Laboratório 6

SEL0359 - Controle Digital - 2° Semestre de 2024

Prof. Marcos R. Fernandes

Projeto de controladores na frequência - parte 1

Entrega:[1] entregue um arquivo PDF com os gráficos e código fonte utilizado para cada questão além da resposta para as perguntas dos enunciados. Não esqueça de colocar título na figura para identificar o que cada figura representa, nome dos eixos, e legenda.

O objetivo dessa aula de laboratório é explorar recursos computacionais do Matlab para projeto de controladores na frequência para sistemas em tempo discreto.

Dica: Confira sempre o help do matlab para cada comando antes de utiliza-lo!

- dcgain: calcula nível dc do sistema.
- lsim: simula saída de um sistema LIT para entradas arbitrárias;
- pole: obtém polos da FT
- rlocus: obtém lugar das raízes
- stepinfo: calcula características temporais da resposta ao degrau (sobressinal, tempo de acomodação, tempo de subida, etc)

1 Atividade 1

Nessa atividade o objetivo é verificar o erro em regime permanente para sistemas em tempo discreto.

Para isso, considere a seguinte função de transferência como exemplo:

$$G(z) = \frac{z}{(z-1)(1.2706z - 0.7706)} \tag{1}$$

O seguinte código em Matlab apresenta um exemplo para fazer o gráfico da resposta ao degrau, rampa e parábola do sistema usando **lsim**.

```
%% resposta ao degrau
td=0:T:100; %vetor de tempo discreto
y1=step(Gf,td); %respota ao degrau
figure(1)
stairs(td,y1,'LineWidth',2);
hold on
```

[1] Ultima atualização: 03/10/2024

```
line([0 20],[1 1],'linestyle','--','color','black')
%% resposta a rampa
figure(2)
r=td'; %sinal de entrada (rampa)
stairs(td,td,'k--')
y_ramp=lsim(Gf,r,td); %simula entrada de sinal rampa
stairs(td,y_ramp,'LineWidth',2)
figure(3)
plot(r-y_ramp) %mostra sinal de erro (desvio de controle)
title('Erro')
%% resposta a parabola
y3=step(Gf*T*z/(z-1)^2,td);
figure(4)
r=td.*td; %sinal parabola
r=r'; %transposto para tornar vetor coluna
plot(td,td.^2,'k--')
hold on
y_parab=lsim(Gf,r,td); %simula sinal de entrada tipo parabola
stairs(td,y_parab,'--r')
figure(5)
plot(r-y_parab) %mostra sinal de erro
```

Seguindo a mesma lógica, encontre os erros de regime permanente e apresente os gráficos para entrada degrau, rampa e parábola para o seguinte sistema:

$$G(z) = \frac{0.0912z^3 - 0.0012z^2 - 0.0544z}{z^4 - 2.655z^3 + 2.679z^2 - 1.391z + 0.3679}, \quad T = 0.5s$$

Pelos erros em regime permanente, pode-se concluir que esse sistema é tipo 0, tipo 1, tipo 2 ou tipo 3?

2 Atividade 2

Nessa atividade o objetivo é avaliar a região de factibilidade para o projeto de um controlador que atenda a certos critérios de desempenho.

Para isso, considere os pólos dominantes de 2° ordem de um sistema em malha-fechada dado por

$$\frac{Y(s)}{R(s)} = \frac{\omega_n^2}{s^2 + 2\xi\omega_n s + \omega_n^2}$$

Suponha que o sistema possa ter um sobressinal máximo de 10%, frequência amortecida $\omega_d \leq 2.5 rad/s$ e $\sigma = \xi\omega_n \geq 1$ com período de amostragem $T = 0.25s$.

