Projeto de controladores PID via método do lugar das raízes

Sistemas Realimentados

Conteúdo

1. Especificação de projeto no domínio do tempo
2. Projeto de controladores PD
3. Projeto de controladores PI
4. Projeto de controladores PID

Projeto de controladores via LR

Ao projetar controladores no domínio do tempo, as especificações usuais são:

- Estabilidade
- Erro em regime
- Sobreelevação
- Tempo de estabelecimento
- IAE

Elas são utilizadas para orientar os projetos, e o atendimento das especificações é verificado via simulação ao degrau.

Projeto de controladores via LR

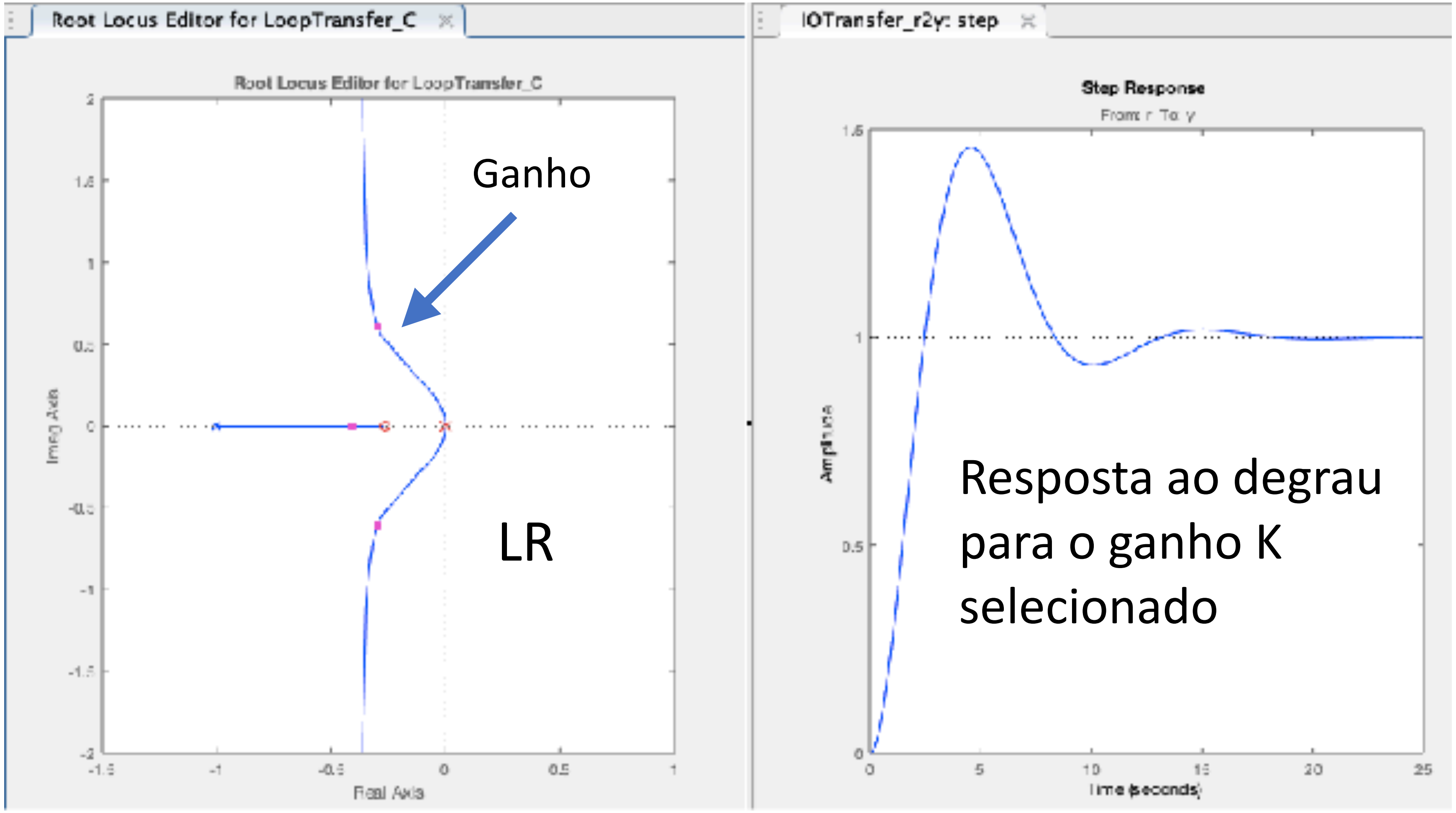
O controlador PID tem 3 parâmetros: $C(s) = K_P + \frac{K_I}{s} + K_D s$ e o método do lugar das raízes permite avaliar o efeito de um dos parâmetros.
O projeto é separado em dois: PI e PD, e um dos parâmetros ficará constante no projeto.

Projeto do controlador PD: $C(s) = K_P(s + \frac{K_D}{K_P}s)$ K_P atende condição de regime e K_D varia.

Projeto do controlador PI: $C(s) = K_P \frac{s + \frac{K_I}{K_P}}{s}$ Escolhe-se uma relação $\frac{K_I}{K_P}$ e o LR é feito em função de K_P.

Projeto de controladores via LR

A ferramenta rltool facilita grandemente os projetos do PID, pois a resposta ao degrau é mostrada junto com o LR.



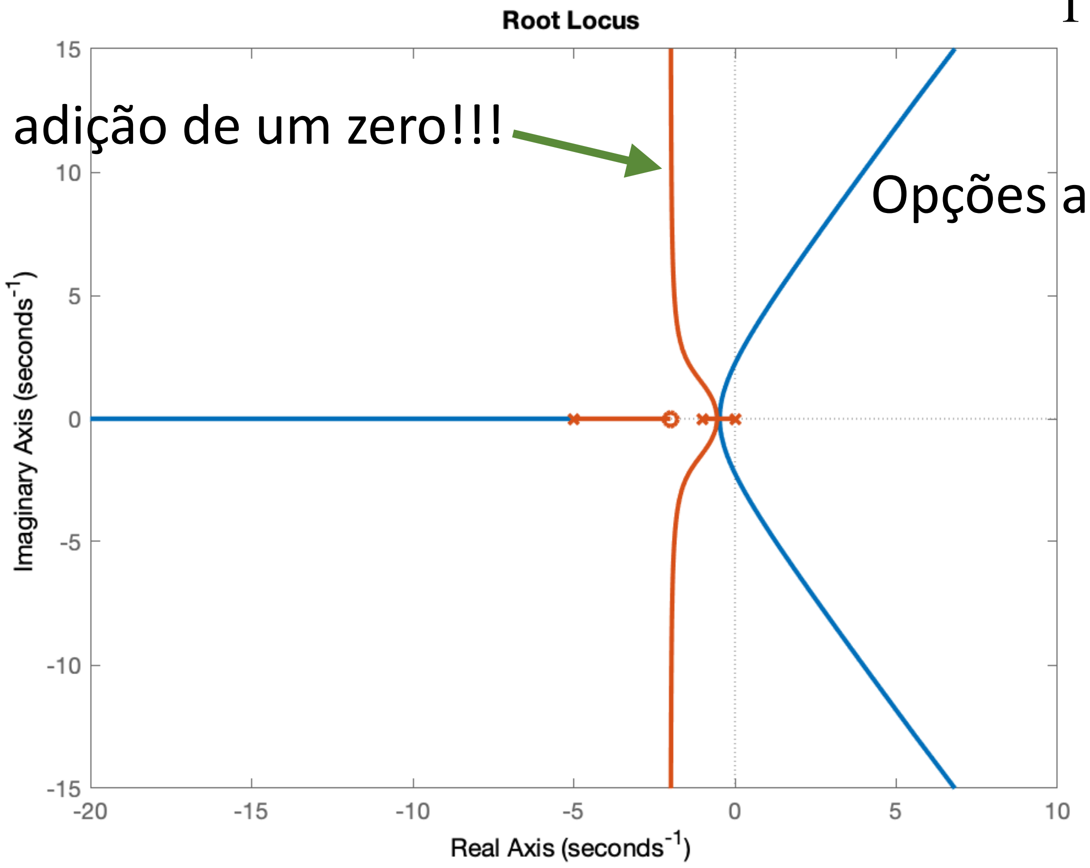
Projeto de controladores via LR

Além de alterar o ganho, agora adicionaremos polos de zeros no LR.

$$1 + K\frac{1}{s(s+1)(s+5)} = 0$$

Opções após a adição de um zero!!!

Opções apenas variando o ganho K



1. Projeto de controladores PD

O controlador proporcional derivativo é dado por

$$G_c(s) = K_p + K_d s$$

ou

$$G_c(s) = K_p(1 + \frac{K_d}{K_p}s)$$

Assim, o ganho K_p deve ser escolhido para atender o erro em regime e o tempo de estabelecimento, e K_d para atender a condição de transitório, especificado em termos de sobreelevação.

1. Projeto de controladores PD

Uma vez definido o valor de K_p, o LR é feito em função de K_D,

ou

$$1 + (K_P + K_D s)G(s) = 1 + (K_P + K_D s)\frac{N(s)}{D(s)}$$

Isolando o termo com K_D ,

$$D(s) + (K_P + K_D s)\frac{N(s)}{D(s)} = 1 + K_D s\frac{N(s)}{D(s) + K_P N(s)} = 1 + \frac{K_D s}{\frac{D(s)}{N(s)} + K_P}$$

1. Projeto de controladores PD

Exemplo 1: Seja a FT de MA dada por

$$G(s) = \frac{1}{s(s+10)}$$

Como se trata de um sistema tipo 1, o erro ao degrau é nulo. Vamos considerar o requisito de um erro à entrada rampa menor do que 10%. Neste caso, o ganho Kp deve ser maior ou igual a 100, assumindo K=1.

Caso o que se queira seja apenas erro nulo ao degrau, o ganho Kp pode ser escolhido livremente para tornar a resposta mais rápida.

1. Projeto de controladores PD

As FT de MA G(s) e MF M(s) são dadas por

$$G(s) = \frac{K_p + K_d s}{s(s+10)}$$

$$M(s) = \frac{K_p + K_d s}{s^2 + (10 + K_d)s + K_p}$$

Para usar o LR precisamos escrever o polinômio característico na forma

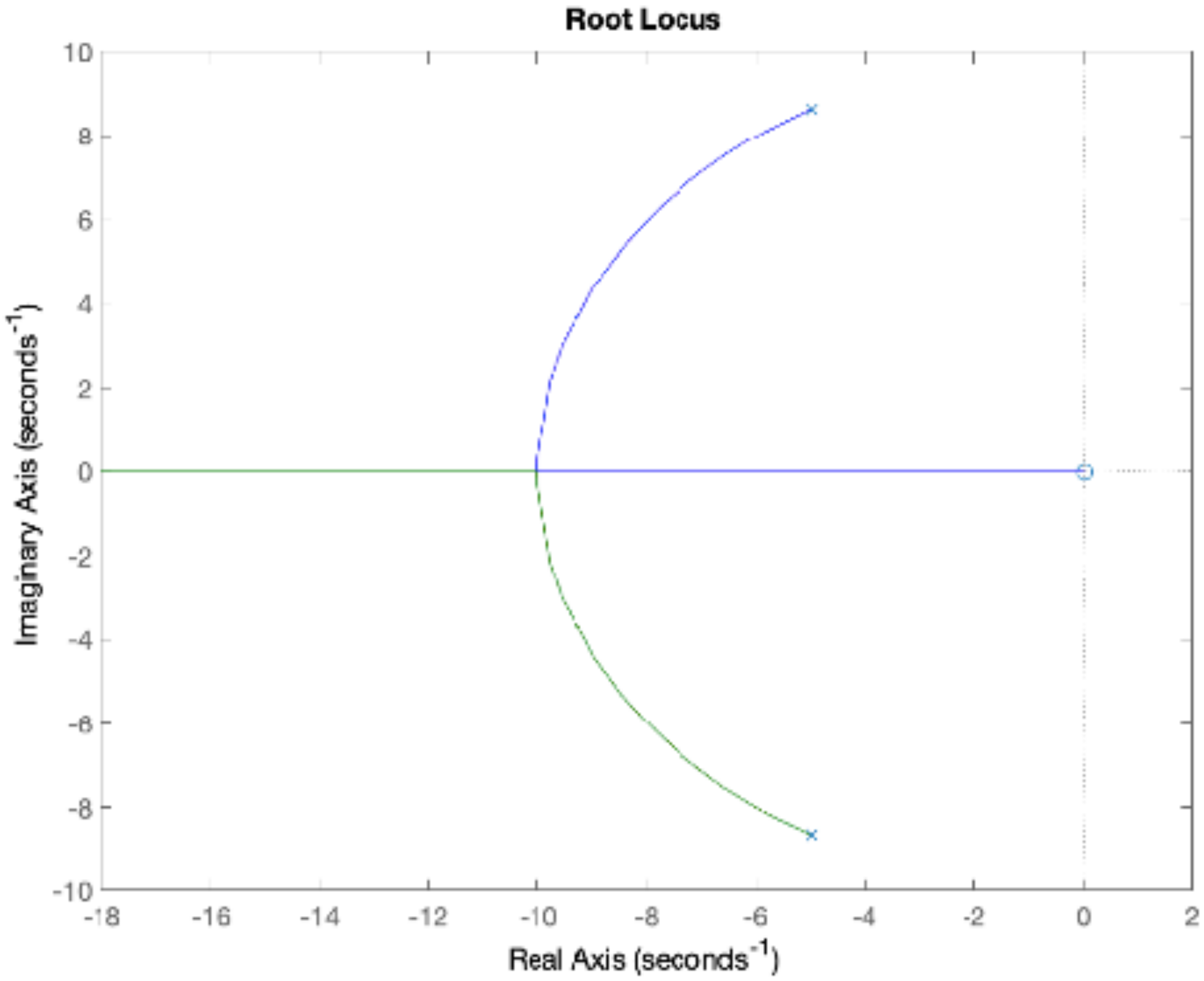
$$1 + KG_1(s) = 1 + K\frac{N(s)}{D(s)} = 0$$

$$\boxed{1 + \frac{K_d s}{s^2 + 10s + 100} = 0}$$

$K_P = 100$

1. Projeto de controladores PD

g=tf([1 0],[1 10 100])
rlocus(g)



Observa-se que pode-se conseguir qualquer amortecimento variando K_D.

Entretando, valores grandes de K_D fazem um dos polos de MF ficar perto da origem (lento).

