

Sistemas Realimentados - 2024/1

EP 26 - Projeto do Controlador Proporcional Derivativo

Nome: André Thomaz Fabris e Cayque Pires Monteiro

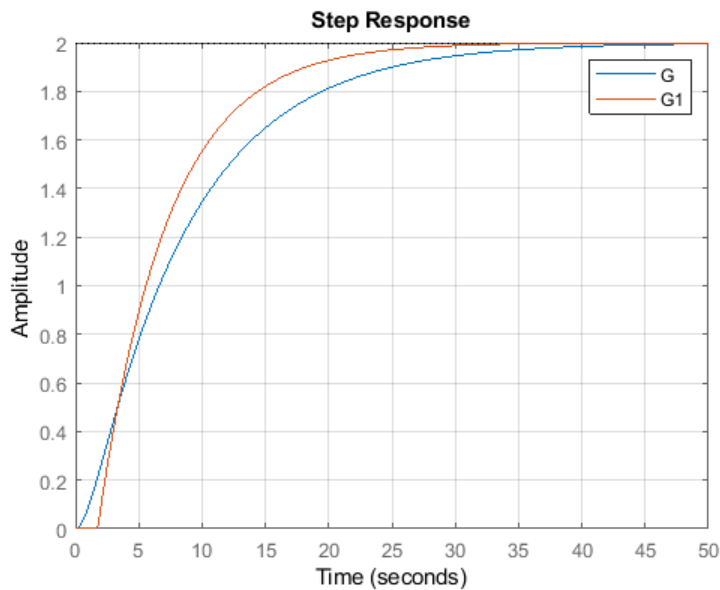
Seja a FT: $G(s) = \frac{2}{(s+1)(8s+1)}$

1) Mostre que o modelo de ordem 1 dado por $G_1(s) = \frac{2e^{-1,75s}}{5,5s+1}$ é uma boa aproximação de $G(s)$, simulando ambos ao degrau

```
warning off
G = tf(2,[8 9 1]);
pole(G)
```

```
ans = 2x1
     -1.0000
     -0.1250
```

```
G1 = tf(2,[5.5 1]);
G1.InputDelay = 1.75;
figure
step(G,G1)
legend('G','G1')
grid on
```



```
L = stepinfo(G);
Tipo = {'G'};
ts = L.SettlingTime;
tr = L.RiseTime;
Tabela_G = table (Tipo,ts,tr);
P = stepinfo(G1);
Tipo = {'G1'};
ts = P.SettlingTime;
tr = P.RiseTime;
Tabela_G1 = table (Tipo,ts,tr);
Tabela_Unica = vertcat(Tabela_G,Tabela_G1)
```

Tabela_Unica = 2x3 table

	Tipo	ts	tr
1	'G'	32.3646	17.8119
2	'G1'	23.2668	12.0843

Ao simular ambas FTs ao degrau, percebe-se que as curvas se assemelham tal qual os tempos de estabelecimento e tempo de subida. Apesar de G1 possuir tempo morto, ela ainda se estabiliza na referência mais rápido que G ($G_{ts} > G1_{ts}$). Ambas as curvas tendem para o mesmo valor de regime.

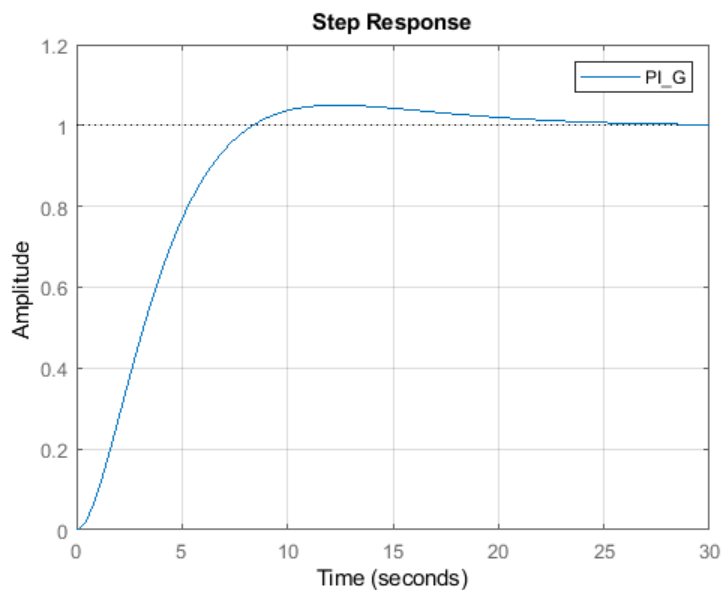
2) Projete um controlador PI usando o método do IAE ótimo em $G1(s)$ e teste o controlador $C(s)$ em $G(s)$.

```
C=pidtuning(g1,'method','iaeot','type','PI');
```

Mostre a resposta ao degrau com este controlador e os gráficos de Bode de $G(s)$ e $C(s)G(s)$, avaliando como ficaram as margens de fase e ganho.

