Compensador por Avanço

Professor: Douglas Haupt
09/2023

1) Projetar um compensador por avanço de fase para a planta cuja função de transferência, $G(s)$, é:

$$G(s) = \frac{4}{s \cdot (s+2)}$$

Os requisitos do projeto:

- Margem de fase (m.f.) 50º
- Margem de ganho (m.g.) 10 dB;
- Coeficiente do erro K_v = 25 s^{-1}

Solução:

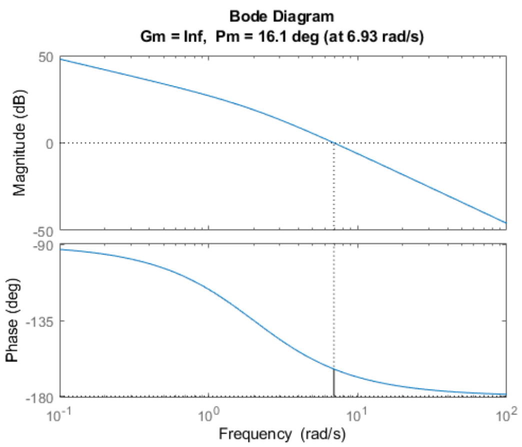
Para o projeto foi utilizado o Matlab, com o seguinte código:

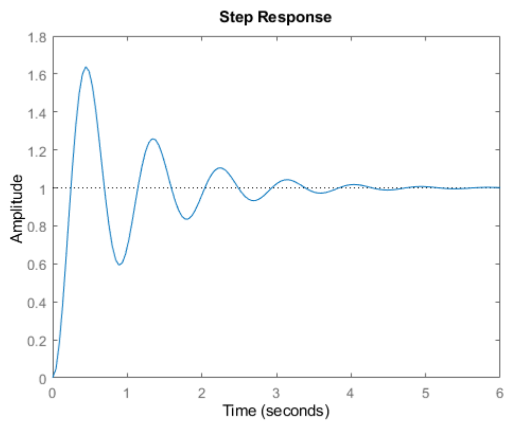
```
% Compensador por Avanco
% Project G(s) = (4)/(s(s+2))
% mf = 50
% mg = 10 dB
% Kv = 25

clc; % Limpa o que estava antes na Command Window

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%             Encontrar o Valor de Kc                     %
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

syms s T Kc alfa % Declara Variaveis simbolicas
G = 4 / (s*(s+2)); % Planta
Gc = Kc*((T*s+1)/(alfa*T*s+1)); % Ganho do compensador

Kv = limit(s*G*Gc, s, 0) == 25; % Equacao do Coeficiente do
    erro

Kc = solve(Kv, Kc); % Rosolver a equacao
Kc = eval(Kc) % Transformar o valor simbolico para double
% Kc = 12.5


%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%    Declarar a FT do Sistema nao compensado      %
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

s = tf('s');  % Declara s no estado de frequencia
G = 4 / (s*(s+2));
```

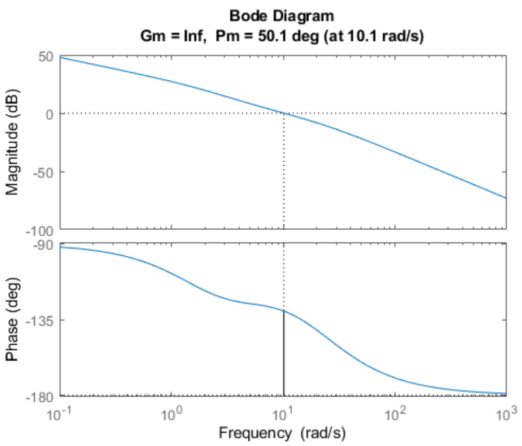
```
sys1 = Kc*G    % Sistema nao compensado, porem com ganho ajustado


%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%          Achar o valor de fase                             %
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

bode(sys1) % Para isso plotar o bode
margin(sys1) %  achar as margens e as frequencias de cruzamento
% Rodando o codigo uma vez temos que o Valor e' de 16.1

% Para entender melhor nosso sistema atual pode se plotar a
% resposta ao degrau do mesmo

sys1_cl = feedback(sys1, 1)
figure
step(sys1_cl) % Degrau aplicado


%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%         Encontrar os Parametros da                         %
%           Funcao de transferencia                          %
%        compensador por avanco de fase                      %
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%


Fim = 50 - 16.1 + 5; % defasagem igual a 38.9
a = sin((Fim * pi)/180); % convert to rad/s factor alpha
alfa = (1-a)/(1+a);
b = 20*log10(1/sqrt(alfa));
wcg = 10.1;
T = 1/(wcg*sqrt(alfa));


%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%             Achar FT compensada                            %
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%


Gc = Kc*(s*T+1)/(s*alfa*T + 1);
sys_comp = Gc * G

figure % Plotar o bode com o valor da margin
bode(sys_comp)
margin(sys_comp)