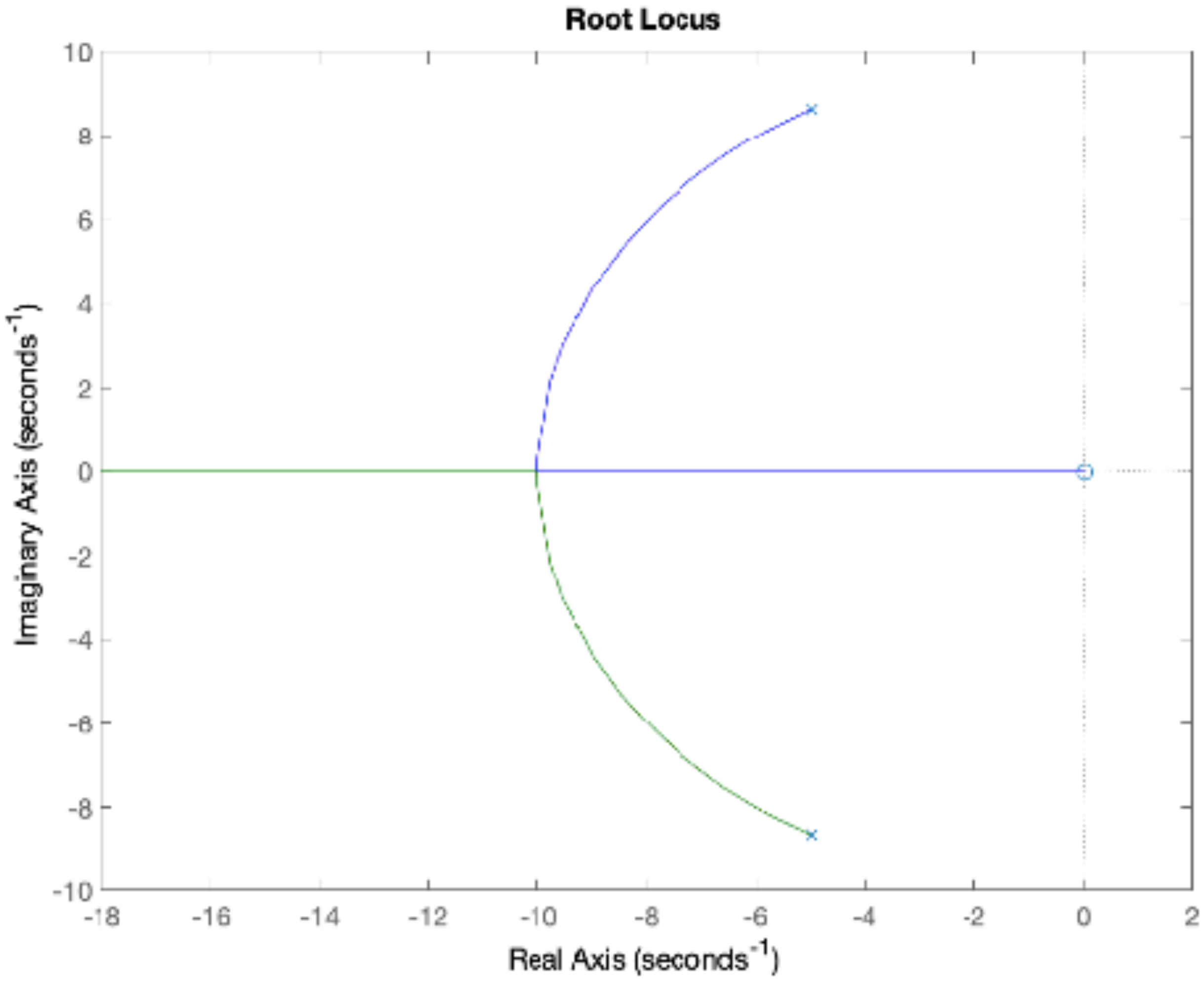
1. Projeto de controladores PD

```
g=tf([1 0],[1 10 100])
rlocus(g)
```

Para obter Kd, seleciona-se um ponto sobre o LR.
Por exemplo, s=-10.
Neste caso,

$$K_d = -\frac{s^2 + 10s + 100}{s}$$

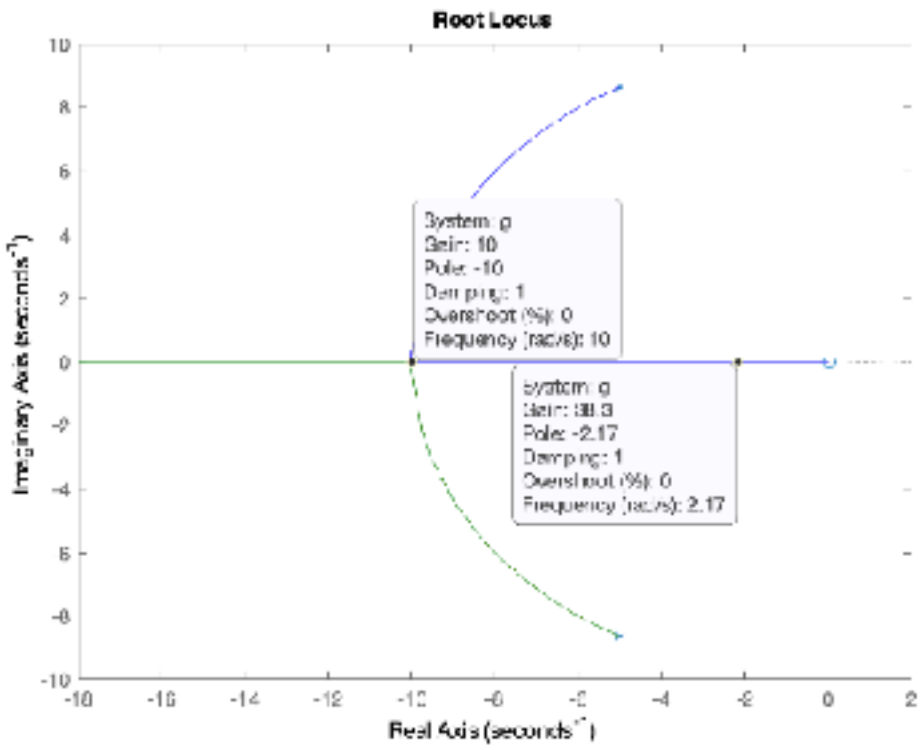
$$\boxed{K_d = -\frac{(-10)^2 - 100 + 100}{(-10)} = 10}$$

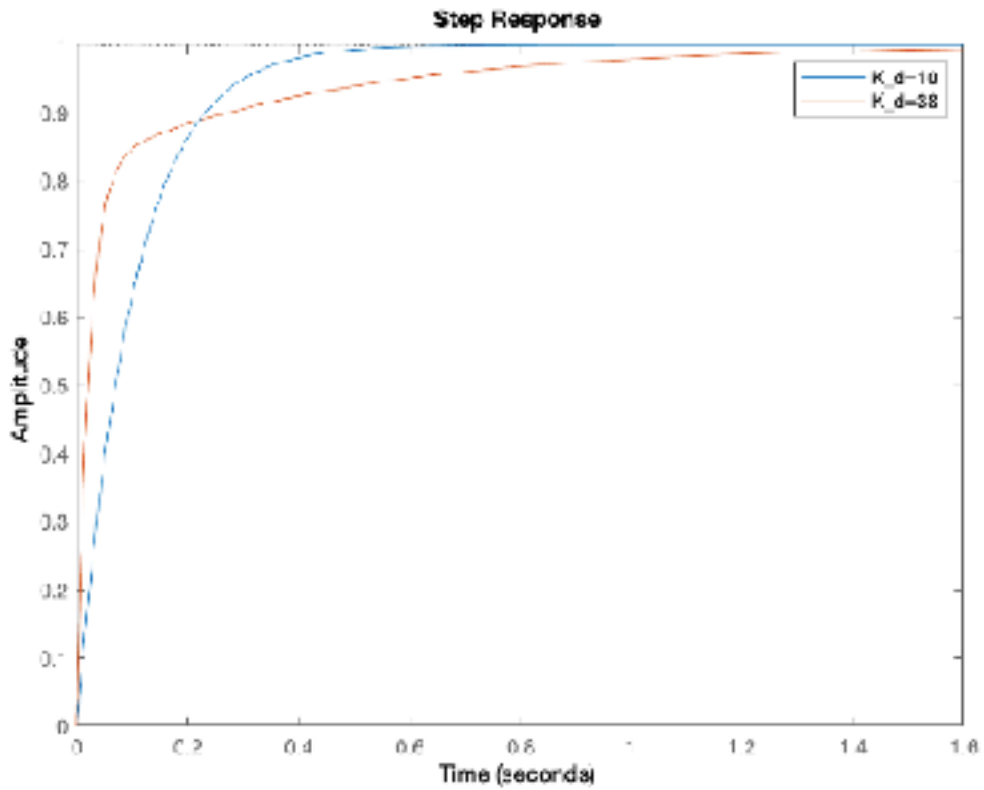


1. Projeto de controladores PD

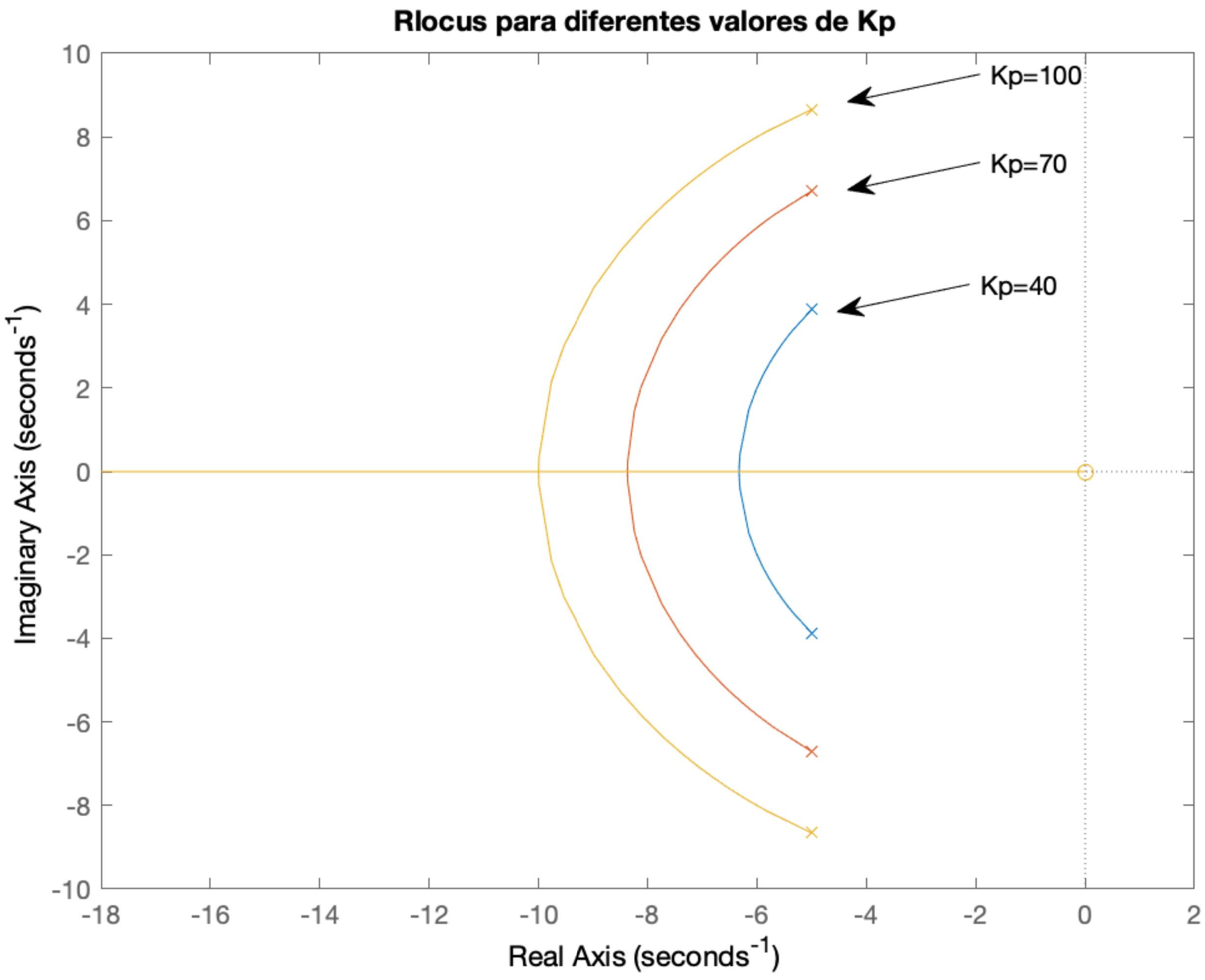
Zero do PD=Kp/Kd=100/10=10

Cancelou o polo em s=-10!!





1. Projeto de controladores PD



Para o caso de haver liberdade de escolha de Kp

Projeto de controladores PD

A segunda alternativa de projeto do PD é fazê-lo diretamente no rltool!

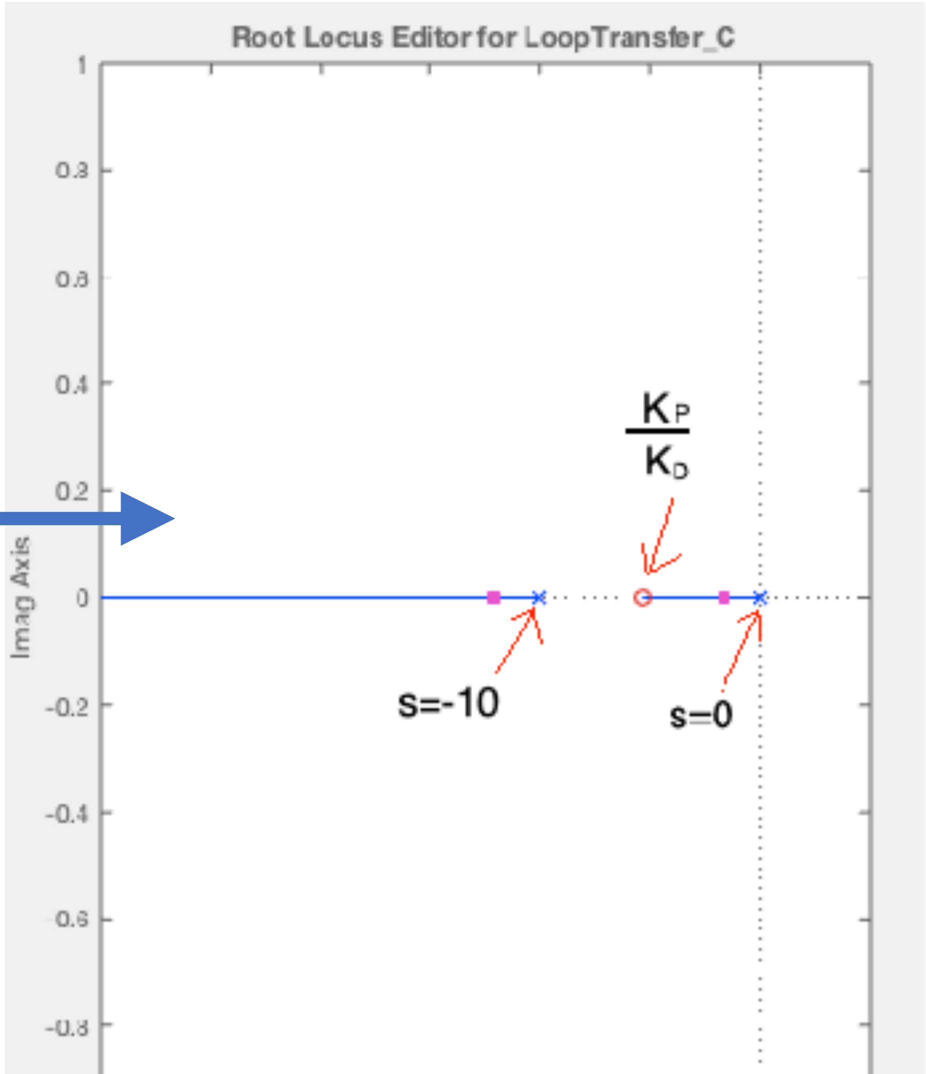
Adiciona-se o zero do PD e ajusta-se o ganho K_D para a resposta desejada, enquanto $\dfrac{K_P}{K_D}$ fica fixo.