Como mostrada no tópico anterior, a aproximação se faz necessária pois o método de sintonia do IAE Ótimo só pode ser aplicado em sistemas de primeira ordem com tempo morto.

```
cpi_iae = pidtuning(G1,'method','iaeot','type','PI');  
m1 = feedback(cpi_iae*G1,1);  
m2 = feedback(cpi_iae*G,1);  
step(m2)  
legend('PI_G')  
grid on
```



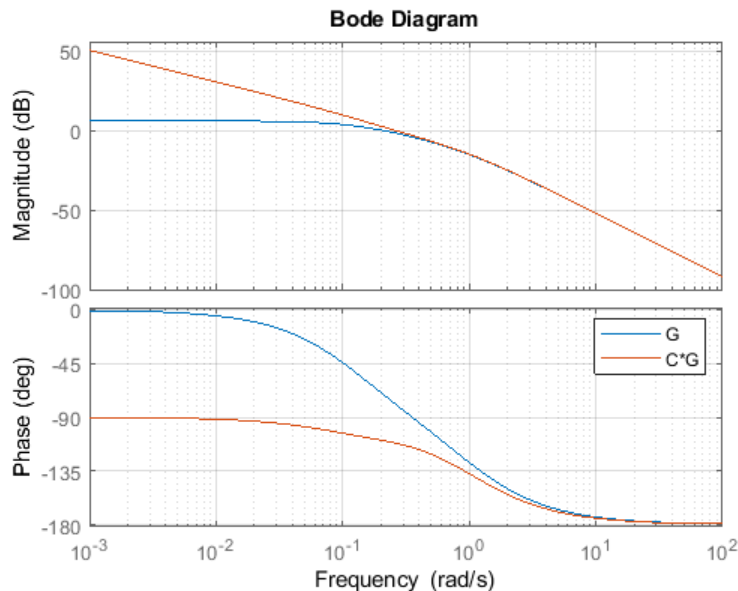
```
P = stepinfo(m2);  
Tipo = {'PI_G'};  
Up = P.Overshoot;  
ts = P.SettlingTime;  
Tabela_m2 = table (Tipo,Up,ts)
```

Tabela_m2 = 1×3 table

	Tipo	Up	ts
1	'PI_G'	5.0261	19.9754

Inicialmente foi projetado o controlador PI via IAE ótimo em G1. Após este passo, o mesmo controlador foi utilizado para fechar malha em G. Ambas as respostas ao degrau podem ser observadas no primeiro gráfico.

```
bode(G,cpi_iae*G)
legend('G','C*G')
grid on
```



Pelo gráfico de Bode, observa-se que a Margem de Ganho nos dois casos é infinita, pois as duas curvas não cruzam -180° no diagrama de fase. Já a Margem de Fase analisa-se quando o gráfico de magnitude cruza 0 db, logo temos os seguintes valores:

Margem de fase G = 109 graus

Margem de fase C*G = 68 graus

Como o PI adiciona um polo na origem, ele desloca o gráfico de magnitude para cima inicialmente. Além disso, com um zero adicionado mais próximo da origem em -1,66 (sendo $K_p = 1,02$ e $K_i = 0,17$), ele deixa o sistema mais rápido diminuindo a margem de fase como se observa acima. É importante ressaltar que a fase se inicia em -90° em C*G devido ao polo na origem.