O código em Matlab a seguir apresenta um exemplo para avaliar, de forma gráfica, a região de factibilidade para os critérios de desempenho. Além do mais, o código também apresenta um conjunto de 100 soluções factíveis que atendem aos critérios.

```
close all
clear all
clc
%%
Mp=0.1;
xi_min=abs(log(Mp)/sqrt(pi^2+log(Mp)^2))
wd_max=2.5;
sigma_max=1;
T=0.25;
dw=0.001;
wn=0:dw:4.5*pi;
xi=xi_min;
s1=-xi*wn+1j*wn*sqrt(1-xi^2);
figure
subplot(131)
hold on
line([-1.2*pi 1.2*pi],[0 0],'linewidth',2,'color','black') %eixo-real
line([0 0],[-1.2*pi 1.2*pi],'linewidth',2,'color','black') %eixo-imag
sgrid
plot(real(s1),imag(s1),'LineWidth',2,'Color','red')
plot(real(s1),-imag(s1),'LineWidth',2,'Color','red')
wd=0:dw:4*pi;
sigma=1;
s2=-sigma+1j*wd;
plot(real(s2),imag(s2),'LineWidth',2,'Color','blue')
plot(real(s2),-imag(s2),'LineWidth',2,'Color','blue')
wd=wd_max;
sigma=0:0.001:30;
s3=-sigma+1j*wd;
plot(real(s3),imag(s3),'LineWidth',2,'Color','green')
plot(real(s3),-imag(s3),'LineWidth',2,'Color','green')
xlim([-1.2*pi 1.2*pi])
ylim([-1.2*pi 1.2*pi])
subplot(132)
hold on
z1=exp(s1*T);
z2=exp(s2*T);
z3=exp(s3*T);
line([-1.2 1.2],[0 0],'linewidth',2,'color','black') %eixo-real
line([0 0],[-1.2 1.2],'linewidth',2,'color','black') %eixo-imag
plot(real(z1),imag(z1),'LineWidth',2,'Color','red')
plot(real(z1),-imag(z1),'LineWidth',2,'Color','red')
plot(real(z2),imag(z2),'LineWidth',2,'Color','blue')
plot(real(z2),-imag(z2),'LineWidth',2,'Color','blue')
plot(real(z3),imag(z3),'LineWidth',2,'Color','green')
plot(real(z3),-imag(z3),'LineWidth',2,'Color','green')
zgrid
```

```
subplot(133)
hold on
line([0 10],[1+Mp 1+Mp],'linestyle','--','color','black')

%%
for N=1:100
    sigma=3*rand; %xi*wn
    wd=2.5*rand; %wn*sqrt(1-xi^2)
    xi=(sigma/wd)/sqrt(1+sigma^2/wd^2);
    wn=wd/xi;
    if -sigma≤-sigma_max && wd≤wd_max && xi≥xi_min
        figure(1)
    subplot(131)
    s=-sigma+1j*wd;
    plot(real(s),imag(s),'x','Color','c','LineWidth',2)
    plot(real(s),-imag(s),'x','Color','c','LineWidth',2)
    subplot(132)
    z=exp(s*T);
    plot(real(z),imag(z),'x','Color','c','LineWidth',2)
    plot(real(z),-imag(z),'x','Color','c','LineWidth',2)
    subplot(133)
    G=tf([wn^2],[1 2*xi*wn wn^2]);
    t=0:0.001:10;
    y=step(G,t);
    plot(t,y)
    drawnow
    end
end
```

Repita o código de exemplo para um projeto de controlador com a seguintes especificações:

- Sobressinal: $M_p \leq 15\%$
- Tempo de pico: $t_p \leq 0.5s$
- Tempo de acomodação: $t_s \leq 3s$ ($\pm 2\%$)

Obs:

$$M_p = e^{-\pi \frac{\xi}{\sqrt{1-\xi^2}}}$$
$$t_p = \frac{\pi}{\omega_d}$$
$$t_s = \frac{4}{\xi \omega_n}$$

3 Atividade 3

Nessa atividade, o objetivo é usar os recursos computacionais do Matlab para projetar um controlador para um sistema LIT SISO via **Emulação**.

Para isso, considere a planta em tempo contínuo descrita por

$$G(s) = \frac{1}{s(s+0.3)}$$

Deseja-se projetar um controlador em tempo contínuo com a seguinte estrutura

$$C(s) = K\frac{s-a}{s-b}$$

de tal forma que o sistema em malha fechada apresente sobressinal máximo de 16.3% ($M_p \leq 16.3\%$) e tempo de pico $t_p = 1s$.

Uma forma de fazer o projeto para esse sistema é seguir os passos:

1. O primeiro passo é, supondo que a dinâmica do sistema em malha fechada segue os pólos dominantes de 2° ordem, obter os valores de ξ e ω_n.