$$G(s) = \frac{K_P + K_D s}{s(s+10)}$$

rltool(g)

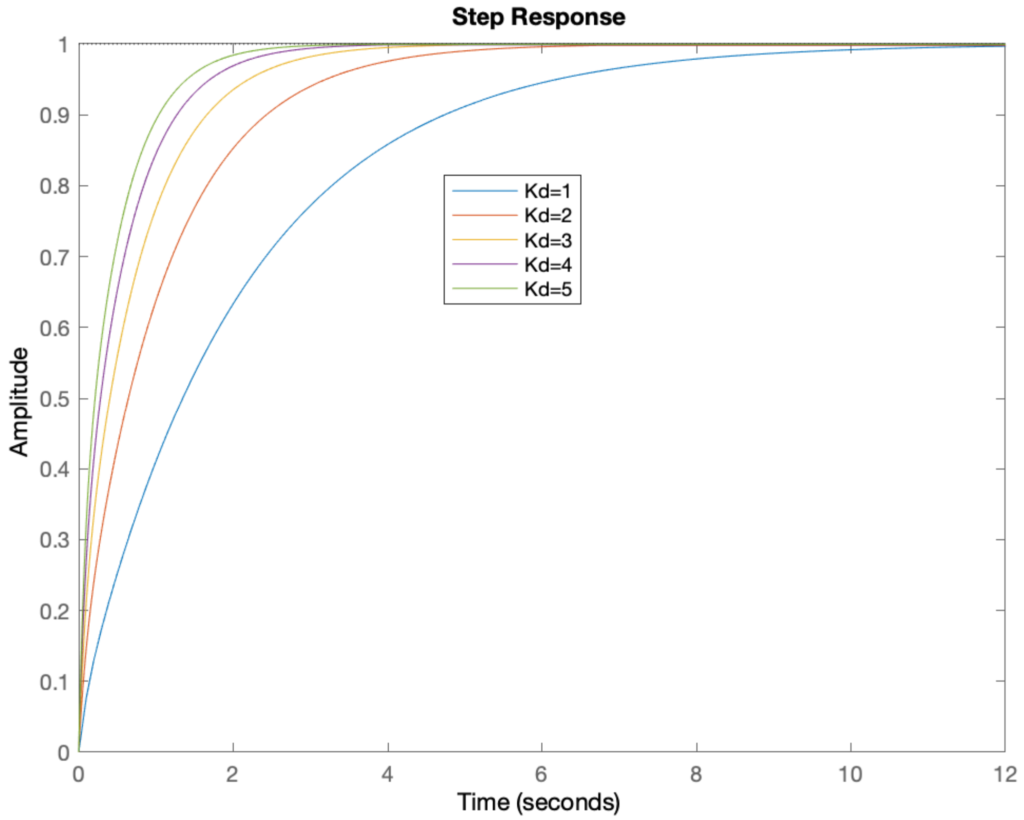
$$G(s) = \frac{K_D(s + K_P/K_D)}{s(s+10)}$$

O zero do PD é adicionado após o rltool abrir



Projeto de controladores PD

LR variando K_D de 1 a 8, com $K_P/K_D = 5.$

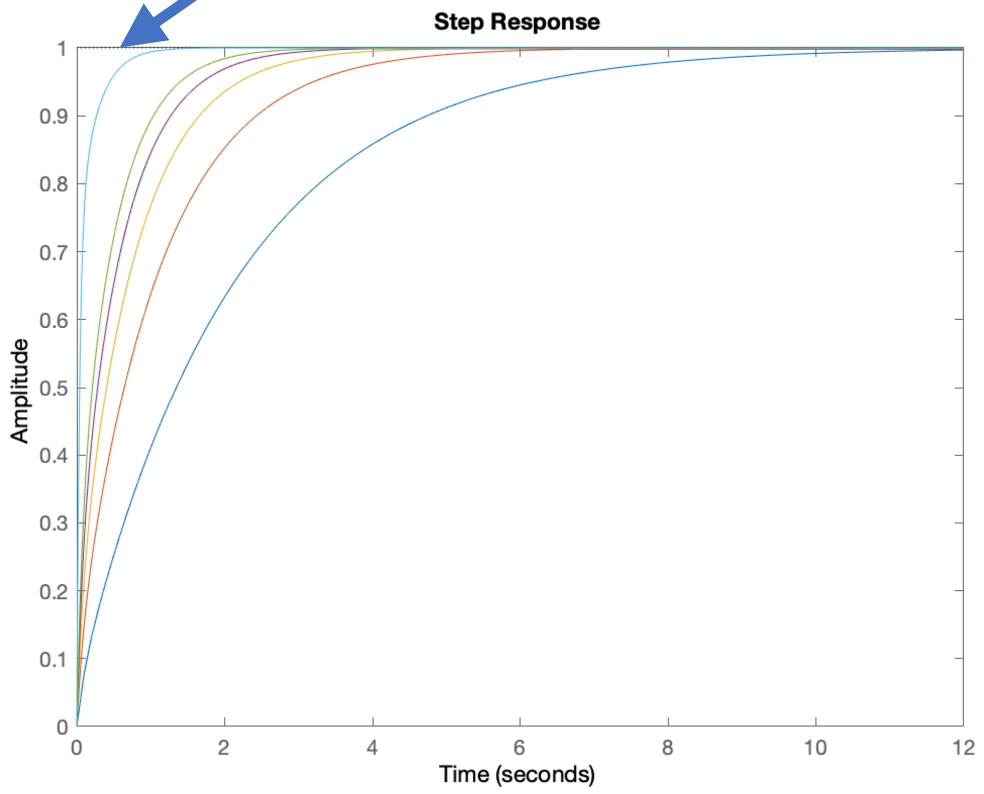


Lembrando que, para o exemplo dado, deve-se ter $K_P \geq 100$,logo escolhe-se $K_D = 20$, de modo que $K_P = 5K_D$ atenda a condição de erro em regime!

Projeto de controladores PD

$K_D = 20, K_P = 100$

Resposta rápida, sem sobreelevação, atendendo condição de erro em regime.



Projeto de controladores PD

É importante observar a diferença das estratégias de projeto.

Neste caso, escolhe-se a localização do zero do PD $s = -\dfrac{K_P}{K_D}$ e varia-se K_D no LR.

No caso anterior, para um dado ganho K_P fixo, escolhia-se no LR um valor de K_D.

Portanto, nos dois casos o valor de Kp fica fixo enquanto o valor de Kd varia.

2. Projeto do controlador PI

A FT deste controlador é dado por

$$G_c(s) = K_p + \frac{K_i}{s} \qquad (5)$$

ou

$$G_c(s) = K_p \frac{(s + \frac{K_i}{K_p})}{s} \qquad (6)$$

ou

$$G_c(s) = K_i \frac{(1 + \frac{K_p}{K_i} s)}{s} \qquad (7)$$

Este controlador adiciona um polo na origem, aumentando assim o tipo do sistema, contribuindo para atender o erro em regime.

Como o ganho do controlador (de (7)) depende de Ki, este é o parâmetro que terá um valor mínimo para atender o erro em regime.

2. Projeto do controlador PI

Usando a FT

$$C(s) = K_p \frac{\left(s + \frac{K_I}{K_p}\right)}{s}$$

Observa-se que o controlador PI adiciona um polo na origem e um zero.

O projeto do PI via LR envolve escolher a localização do zero e o ganho Kp. Depois disto, calcula-se o ganho KI.

2. Projeto do controlador PI

Exemplo: Seja G(s) dada por

$$G(s) = \frac{10}{(s+1)(s+5)}$$

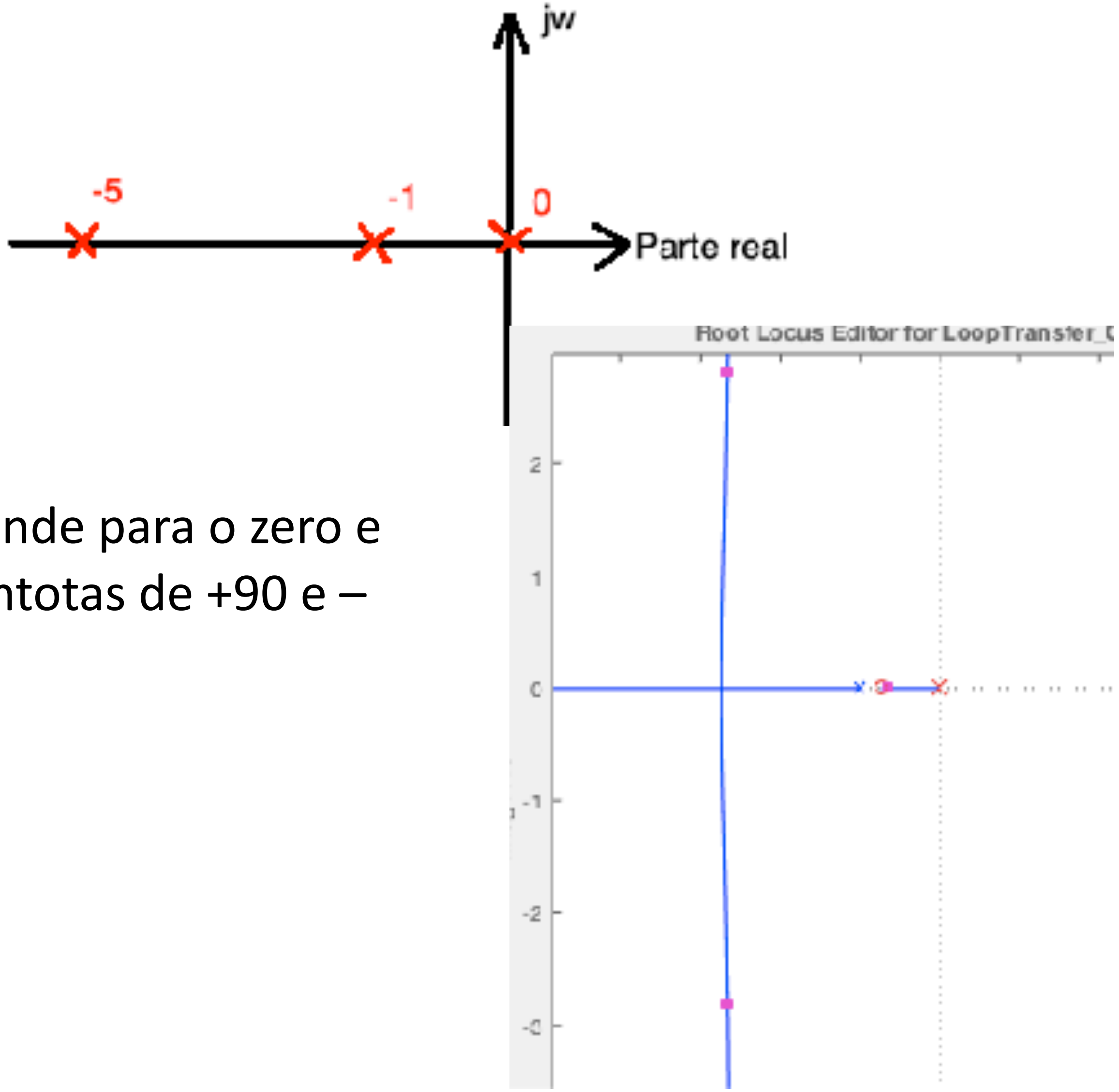
Com o controlador PI, resulta

$$C(s)G(s) = \frac{K_p\left(s + \frac{K_I}{K_p}\right)10}{s(s+1)(s+5)}$$

O formato do LR vai depender da localizacão do zero $s = -\dfrac{K_I}{K_P}$.

2. Projeto do controlador PI

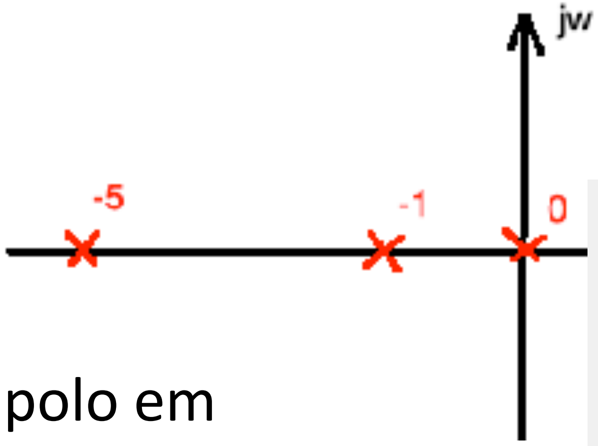
$$C(s)G(s) = \frac{K_p\left(s+\frac{K_I}{K_p}\right)10}{s(s+1)(s+5)}$$

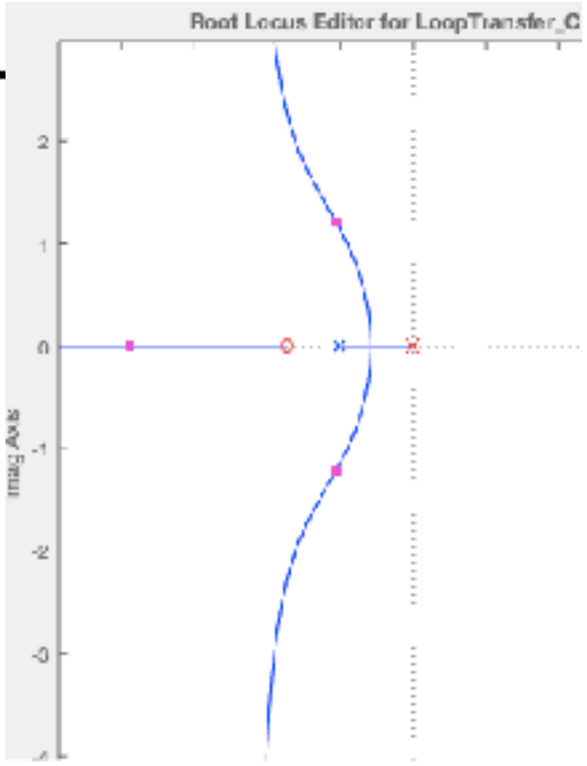


Zero entre 0 e –1: O polo na origem tende para o zero e os polos em –1 e –5 tendem para assintotas de +90 e –90 graus

2. Projeto do controlador PI

$$C(s)G(s) = \frac{K_p\left(s + \frac{K_I}{K_p}\right)10}{s(s+1)(s+5)}$$





Zero entre -1 e –5: O polo na origem e o polo em –1 tendem para assíntotas de +90 e –90 graus.

O polo em –5 tende para o zero (ver onde há LR para K>0)

A interseção das assíntotas é em -6+(zero PI)/2.
O zero mais longe da origem aproxima os dois polos do eixo jw, menor amortecimento!!!