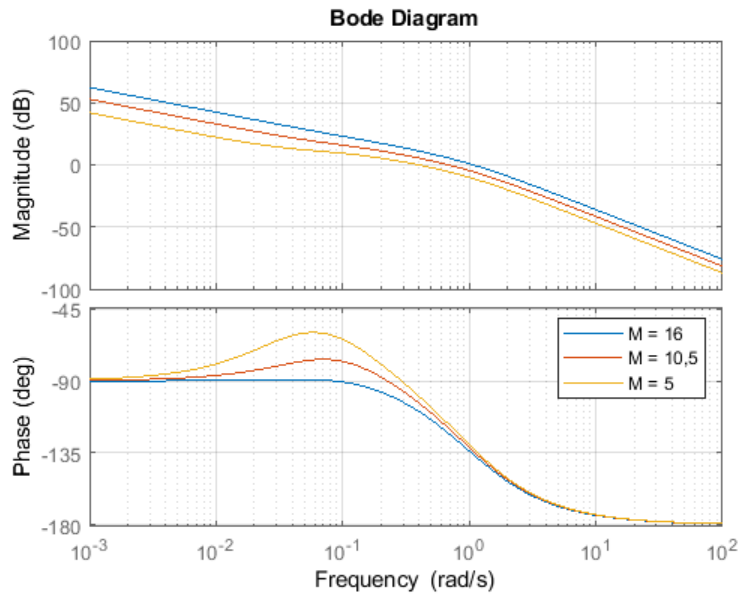
3) Use o método de projeto do controlador PI via gráfico de Bode apresentado em aula em G(s) para obter o mesmo desempenho em termos de sobreelevação e tempo de estabelecimento do item 2.

```
% O funcionamento da função projpi está sendo explicado no item 4
[c1, wg1] =projpi(G,10.5);
[c2, wg2] =projpi(G,16);
[c3, wg3] =projpi(G,5);
```

```

bode(c2*G,c1*G,c3*G)
legend('M = 16','M = 10,5','M = 5')
grid on

```



Margem de fase para $M = 16 \rightarrow 44,1$ graus

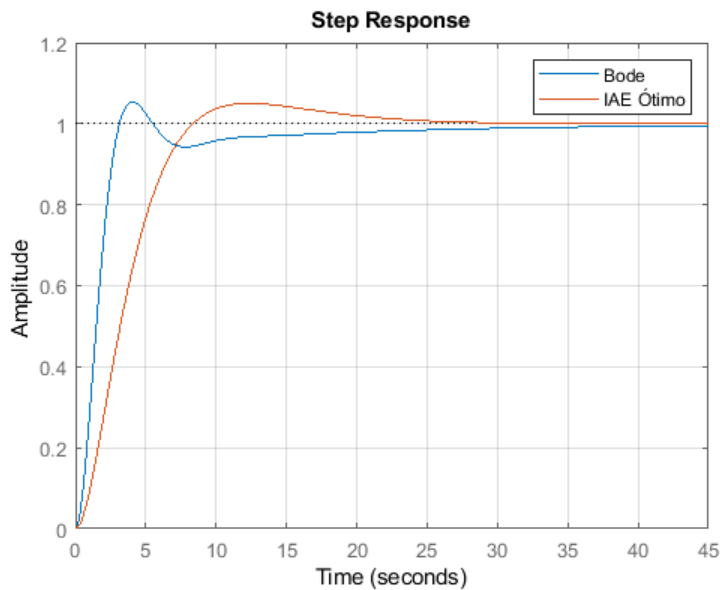
Margem de fase para $M = 10,5 \rightarrow 60,5$ graus

Margem de fase para $M = 5 \rightarrow 80,8$ graus

```

m3 = feedback(c1*G,1);
m4 = feedback(c2*G,1);
m5 = feedback(c3*G,1);
step(m3,m2)      %Sendo m2 a FTMF de G utilizando o controlador obtido pelo
método do IAE Ótimo, mostrado no item 2.
legend('Bode','IAE Ótimo')
grid on