```

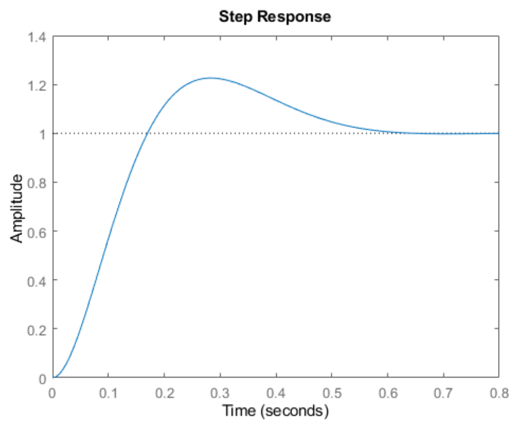
```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%           Calcular o erro da FT                   %
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

% Degrau
Ess_esp = 1/Kv % erro especificado = 1/25
planta_comp = feedback(sys_comp, 1)
figure
step(planta_comp)

planta_ramp = planta_comp/s
figure
step(planta_ramp)
[y,t] = step(planta_ramp);
ess = t(length(t)) - y(length(y)) % valor encontrado =
    1/25, fechou!!!!

figure
bode(Gc)
```

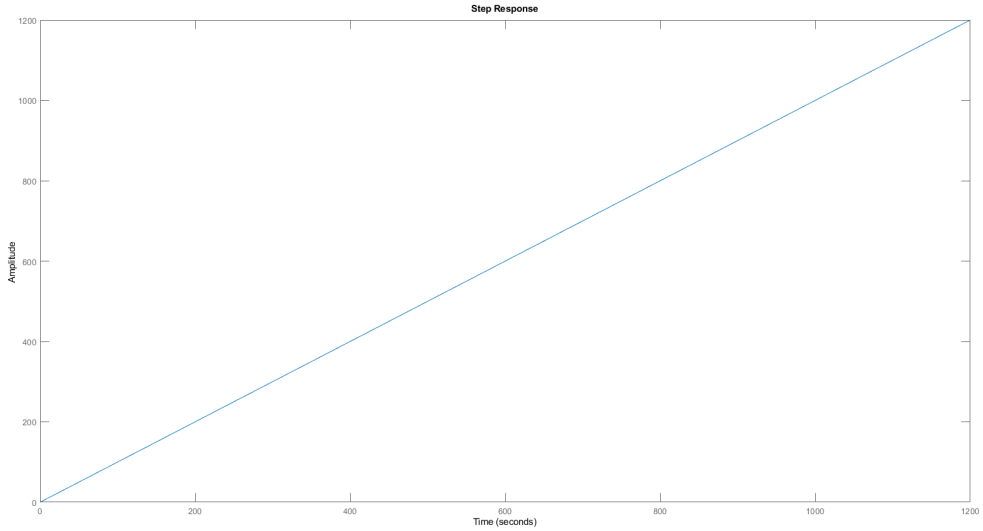
Images Output:

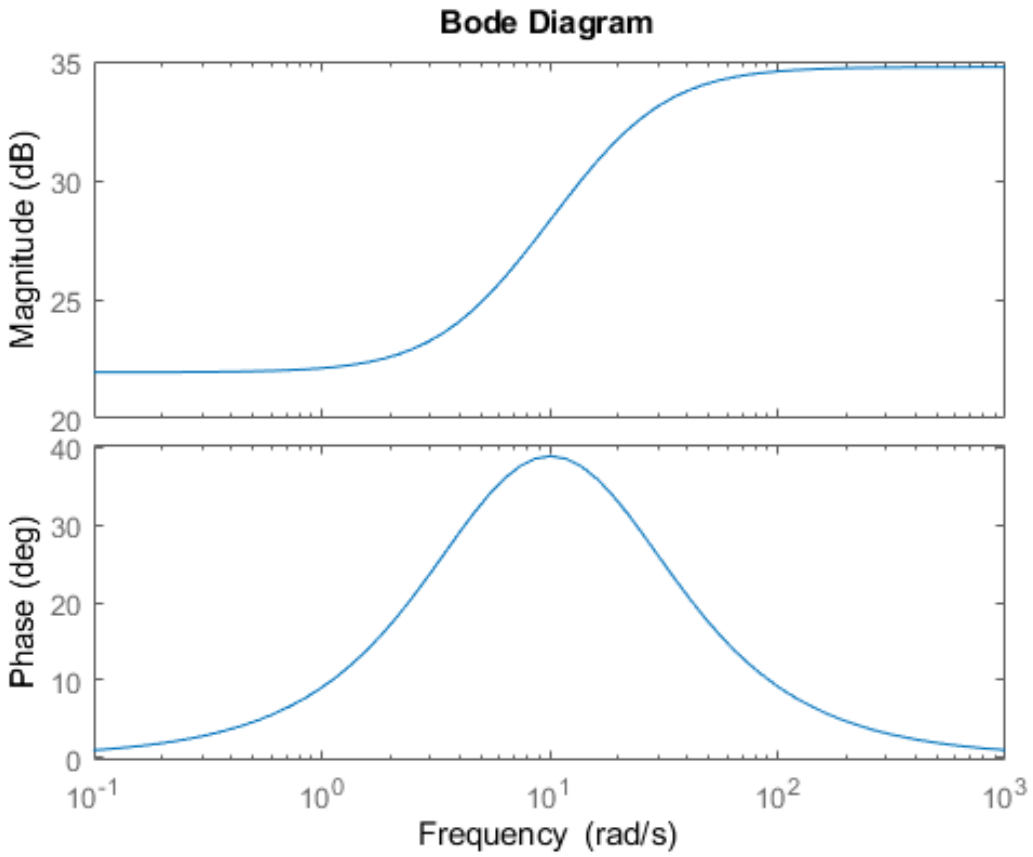












O código pode ser encontrado em meu github:

www.github.com/slemanz/CONTROL_SYSTEM