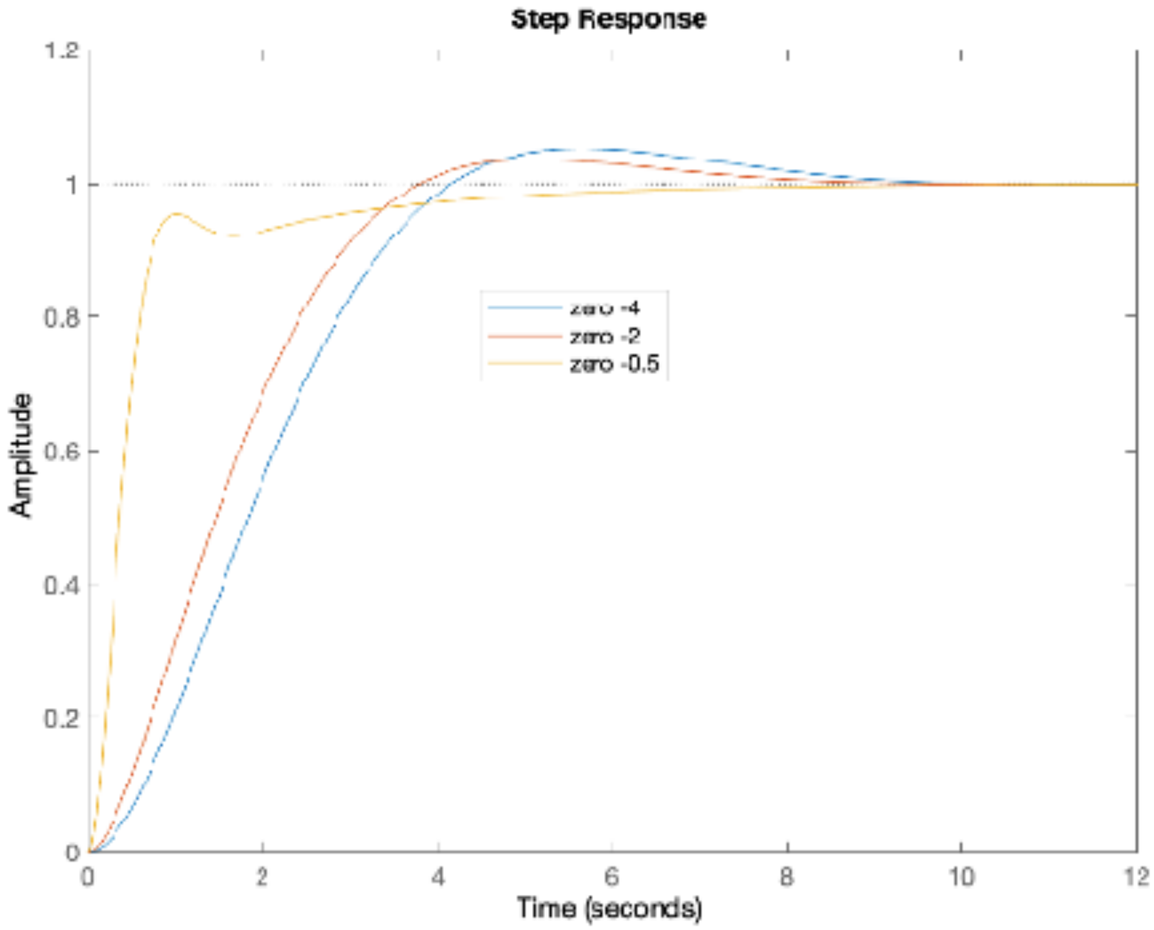
2. Projeto do controlador PI

Resposta para três localizações do zero, aumentando o Kp:

$$\frac{K_I}{K_p} = -4 \quad K_p = 0.28$$

$$\frac{K_I}{K_p} = -2 \quad K_p = 0.32$$

$$\frac{K_I}{K_p} = -0.5 \quad K_p = 0.8$$



2. Projeto do controlador PI

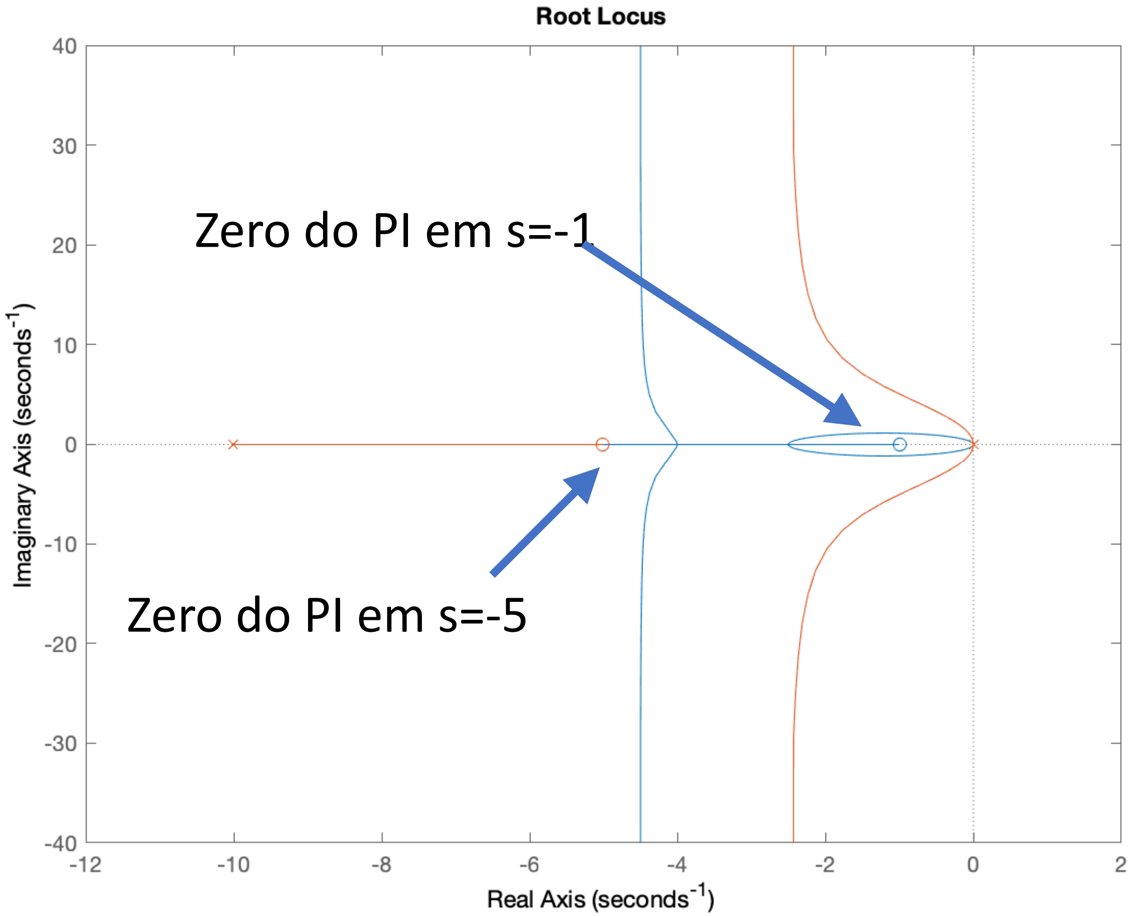
Resumindo o projeto do PI via LR:

- O zero do PI próximo à origem tende a deixar a resposta lenta
- O zero do PI muito longe da origem tende a deixar a resposta menos amortecida (maior sobreelevação)
- O ganho Kp deve ser aumentado de modo a conseguir respostas mais rápidas.

2. Projeto do controlador PI

Exemplo de projeto: analisar melhoria da estabilidade relativa com um PI aplicado a

$$G(s) = \frac{1}{s(s+10)}$$



3. Projeto do controlador PID

O controlador PID na forma paralela é dado por

$$C(s) = K_P + \frac{K_I}{s} + K_D s$$

que pode ser escrito como

$$C(s) = (1 + K_{D1}s)(K_{P1} + \frac{K_{I1}}{s}) = K_{P1}(1 + K_{D1}s)(1 + \frac{K_{I1}}{K_{P1}s})$$

$$C(s) = (1 + K_{D1}s)(K_{P1} + \frac{K_{I1}}{s}) = \frac{K_D s^2 + K_P s + K_I}{s}$$

com

$$K_D = K_{D1}K_{P1}$$

$$K_I = K_{I1}$$

$$K_P = K_{P1} + K_{D1}K_{I1}$$

3. Projeto do controlador PID

$$C(s) = (1 + K_{D1}s)(K_{P1} + \frac{K_{I1}}{s}) = \frac{K_D s^2 + K_P s + K_I}{s}$$

Com este arranjo, pode-se fazer o projeto do PI na forma $C_{PI} = K_{P1} + \frac{K_{I1}}{s}$

da forma usual, corrigindo o erro em regime, e aumentando o ganho proporcional K_{P1} de forma a obter uma resposta rápida, mesmo que com sobreelevação.

O controlador PD é então adicionado para melhorar a estabilidade relativa, reduzindo a sobreelevação.

Exemplo de projeto PID:

$$G(s) = \frac{4}{(s+1)(s+5)}$$

Especificações:

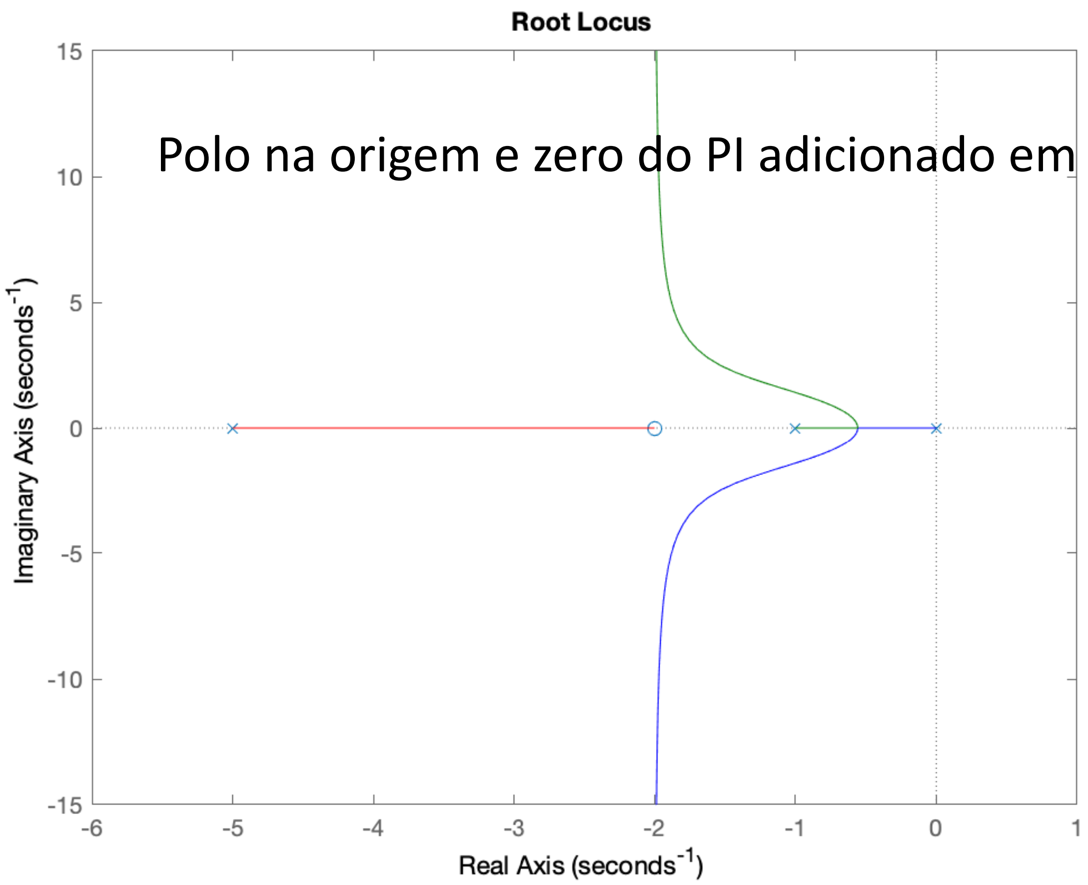
- Erro nulo ao degrau
- Tempo de estabelecimento < 2s
- Sobreelevação UP<10%

Exemplo de projeto PID:

$$G(s) = \frac{4}{(s+1)(s+5)}$$

Especificações:

- Erro nulo ao degrau
- Tempo de estabelecimento < 2s
- Sobreelevação UP<10%



Polo na origem e zero do PI adicionado em s=-2

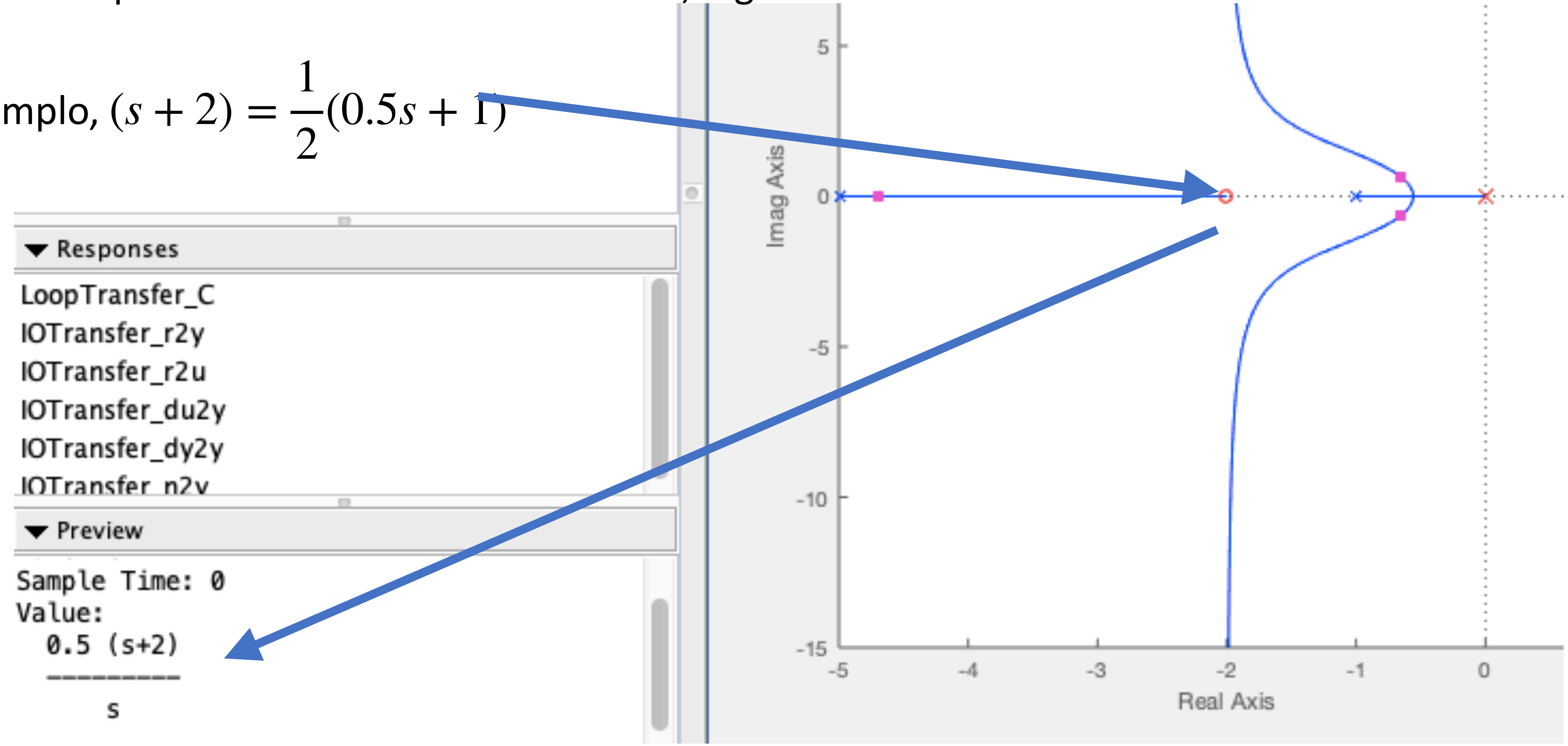
Exemplo de projeto PID: IMPORTANTE

Foram adicionados o integrador e o zero do PI em s=-2

$$G(s) = \frac{4}{(s+1)(s+5)}$$

Quando um zero ou polo é adicionado ao LR no rltool, o ganho do controlador é mantido o mesmo.