   Nesse exemplo, como $M_p = 16.3\%$, tem-se

   $$\xi = \left|\frac{\ln(M_p)}{\sqrt{\pi^2 + \ln(M_p)^2}}\right| = 0.5$$

   E também, como $t_p = 1s$, tem-se

   $$\omega_d = \frac{\pi}{t_p} = \pi rad/s$$

   Portanto,

   $$\omega_n = \frac{\omega_d}{\sqrt{1-\xi^2}} = 3.62 rad/s$$

2. Escolha do tempo de amostragem:

   Como $\omega_d = \pi rad/s$, então $T_d = \frac{2\pi}{\omega_d} = 2s$ é o tempo de oscilação da resposta amortecida do sistema em malha fechada. Logo, é desejável um período de amostragem, pelo menos, menor que $0.2s$ (x10 a frequência de oscilação do sistema).

3. Pólos dominantes:

   Sabendo que um sistema de 2° ordem padrão é dado por

   $$\frac{Y(s)}{R(s)} = \frac{\omega_n^2}{s^2 + 2\xi\omega_n s + \omega_n^2}$$

Verifica-se que os pólos, para $0 < \xi < 1$, são dados por

$$s_{1,2} = -\xi\omega_n \pm j\omega_n\sqrt{1-\xi^2}$$

Logo, para esse exemplo, tem-se

$$s_{1,2} = -1.81 \pm j\pi$$

Dessa forma, para que o sistema atenda os critérios de desempenho é necessário que o Lugar Geométrico das Raízes (LGR) do sistema compensado passe por esses pólos!

4. Sintonia do compensador:

   A sintonia do controlador consiste em encontrar os valores de a e b que satisfaçam os critérios de desempenho especificados para o sistema de controle. Isso pode ser feito analiticamente ou através de algoritmos de otimização.

   Para esse exemplo, como existe um pólo **estável** em $s = -0.3$, pode-se escolher o zero do compensador de forma a cancelar esse pólo tomando $a = -0.3$. O que resulta em

   $$C(s)G(s) = K\frac{(s+0.3)}{s-b}\frac{1}{s(s+0.3)} = K\frac{1}{s(s-b)}$$

   Assim, em malha fechada tem-se

   $$\frac{Y(s)}{R(s)} = \frac{C(s)G(s)}{1+C(s)G(s)} = \frac{K}{s^2-bs+K}$$

   Logo, por inspeção com a função de transferência de sistemas de 2° ordem padrão, conclui-se que

   $$K = \omega_n^2 = 13.16, \quad b = -2\xi\omega_n = -3.62$$

   Então, o controlador resultante é

   $$C(s) = 13.16\frac{s+0.3}{s+3.62}$$

Fazendo a discretização usando método de mapeamento casado de pólos e zeros com $T = 0.2s$, tem-se

$$C_d(z) = \frac{U(z)}{E(z)} = \frac{9.6416(z-0.9418)}{z-0.484}$$

Dessa forma, a equação à diferenças que implementa esse controlador é

$$u[k] = 0.494u[k-1] + 9.6416e[k] - 9.08e[k-1]$$

O seguinte código em Matlab simula esse controlador para diferentes tempos de amostragem.

```
close all
clear all
clc
%%
Mp=0.163;
tp=1;
s=tf('s');
G=1/(s*(s+0.3))
%%
xi=abs(abs(log(Mp)/sqrt(pi^2+log(Mp)^2)))
wd=pi/tp
wn=wd/sqrt(1-xi^2)
Td=2*pi/wd
fd=1/Td
fs=10*fd
Ts=1/fs
sigma=xi*wn
a=-0.3;
K=wn^2
b=-2*sigma
%% Tempo continuo
C=zpk(a,b,K)
figure
rlocus(C*G)
hold on
Gf=C*G/(1+C*G);
p=pole(Gf)
plot(real(p),imag(p),'xr','LineWidth',2)
ylim([-4 4])
info=stepinfo(Gf)
dt=0.0001;
t=0:dt:10;
y=step(Gf,t);
figure
plot(t,y,'LineWidth',1.5)
hold on
%% Tempo discreto (emulacao)
Ts=[0.2 0.1 0.01];
for i=1:numel(Ts)
Cd=c2d(C,Ts(i),'matched');
Gd=c2d(G,Ts(i),'matched');
Gfd=Cd*Gd/(1+Cd*Gd);
td=0:Ts(i):10;
info=stepinfo(Gfd)
yd=step(Gfd,td);
stairs(td,yd,'LineWidth',1.5)
end
```

```
line([0 max(t)],[1 1],'color','black','linestyle','--')
legend('Continuo','Ts=0.2','Ts=0.1','Ts=0.01')
```

Usando a mesma estratégia descrita anteriormente, projete um controlador para a planta

$$G(s) = \frac{1}{s(s+0.5)}$$

que garanta:

- $M_p \leq 12\%$
- $t_p \leq 2s$

Escolha um tempo de amostragem 10x menor que o tempo de oscilação amortecida do sistema e depois repita com tempos de amostragem cada vez menores.

