root is on the right hand side of the complex plane. To inform more about the stability of the system and misconceptions about the MATLAB figure, all of the roots of the system are analytically given below:

-3.52869734856663 + 4.65155780729970i
-3.52869734856663 - 4.65155780729970i
-0.993521651967020 + 1.44278914865505i
-0.993521651967020 - 1.44278914865505i
-0.886579710221883 + 0.00000000000000i
-0.0344911443554176 + 0.338173208442570i
-0.0344911443554176 - 0.338173208442570i

Table 1: Roots of the Closed Loop System

By looking at the real parts of all roots, as they are all negative values, i.e roots are on the left hand side of the complex plane, it can be stated that the closed loop system is stable.

Part C)

In this part, we find the gain, phase and delay margin of the nominal system defined in part B. The corresponding margins, calculated by MATLAB, are:

Gain Margin (GM)	2.245
Phase Margin (PM)	8.805
Delay Margin (DM)	0.452

Table 1: Gain, Phase and Delay Margins

The sensitivity function of the system can be defined as

$$S(s) = \frac{1}{1 + P_0(s)C(s)}$$

The plot for magnitude of the sensitivity function is given below:

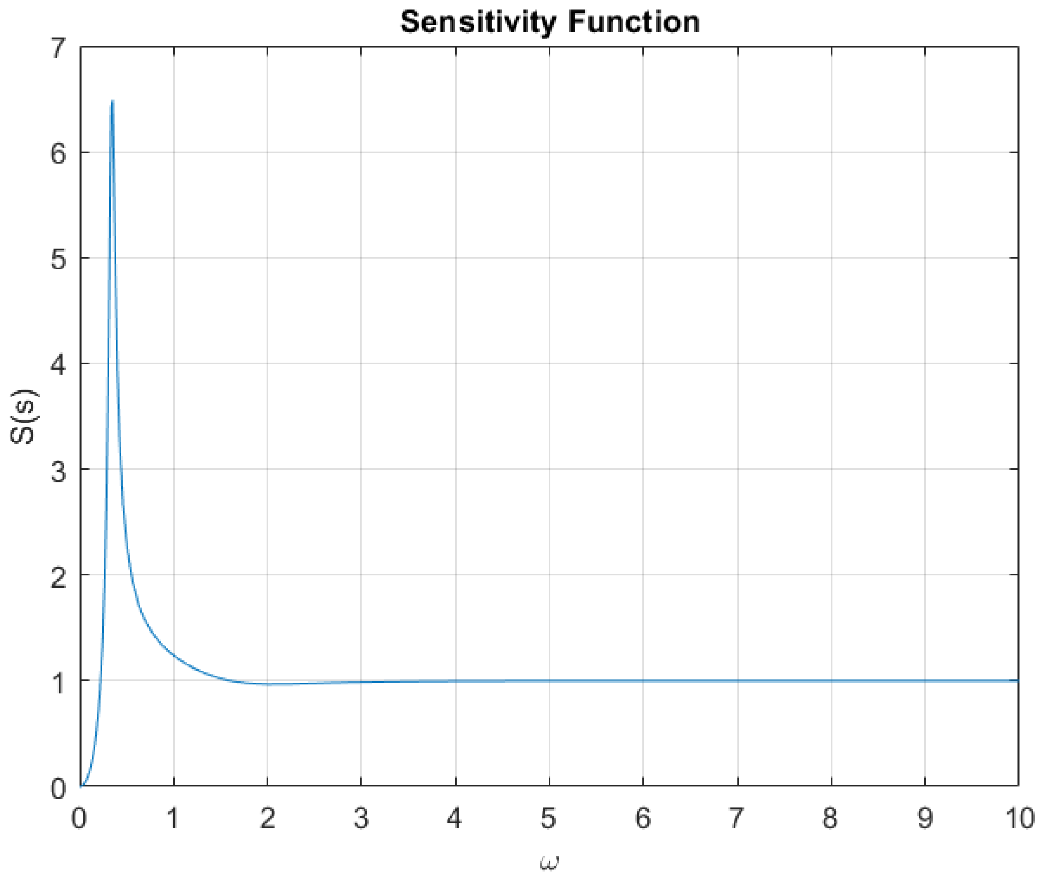

Figure 2: Sensitivity Function

Using MATLAB, the peak value of this sensitivity function is computed as **6.4904**.