No exemplo, $(s+2) = \dfrac{1}{2}(0.5s+1)$

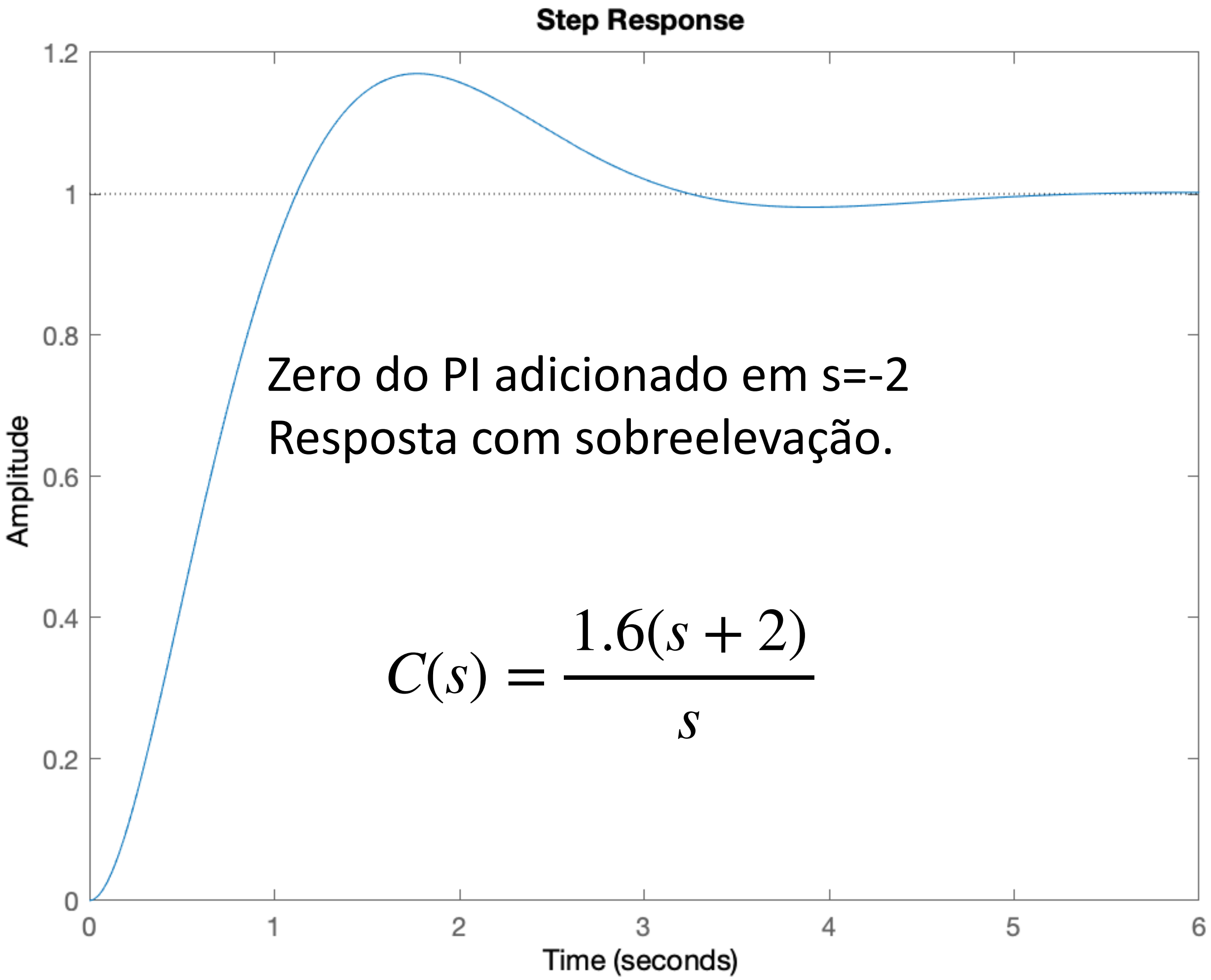


Exemplo de projeto PID:

$$G(s) = \frac{4}{(s+1)(s+5)}$$

Especificações:

- Erro nulo ao degrau
- Tempo de estabelecimento < 2s
- Sobreelevação UP<10%



Zero do PI adicionado em s=-2
Resposta com sobreelevação.

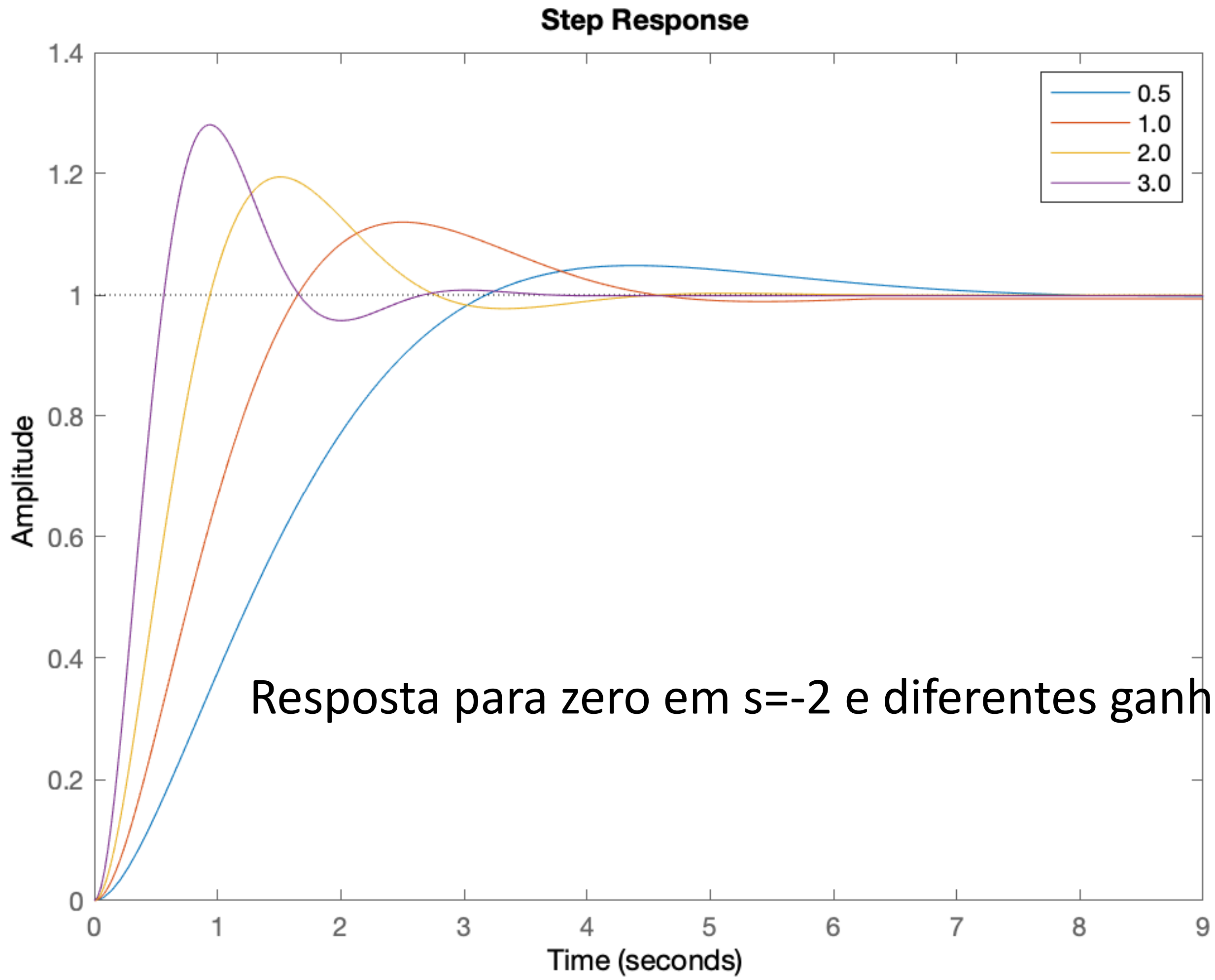
$$C(s) = \frac{1.6(s+2)}{s}$$

Exemplo de projeto PID:

$$G(s) = \frac{4}{(s+1)(s+5)}$$

Especificações:

- Erro nulo ao degrau
- Tempo de estabelecimento < 2s
- Sobreelevação UP<10%



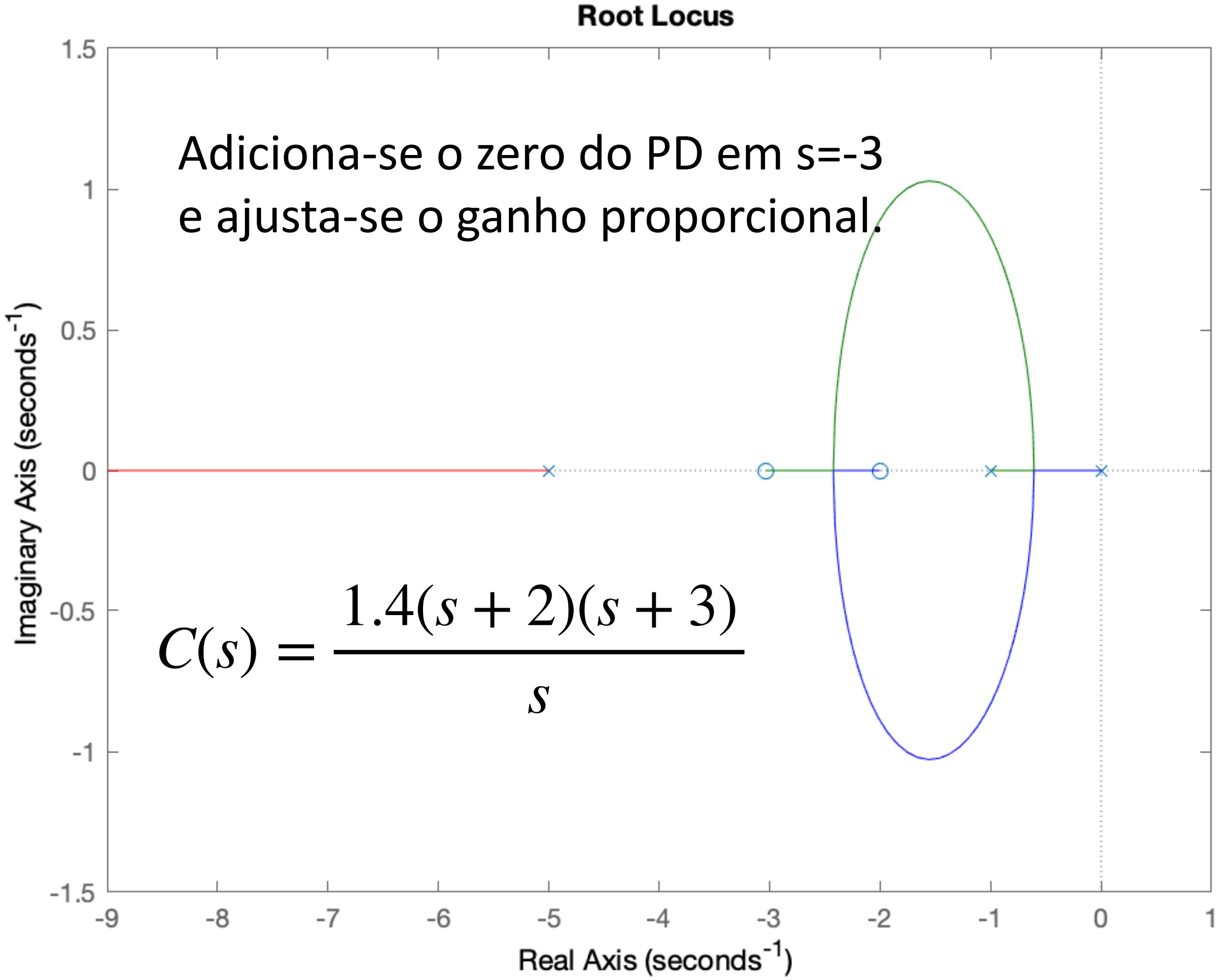
Resposta para zero em s=-2 e diferentes ganhos.