O que pode-se concluir em relação ao comportamento em tempo discreto e tempo contínuo?

4 Atividade 4

Nessa atividade, o objetivo é projetar um controlador direto no domínio Z.

Para isso, considere o primeiro passo é discretizar a planta considerando o ZOH. Uma forma prática de fazer é tomar o método degrau-invariante,

```
Gd=c2d(G,'zoh')
```

Considerando novamente o exemplo da atividade 3, tem-se

$$G_d(z) = K_g \frac{z-\alpha}{(z-1)(z-\beta)}$$

com $K_g = 0.0196$, $\alpha = -0.9802$ e $\beta = 0.9418$. Em seguida, busca-se obter a função de transferência do controlador na forma

$$C(z) = K\frac{z-a}{z-b}$$

Pode-se escolher o zero do controlador de forma a cancelar o pólo estável do sistema, resultando em

$$C(z)G_d(z) = K_g K \frac{z-\alpha}{(z-b)(z-1)}$$

Em malha-fechada, se torna:

$$\frac{Y(z)}{R(z)} = \frac{K_g K(z-a)}{z^2-(b+1-K_g K)z+(b-K_g K\alpha)}$$

Da mesma forma que no exemplo da atividade 3, os pólos dominantes em malha fechada devem ser

$$s_{1,2} = -1.81 \pm j\pi$$

Fazendo o mapeamento para o plano-z, com $T = 0.2s$, resulta em

$$z_{1,2} = e^{s_{1,2}T} = 0.5629 \pm j0.4089$$

Logo, a equação característica em malha-fechada é

$$(z - z_1)(z - z_2) = z^2 - \gamma z + \rho$$

como $\gamma = 1.1258$ e $\rho = 0.4841$. Assim, a sintonia do controlador deve satisfazer

$$\begin{cases} b - K_g K = \gamma - 1 \\ b - K_g K \alpha = \rho \end{cases}$$

A solução, então, é dada por

$$b = 0.3067, \quad K = 9.2294$$

Dessa forma, o controlador resultante é

$$C(z) = 9.2294 \frac{z - 0.9418}{z - 0.3067}$$

O código Matlab a seguir simula esse controlador.

```
close all
clear all
clc
%%
Mp=0.163;
tp=1;
s=tf('s');
G=1/(s*(s+0.3))
%%
xi=abs(abs(log(Mp)/sqrt(pi^2+log(Mp)^2)))
wd=pi/tp
wn=wd/sqrt(1-xi^2)
Td=2*pi/wd
fd=1/Td
fs=10*fd
Ts=1/fs
sigma=xi*wn
a=-0.3;
```

```
K=wn^2
b=-2*sigma
%% Tempo discreto
Ts=0.2;
Cd=zpk(0.9418,0.3067,9.2294,Ts)
Gd=c2d(G,Ts,'zoh');
Gfd=Cd*Gd/(1+Cd*Gd);
td=0:Ts:10;
info=stepinfo(Gfd)
yd=step(Gfd,td);
stairs(td,yd,'LineWidth',1.5)
line([0 max(td)],[1 1],'color','black','linestyle','--')
legend('Ts=0.2')
```

Seguindo a mesma lógica, projete um controlador diretamente no domínio Z para a planta

$$G(s) = \frac{1}{s(s+0.5)}$$

que garanta:

- $M_p \leq 12\%$
- $t_p \leq 2s$

Escolha um tempo de amostragem 10x menor que o tempo de oscilação amortecida do sistema.

5 Atividade 5

Dada a função de transferência em cascata com um zero order hold (ZOH) e realimentação unitária:

$$G(s) = \frac{15}{(s+2)(s+4)}$$

Projete um controlador cujo sobressinal máximo seja $M_p \leq 10\%$, tempo de acomodação $t_s \leq 2$ e erro para entrada a degrau nulo. Considere $T = 0.05s$.

Encontre a equação à diferenças que implementa o controlador calculado.

Num mesmo gráfico, trace a resposta ao degrau do sistema original e do sistema compensado. O sistema atendeu as especificações? Caso não atenda, explique os motivos.