Codul in Matlab

clear all

clc

B = [1 1 5];

A = [10 7 1 0];

R = [1 3];

S = [5 2];

T = [1 2 5 6];

% Se aduc polinoamele la aceleasi dimensiuni

if length(B) <= length(T)

 B = [zeros(1,length(T)-length(B)) B];

elseif length(B) > length(T)

 T = [zeros(1,length(B)-length(T)) T];

end

if length(A) <= length(S)

 A = [zeros(1,length(S)-length(A)) A];

elseif length(A) > length(S)

 S = [zeros(1,length(A)-length(S)) S];

end

if length(B) <= length(R)

 B = [zeros(1,length(R)-length(B)) B];

elseif length(B) > length(R)

 R = [zeros(1,length(B)-length(R)) R];

end

Bo = T.\*B;

Ao = A.\*S;

Co = B.\*R;

if length(Ao) <= length(Co)

 Ao = [zeros(1,length(Co)-length(Ao)) Ao];

elseif length(Ao) > length(Co)

 Co = [zeros(1,length(Ao)-length(Co)) Co];

end

P = Ao + Co;

% Se calculeaza functiile de transfer in bucla inchisa si deschisa

Ho = tf(Bo,P);

Hd = tf(Bo,P-Bo);

% Se ploteaza harta poli-zerouri

figure

h = pzplot(Ho);

grid on

rot = roots(P);

sst = 0;

st = 0;

inst = 0;

for i = 1:length(rot)

 if rot(i) < 0

 sst = sst + 1;

 elseif rot(i) == 0

 st = st + 1;

 else

 inst = inst + 1;

 end

end

if sst >= 1 && st == 0 && inst == 0

 disp('Sistemul in bulca inchisa este strict stabil')

elseif st >= 1 && inst == 0

 disp('Sistemul in bucla inchisa este stabil. Are un pol in origine')

else

 disp('Sistemul in bucla inchisa este instabil')

end

f = 2;

if f ~= 1 || f ~= 0

 f = input('Pentru plotare apasat 1 - nyquist si 0 - bode: ' )

end

% Se ploteaza diagrama Bode/ Nyquist

if f == 0

 [mag, phase] = bode(Hd); % calculeaza raspunsul in frecventa

elseif f == 1

 [mag, phase ] = nyquist(Hd);

end

% Se caluleaza MA ?i M? ?i pulsa?iile lor

[Gm, Pm, Wcg, Wcp] = margin(Hd);

mag = 20\*log10(mag);

if Gm > 0 && Pm >0

 disp('Sisemul in bucla deschisa este stabil')

elseif Gm == 0 && Pm == 0

 disp('Sisemul in bucla deschisa este la limita de stabilitate')

elseif Gm < 0 && Pm < 0

 disp('Sistemul in bucla deschisa este instaibl')

else

 disp('Nu se poate spune nimic despre stabilitate')

end

% Se calculeaza functiile de sensibilitate

if length(A) <= length(R)

 A = [zeros(1,length(R)-length(A)) A];

elseif length(A) > length(R)

 R = [zeros(1,length(A)-length(R)) R];

end

Ar = A.\*R;

% Functiile de sensibilitate ale sistemului

S\_yp = tf(Ao,P)

S\_up = tf(Ar,P)

figure

bode(Hd)

hold on

bode(S\_yp)

bode(S\_up)

legend('Hd','S\_yp','S\_up')