Exemplo de projeto PID:

$$G(s) = \frac{4}{(s+1)(s+5)}$$

Especificações:

- Erro nulo ao degrau
- Tempo de estabelecimento < 2s
- Sobreelevação UP<10%



Adiciona-se o zero do PD em s=-3
e ajusta-se o ganho proporcional.

$$C(s) = \frac{1.4(s+2)(s+3)}{s}$$

Exemplo de projeto PID:

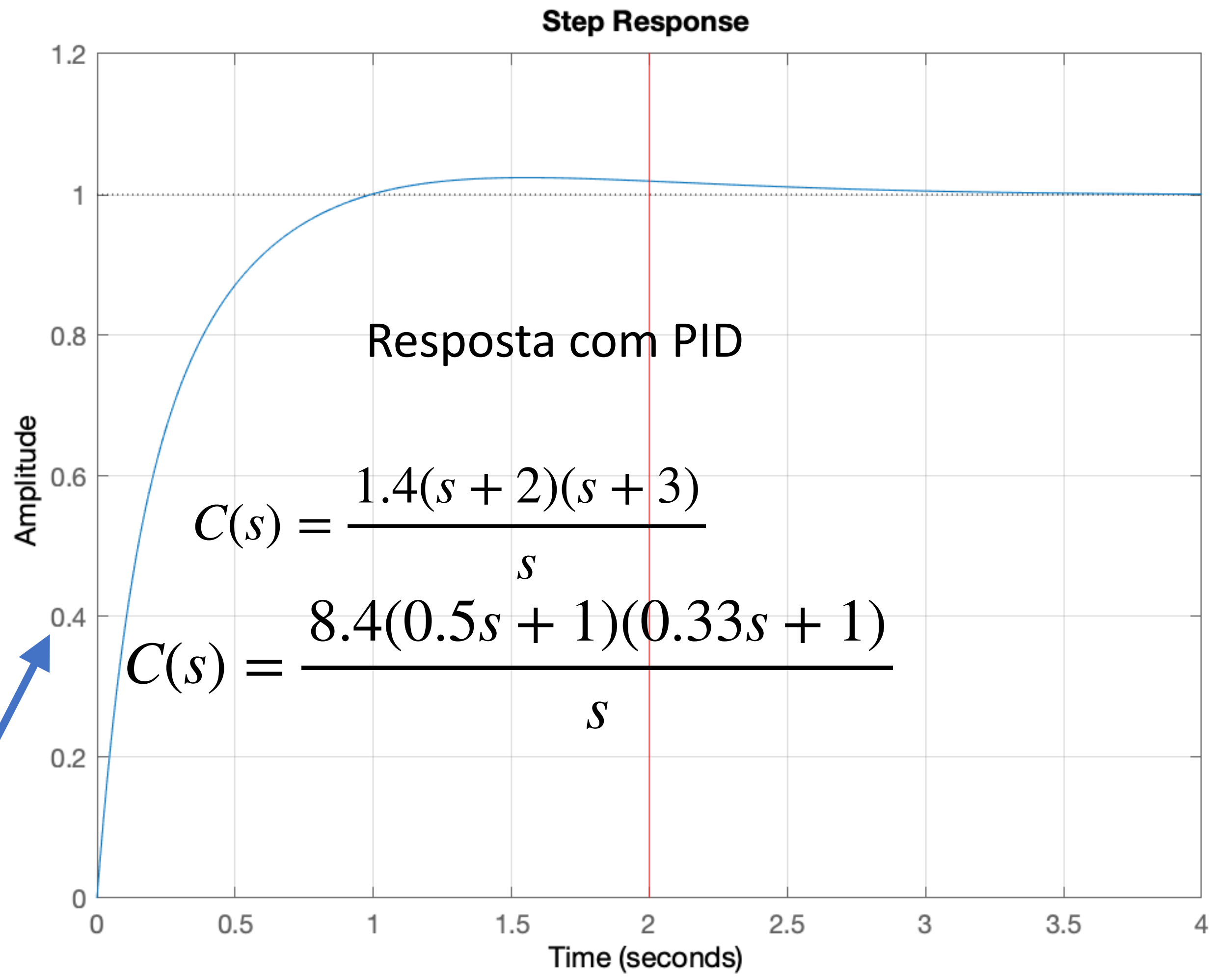
$$G(s) = \frac{4}{(s+1)(s+5)}$$

Especificações:

- Erro nulo ao degrau
- Tempo de estabelecimento < 2s
- Sobreelevação UP<10%

$$C(s) = \frac{1.4s^2 + 7s + 8.4}{s}$$

$$C(s) = K_P + \frac{K_I}{s} + K_D s = \frac{K_D s^2 + K_P s + K_I}{s}$$



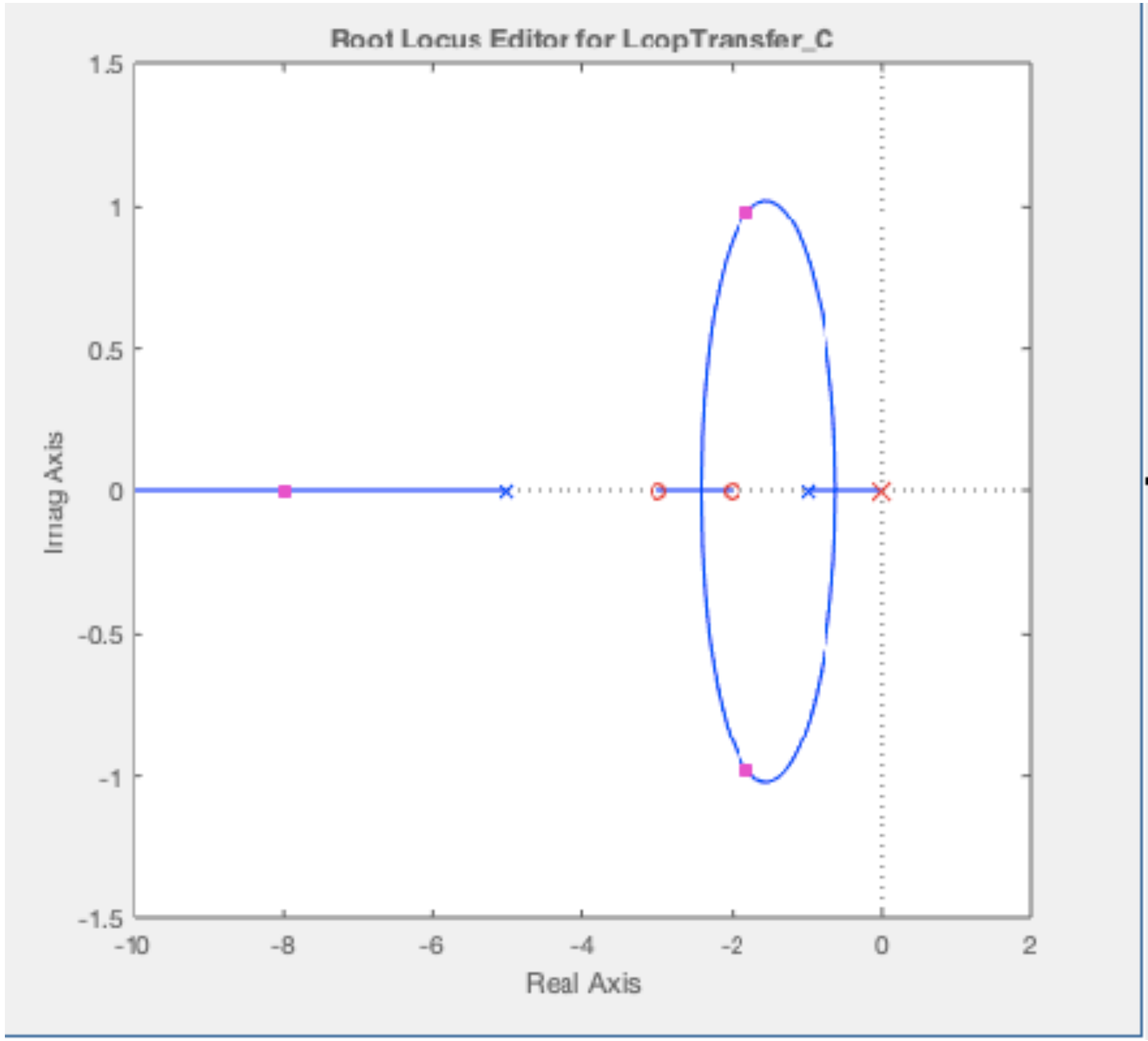
$$C(s) = \frac{1.4(s+2)(s+3)}{s}$$

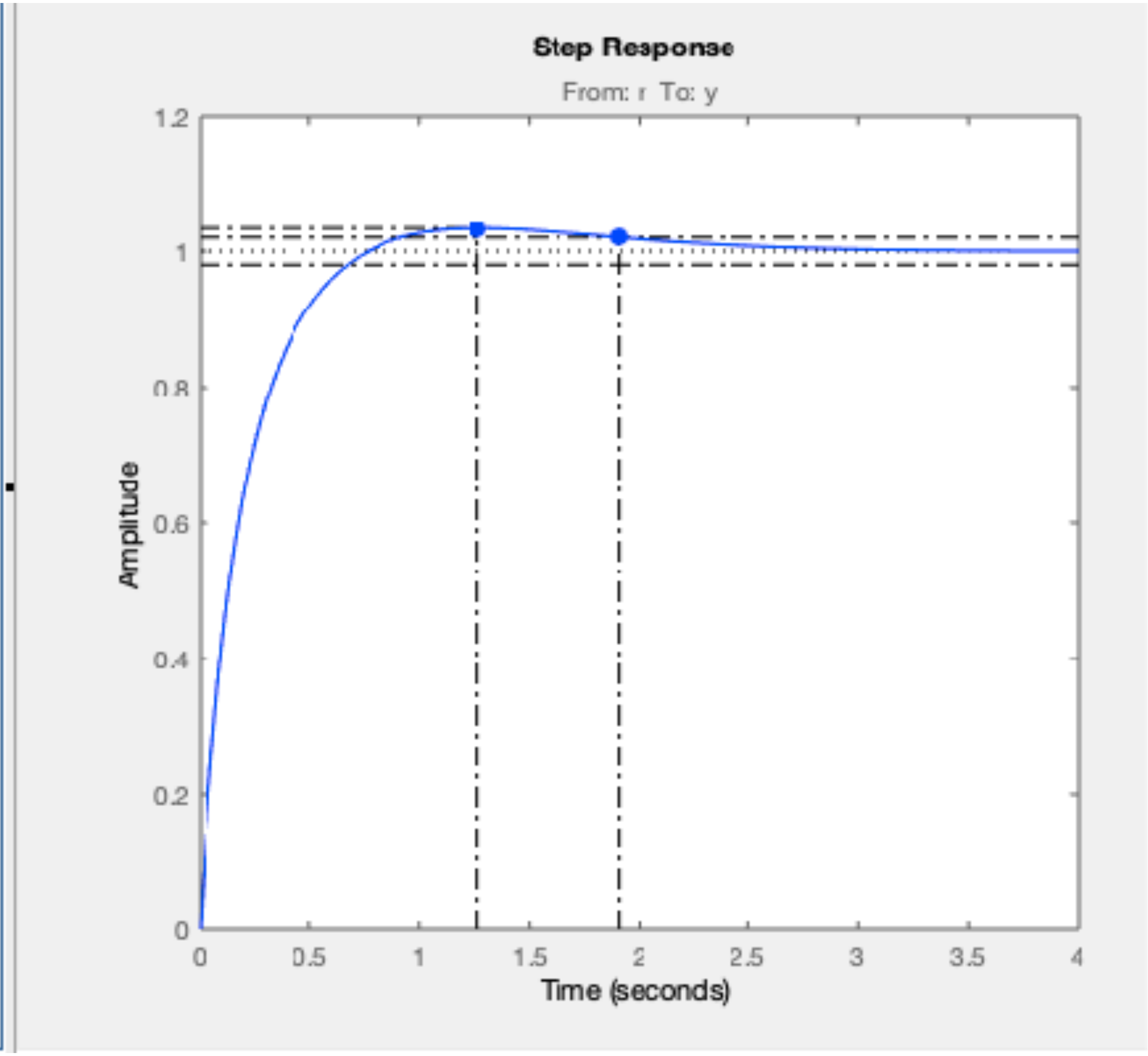
$$C(s) = \frac{8.4(0.5s+1)(0.33s+1)}{s}$$

step(feedback(C*g,1))

Exemplo de projeto PID:

Ao adicionar o zero do PD no rltool, a resposta ao degrau mostrada corresponde ao sistem em malha fechada com o PID.





Exemplo de projeto PID:

O zero do PD foi adicionado em s=-3. Regiões próximas podem ser exploradas.
A cada ajuste do zero, pode-se variar o ganho procurando melhorar a resposta.

Aproximar o zero do PD da origem aumenta o efeito derivativo, e permite assim aumentar o ganho proporcional, que torna a resposta mais rápida.

Simulação no Matlab:

```
c=tf(8.4*conv([1/3 1],[1/2 1]), [1  0])
m=feedback(c*g,1)
```

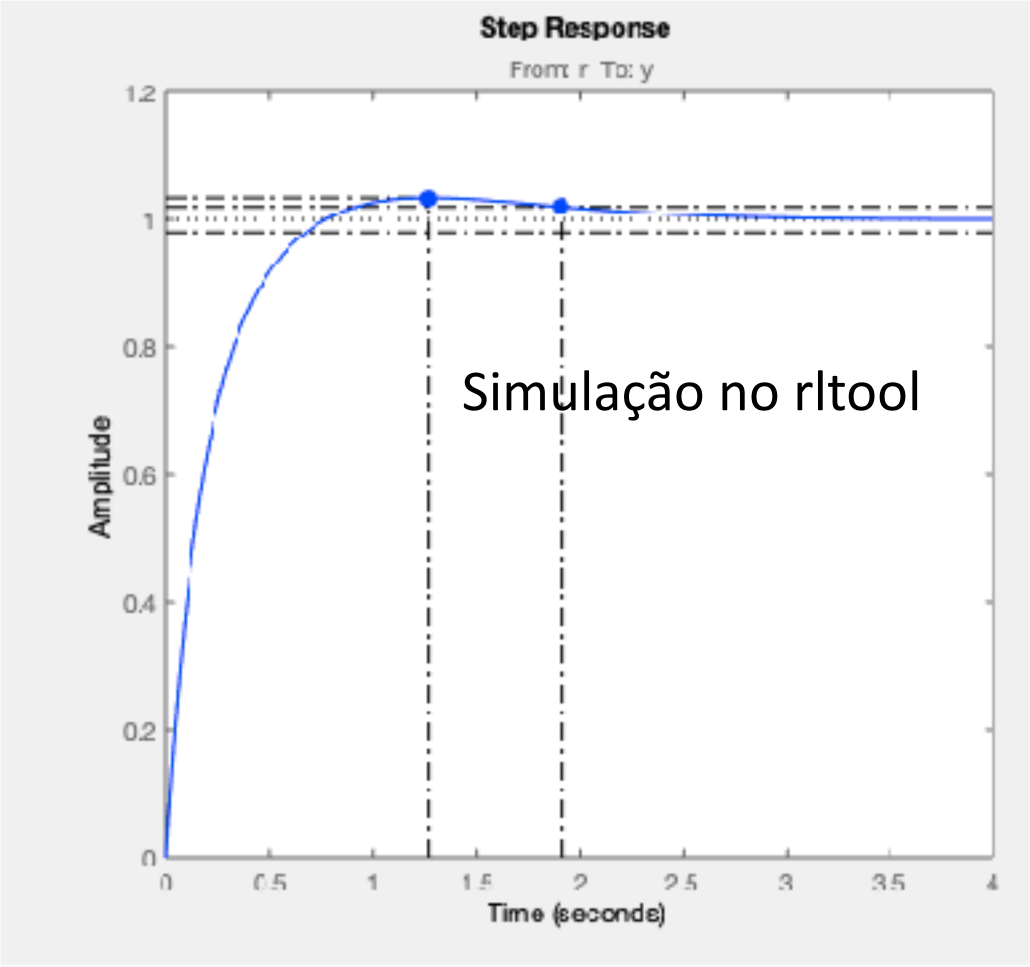
ou

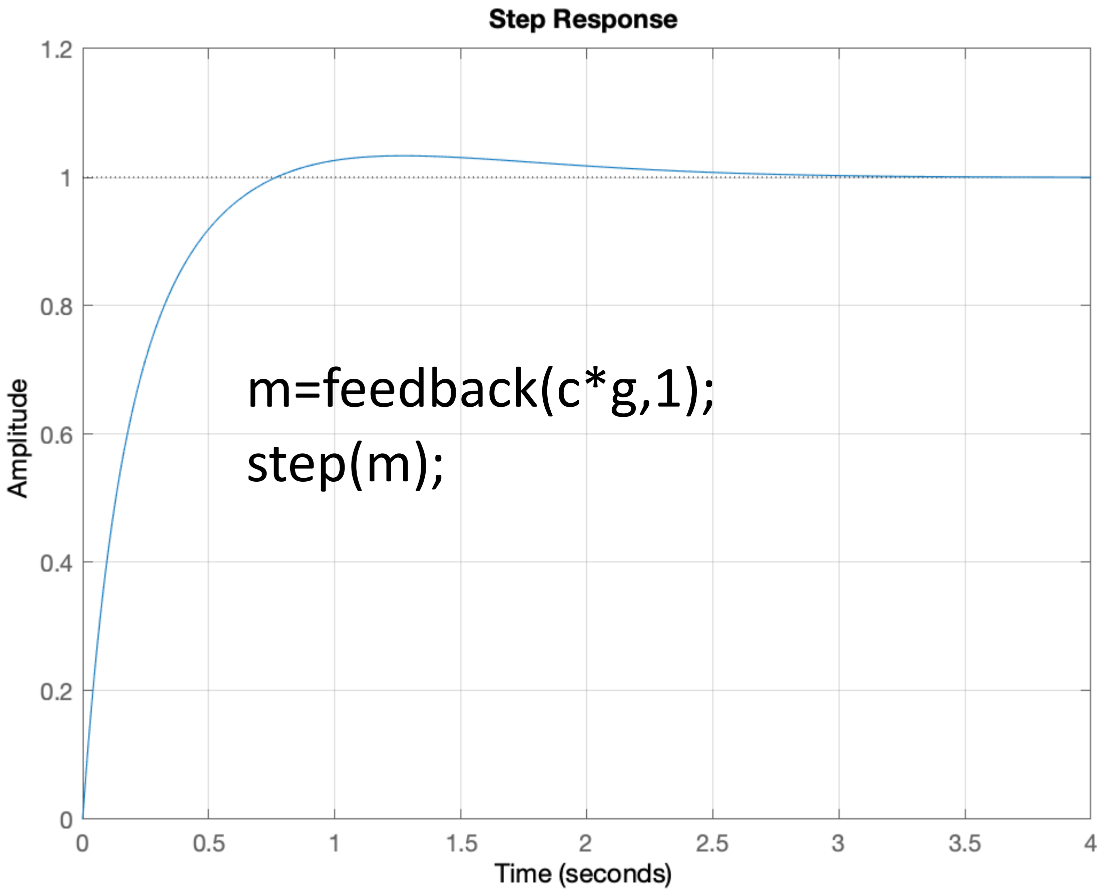
```
c=pid(7, 8.4, 1.4)
m=feedback(c*g,1)
```