```



O controlador projetado no item acima via IAE Ótimo fechou malha com $G(s)$ e obteve os seguintes parâmetros: $UP = 5,02\%$ e $ts = 19,98$ s. Assim, devemos projetar um novo controlador PI, porém agora utilizando o método via gráfico de Bode. Inicialmente, aumentamos o módulo em 5db ($M = 5$), ou seja, deslocamos o gráfico para cima 5 db. Com isso, obtivemos $UP = 0\%$ e $ts = 75,81$ s. O sistema, por sua vez, ficou muito lento, portanto devemos deslocar ainda mais para cima e tornar a resposta mais rápida. Logo, aumentamos o módulo em 16db ($M = 16$) e chegamos nos seguintes parâmetros: $UP = 23,62\%$ e $ts = 6,8$ s. Desta vez, o sistema ficou rápido demais e com muita sobreelevação. Assim, devemos optar por um valor próximo, porém menor que $M = 16$. Escolhemos $M = 10,5$ e chegamos em uma resposta bem próxima do desejado, sendo $UP = 5,42\%$ e $ts = 20,79$ s. Em resumo, devemos deslocar a magnitude do gráfico de Bode para cima, cortar 0 db em frequências mais altas, o suficiente para diminuir a Margem de Fase e tornar o sistema mais rápido. Ainda sim, deve-se balizar para chegar o mais perto possível da especificação desejada. A tabela abaixo sintetiza o resultado das escolhas feitas:

```
H = stepinfo(m3);
Tipo = {'M = 10.5'};
ts = H.SettlingTime;
Up = H.Overshoot;
Tabela_m3 = table (Tipo,Up,ts);
I = stepinfo(m4);
Tipo = {'M = 16'};
ts = I.SettlingTime;
Up = I.Overshoot;
Tabela_m4 = table (Tipo,Up,ts);
J = stepinfo(m5);
Tipo = {'M = 5'};
ts = J.SettlingTime;
Up = J.Overshoot;
Tabela_m5 = table (Tipo,Up,ts);
```

```
Tabela_Unica = vertcat(Tabela_m2,Tabela_m3,Tabela_m4,Tabela_m5)
```

```
Tabela_Unica = 4x3 table
```

	Tipo	Up	ts
1	'PI_G'	5.0261	19.9754
2	'M = 10.5'	5.4214	20.7911
3	'M = 16'	23.6241	6.7969
4	'M = 5'	0	75.8168

4) Compare a localização do zero usando o método do IAE ótimo e o projeto via Bode, discutindo as possibilidades de melhorar o projeto.

```
cpi_iae % Zero = - Ki/Kp = -0,1667
```

```
cpi_iae =
```

$$K_p + K_i * \frac{1}{s}$$

```
with Kp = 1.02, Ki = 0.17
```

```
Continuous-time PI controller in parallel form.
```

```
c1 % Zero = - Ki/Kp = -0,066
```

```
c1 =
```

$$\frac{3.35 s + 0.2199}{s}$$

```
Continuous-time transfer function.
```

Para comparar a localização do zero usando o método do IAE ótimo e o projeto via Bode, analisaremos como cada método determina a posição do zero do controlador PI e discutiremos como melhorar o projeto.

Método do IAE Ótimo (pidtuning)

Por meio da função `pidtuning.m`, ela define um método para sintonização de controladores usando diferentes abordagens, incluindo o método do IAE ótimo. Este método otimiza o Integral of Absolute Error (IAE) para encontrar os ganhos K_p e K_i do controlador PI. O foco está na minimização do erro integral, que influencia diretamente a posição do zero do controlador PI.

Localização do Zero e Melhorias:

- A função calcula K_p e K_i , definindo a localização do zero, para minimizar o IAE.

Projeto via Bode (projpi)

A função projpi.m utiliza um método gráfico baseado no diagrama de Bode para determinar os ganhos do controlador PI. Este método ajusta o ganho proporcional (K_p) por meio da entrada do parâmetro M , sendo M a magnitude em dB que se deseja ajustar no diagrama de Bode. Se M for um único valor, K_p é ajustado para que a magnitude do sistema $G(s)$ no diagrama de Bode atinja M dB, logo $K_p = 10^{(M/20)}$. Para calcular K_i , a função determina a frequência de cruzamento de ganho (ω_g), que é a frequência na qual a magnitude do sistema multiplicado por K_p cruza 0 dB. Isso é feito identificando o índice onde a magnitude em dB é positiva pela última vez. Então faz-se $K_p \times \omega_g/10$, colocando o zero do controlador PI de forma a influenciar a resposta do sistema de maneira desejada, ou seja, a margem de ganho ou fase desejada seja atingida.

Localização do Zero e Melhorias:

- A função determina K_p e K_i ajustando diretamente o ganho e a frequência de cruzamento de ganho (ω_g), que influencia onde o zero do controlador PI é colocado. O zero é posicionado em $-K_i/K_p$.
- Oferece uma maneira mais direta e controlada de posicionar o zero, ajustando K_p e K_i para atingir margens de estabilidade específicas.
- Verifica se a localização do zero atende aos requisitos de desempenho em termos de tempo de subida, overshoot e tempo de estabelecimento. Caso a localização do zero deixe a resposta lenta, ela pode ser ajustada próximo de ω_g reduzindo a margem de fase e o tempo de estabelecimento.

O método do IAE Ótimo somente pode ser usado em FOPTD, enquanto o método de Bode pode ser usado em qualquer FT.