APPENDIX

```
clear all;

q0min = 0.95;
q0max = 1;
q1min = 0.35;
q1max = 0.4;
r0min = 0.9;
r0max = 1;
r1min = 9;
r1max = 12;
r2min = 50;
r2max = 100;
r3min = 120;
r3max = 150;
r4min = 195;
r4max = 200;
r5min = 120;
r5max = 130;

%part a

K_interval = zeros(1,100001); %K candidates

del_max = 3.151626793; %experimental finding

K_elements = 0;
inter_index = 1;

for K = 0:0.001:100

    r6min = - del_max;
    r6max = + del_max;

    Nc = [-4 K];
    Dc = [1 0];

    N1 = [q0min q1min]; %Kharitonov numerator polynomials
    N2 = [q0max q1max];
    N3 = [q0max q1min];
    N4 = [q0min q1max];

    D1 = [r0min r1min r2max r3max r4min r5min r6max]; %Kharitonov 
denominator polynomials
    D2 = [r0max r1max r2min r3min r4max r5max r6min];
    D3 = [r0min r1max r2max r3min r4min r5max r6max];
    D4 = [r0max r1min r2min r3max r4max r5min r6min];

    XN_1 = zeros(1,8);
    XN_2 = zeros(1,8);
    XN_3 = zeros(1,8);
    XN_4 = zeros(1,8);

    XN_1(6:8) = conv(Nc,N1);
    XN_2(6:8) = conv(Nc,N2);
    XN_3(6:8) = conv(Nc,N3);
```

```
    XN_4(6:8) = conv(Nc,N4);

    XN_matrix = [XN_1;XN_2;XN_3;XN_4];

    XD_1 = conv(Dc,D1);
    XD_2 = conv(Dc,D2);
    XD_3 = conv(Dc,D3);
    XD_4 = conv(Dc,D4);

    X_matrix = (zeros(16,8));

    for i = 1:4
        X_matrix(4*i-3,:) = XN_matrix(ceil((4*i-3)/4),:) + XD_1;
        X_matrix(4*i-2,:) = XN_matrix(ceil((4*i-3)/4),:) + XD_2;
        X_matrix(4*i-1,:) = XN_matrix(ceil((4*i-3)/4),:) + XD_3;
        X_matrix(4*i,:) = XN_matrix(ceil((4*i-3)/4),:) + XD_4;
    end

    r_real_max_vec = zeros(16,1);

    %find the max real root of polynomials

    for i = 1:16
        r = roots(X_matrix(i,:));
        r_real_max = max(real(r));
        r_real_max_vec(i) = r_real_max;
    end

    %find the max K so that all 16 polynomials are stable and
    %find the interval of all Ks that the system is stable

    if(all(r_real_max_vec(:,1) < 0))
        K_interval(inter_index) = K;
        K_elements = K_elements + 1;
        K_max = K;
    end
    inter_index = inter_index + 1;
end

%part b

NF = [1 0.35];
DF = [1 10 60 140 200 121 ((-del_max / 2)-1.4) 0];
F = tf(NF,DF);

roots_K = rlocus(F,K_max);

figure
rlocus(F);
grid on
hold on
plot(roots_K,'r*');
hold off

NC_new = [-4 K_max];
DC_new = [1 0];
```

```
N_P0 = [1 0.35];
D_P0 = [1 10 60 140 200 125 (-del_max/2)];

Cc = tf(NC_new,DC_new);
Pc = tf(N_P0,D_P0);

%part c

Margins = allmargin(Cc*Pc);

Sensitivity = 1/(1 + Cc*Pc);
[mag,phase,w] = bode(Sensitivity);

disp(max(mag));

figure
plot(w,squeeze(mag));
title("Sensitivity Function");
grid on
xlabel("\omega");
ylabel("S(s)");
```