$$C(s) = \frac{1.4s^2 + 7s + 8.4}{s}$$

$$C(s) = K_P + \frac{K_I}{s} + K_D s = \frac{K_D s^2 + K_P s + K_I}{s}$$

Exemplo de projeto PID:





Exemplo de projeto PID:

Caso o controlador PID seja implementado em um controlador digital (PLC, Microcontrolador, etc), os ganhos Kp, Ki e Kd devem ser informados. Deve-se ter o cuidado de saber se o controlador é implementado na forma série (mais comum) ou na forma paralela.

Caso seja feita apenas a simulação, basta ter-se a função de transferência do controlador, não importanto a forma série ou paralela.

Comparação com o método de síntese direta

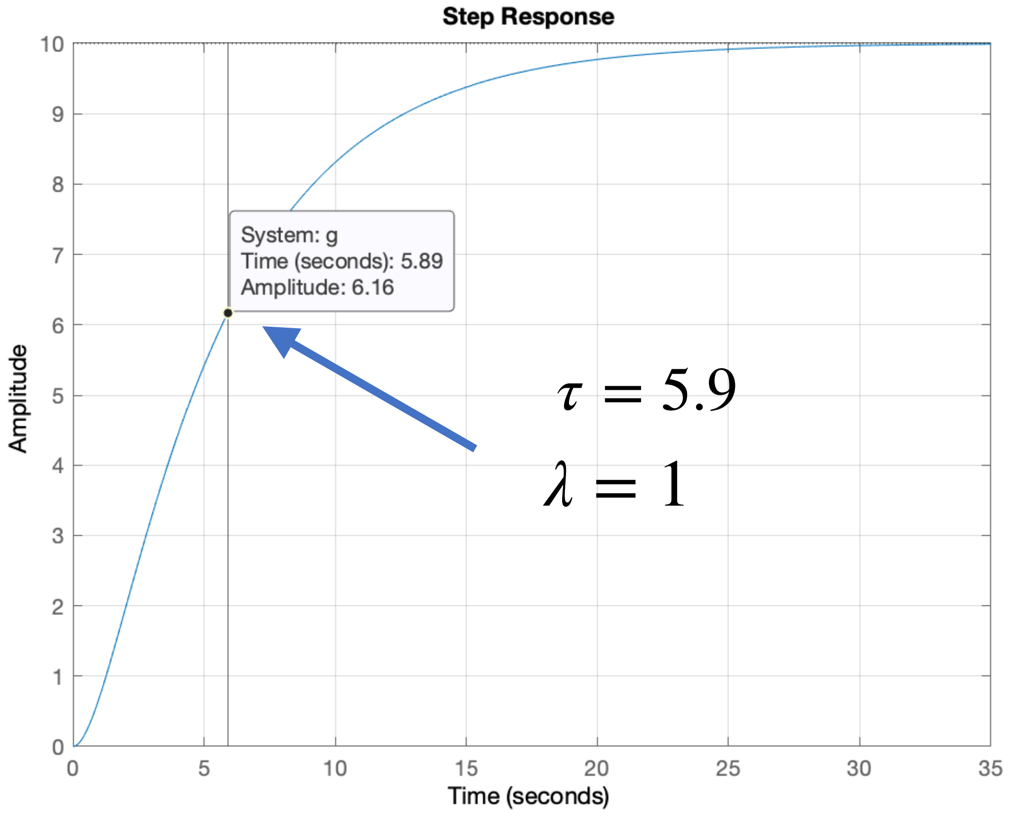
Faremos a comparação dos projetos usando o modelo abaixo.

Seja

$$G(s) = \frac{K}{(\tau_1 s + 1)(\tau_2 s + 1)} = \frac{10}{(s+1)(5s+1)}$$

A resposta desse sistema ao degrau é mostrada ao lado. Ela pode ser aproximada por um sistema de primeira ordem com constante tempo de 5.9 segundos.

Usaremos o modelo de referência $M(s) = \dfrac{1}{s+1}$, ou seja, constante de tempo de malha fechada $\lambda = 1$.



Comparação com o método de síntese direta

Abaixo as expressões para calcular os ganhos:

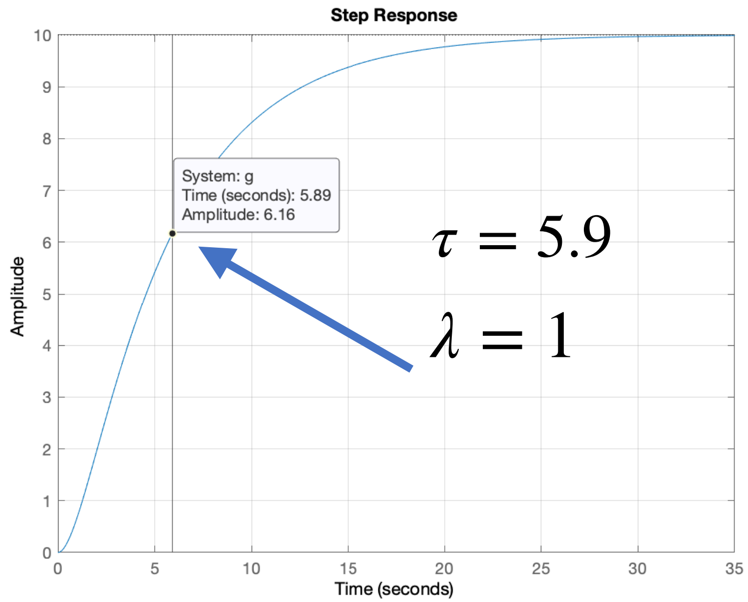
$$\text{Seja } G(s) = \frac{K}{(\tau_1 s + 1)(\tau_2 s + 1)} = \frac{10}{(s+1)(5s+1)}$$

$$\boxed{\begin{aligned} K_P &= \frac{\tau_1 + \tau_2}{\lambda K} \\ K_I &= \frac{1}{T_I} = \frac{1}{\tau_1 + \tau_2} \\ K_D &= T_D = \frac{\tau_1 \tau_2}{\tau_1 + \tau_2} \end{aligned}}$$

$$K_P = \frac{1+5}{\lambda 10}$$

$$K_I = \frac{1}{6}$$

$$K_D = \frac{5}{6}$$



$$\tau = 5.9$$

$$\lambda = 1$$

$$C(s) = K_P\left(1 + \frac{K_I}{s} + K_D s\right) = \frac{K_P K_D s^2 + K_P s + K_P K_I}{s}$$

$$C(s) = \frac{0.5s^2 + 0.6s + 0.1}{s}$$

Comparação com o método de síntese direta

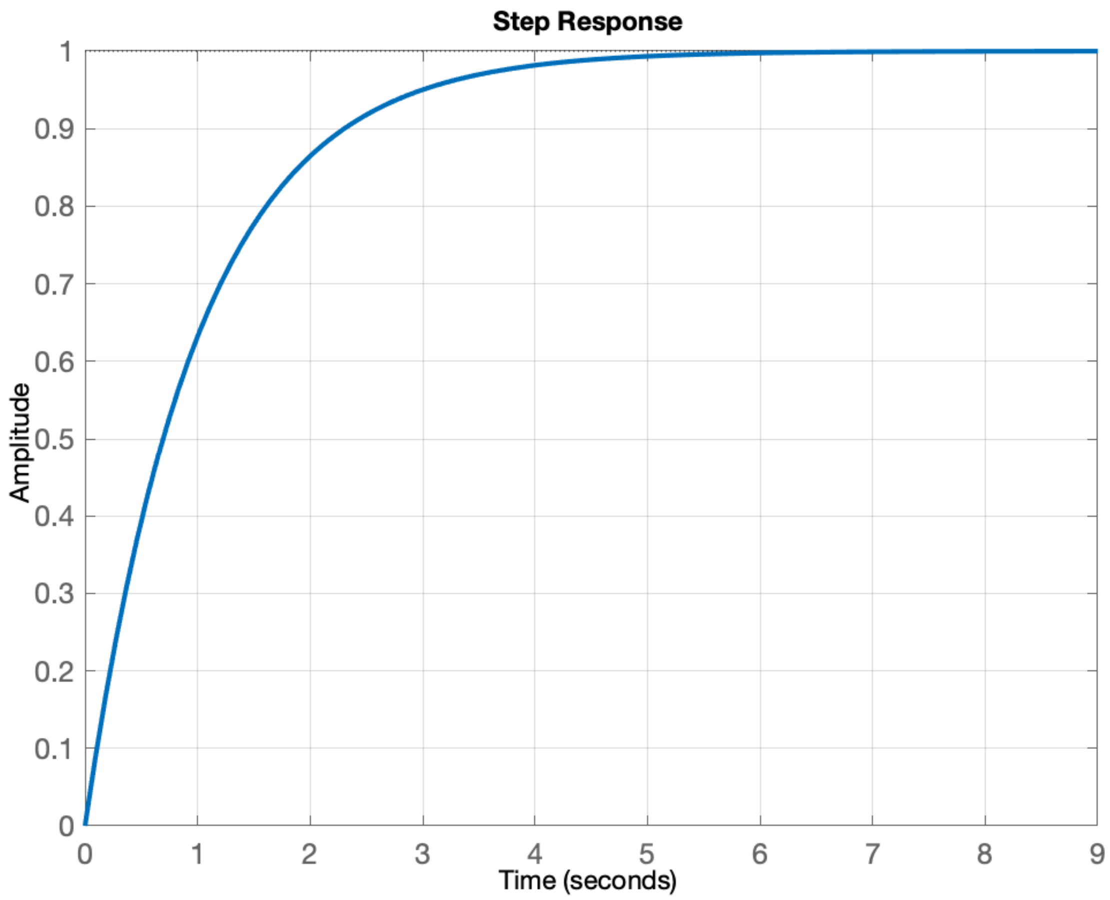
$$C(s) = \frac{0.5s^2 + 0.6s + 0.1}{s}$$

Resposta igual ao modelo de referência,

$$T(s) = \frac{1}{s+1}, \quad \text{pois:}$$

$$C(s)G(s) = \frac{(5s+1)(s+1)}{10s}\frac{10}{(5s+1)(s+1)}$$

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



Os zeros do PID cancelam os polos de G(s)!

Comparação com o método de síntese direta

O projeto equivalente via LR seria:

$$1 + KC(s)G(s) = 1 + K\frac{(5s+1)(s+1)}{s}\frac{10}{(5s+1)(s+1)} = 1 + K\frac{10}{s}$$

Via LR, há liberdade completa sobre onde alocar os dois zeros do PID.

No projeto via síntese direta (ou IMC), abre-se mão desta liberdade para simplificar o procedimento de projeto.

Exercício 1: projeto do PD

Seja $G(s) = \frac{s+1}{s^3}$.

Verifique se este sistema pode ser estabilizado através de um controlador PD,

e obtenha os ganhos K_p e K_d para uma boa resposta.

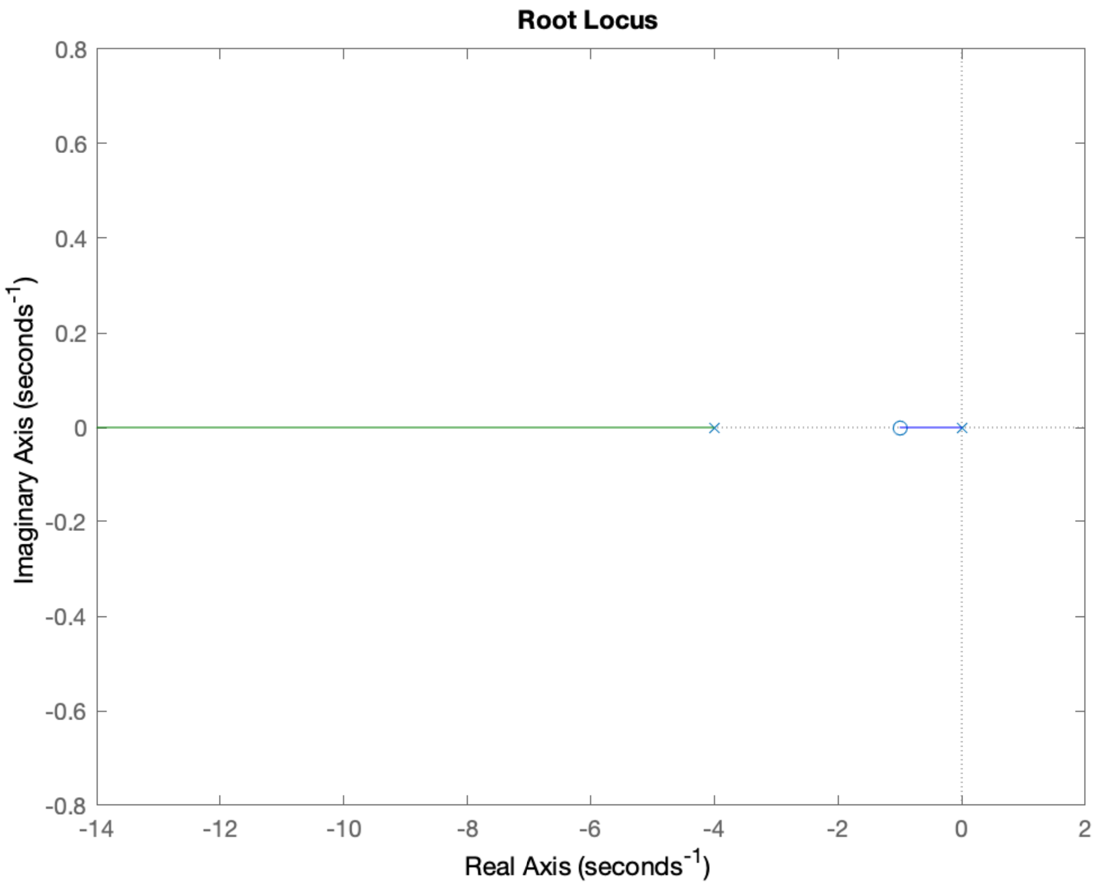
Basta fazer o LR e a resposta aparece!

Exercício 2: projeto do PI

Seja $G(s) = \dfrac{1}{s+4}$.

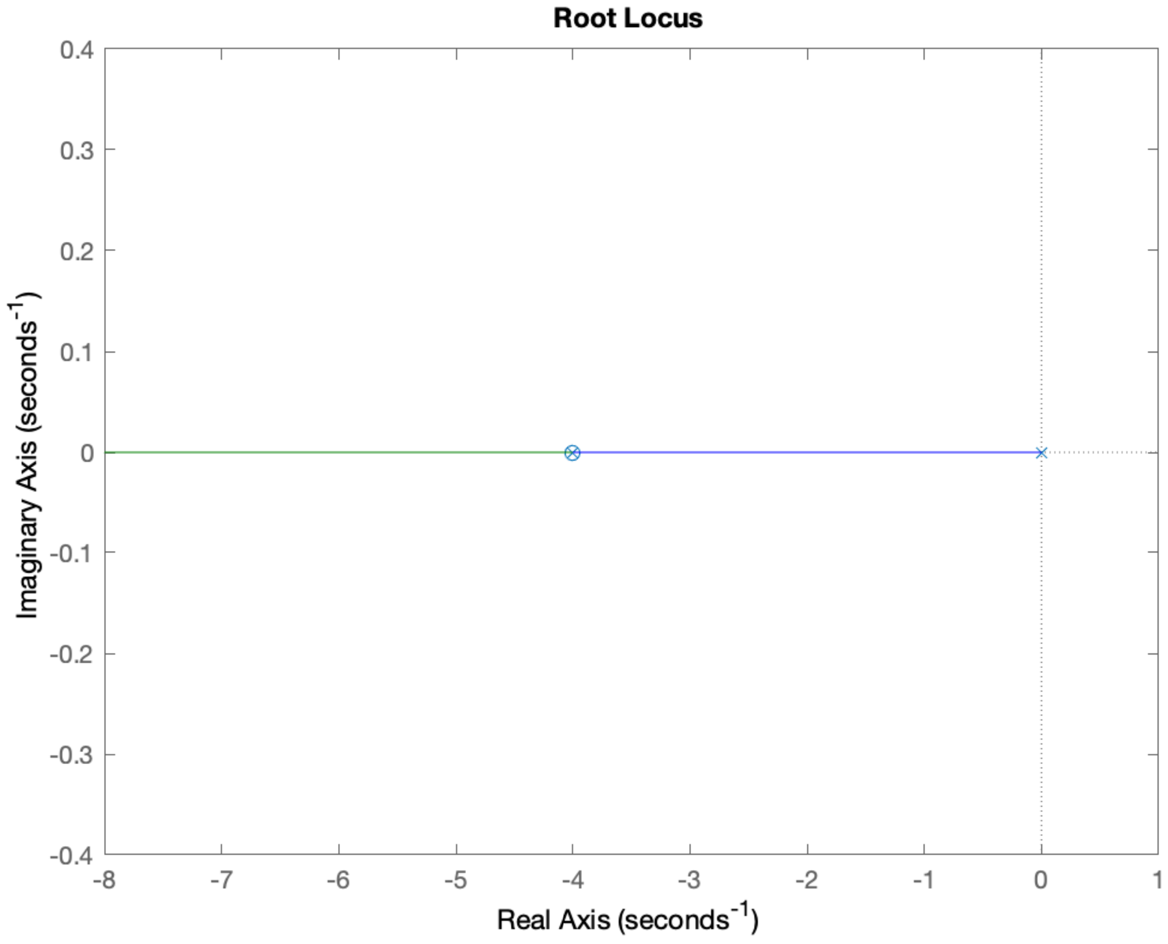
Obtenha os ganhos de um controlador PI de modo que se tenha em malha fechada $\zeta \geq 0.7$ e tempo de estabelecimento < 1 segundo.

Exercício 2: projeto do PI



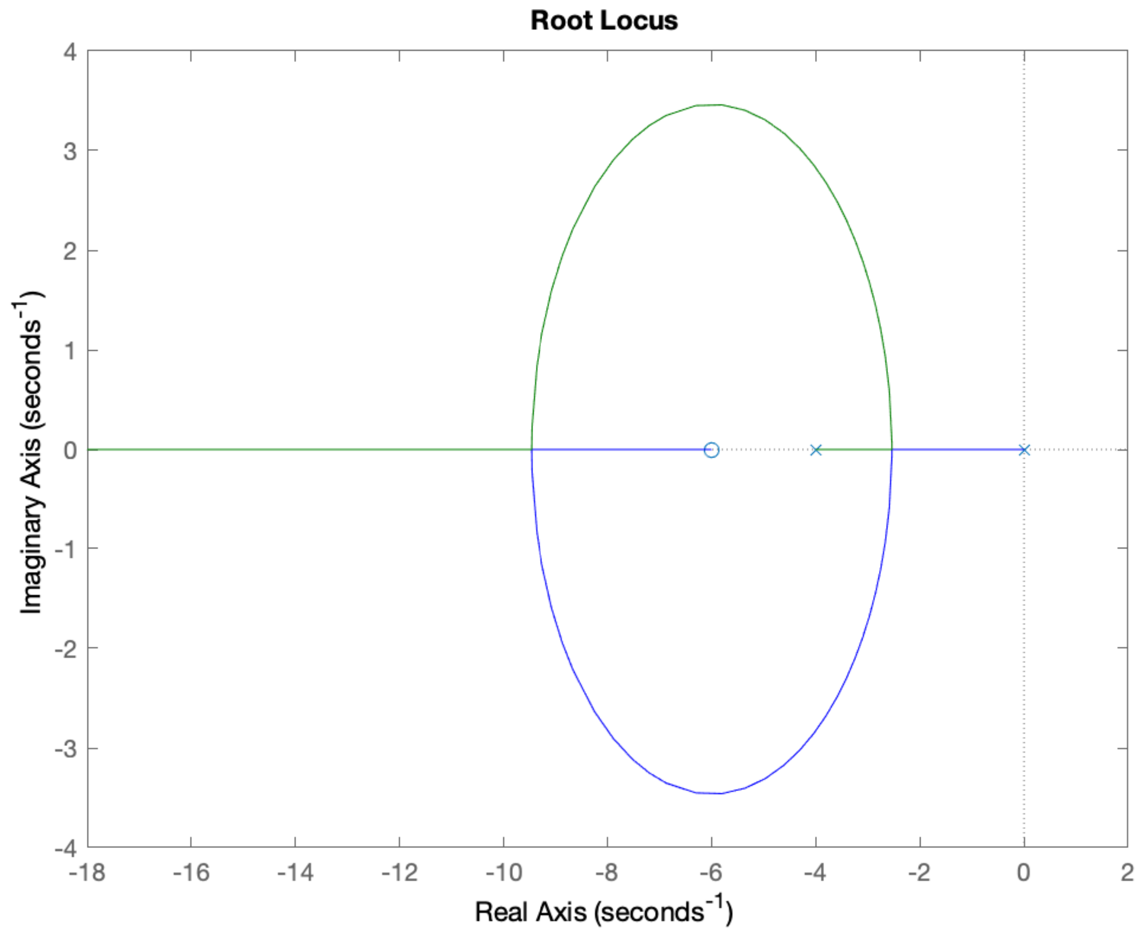
$$C(s)G(s) = K_p \frac{s+1}{s} \frac{1}{s+4}$$

Exercício 2: projeto do PI



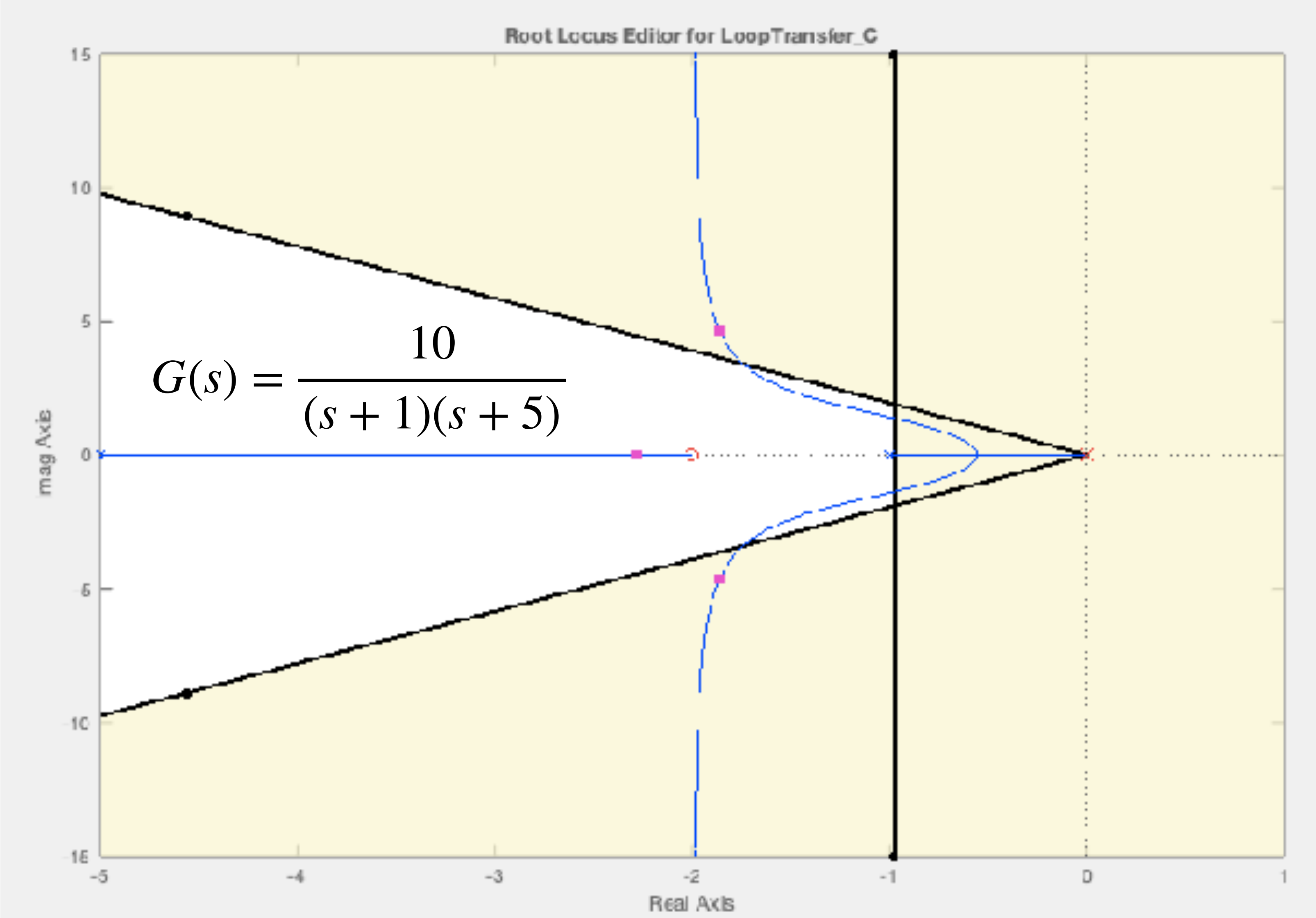
$$C(s)G(s) = K_p \frac{s+4}{s} \frac{1}{s+4}$$

Exercício 2: projeto do PI



$$C(s)G(s) = K_p \frac{s+6}{s} \frac{1}{s+4}$$

Exercício 3: projeto do PI



$$G(s) = \frac{10}{(s+1)(s+5)}$$

Obtenha os valores Kp e Ki de um controlador PI que atendam as especificações mostradas.

Extraia os valores de forma aproximada do LR dado.

Considerações gerais sobre o projeto do controlador PID

Considerações gerais sobre o projeto do controlador PID

Análise dos sinais para avaliar o desempenho
Saída seguindo Ref e rejeitando distúrbio
Sinal de controle

Projeto do PI e PID tomando como base métodos de sintonia ou síntese direta

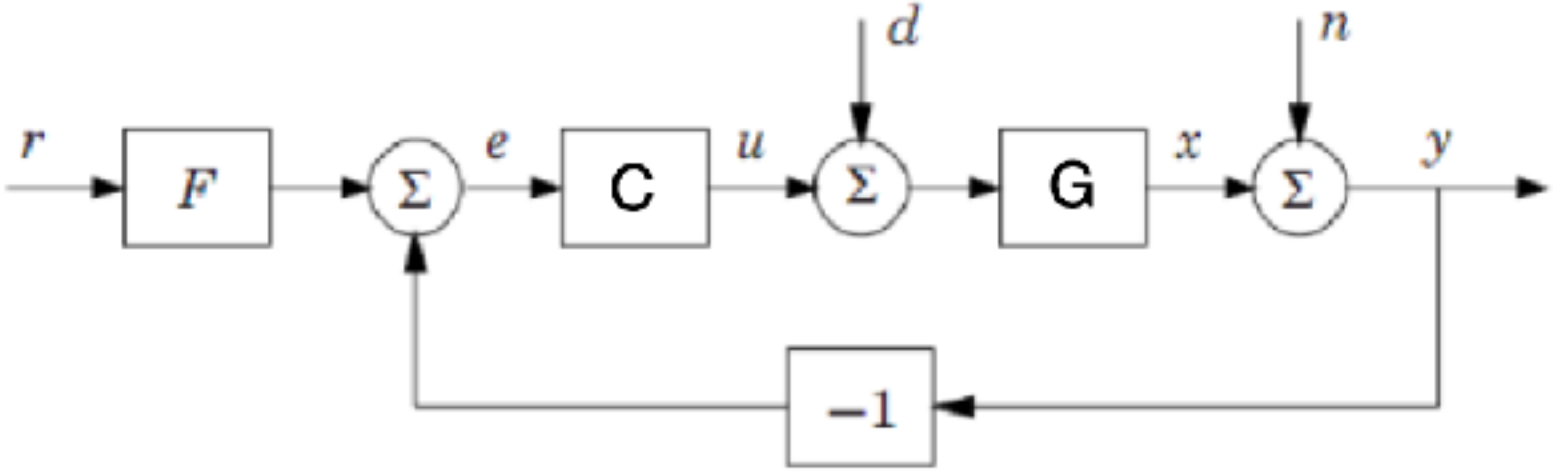
Projeto do PID a partir do PI

Métricas para avaliar o desempenho
- Referência
- Distúrbio

Modelos com atraso e ordens maiores.

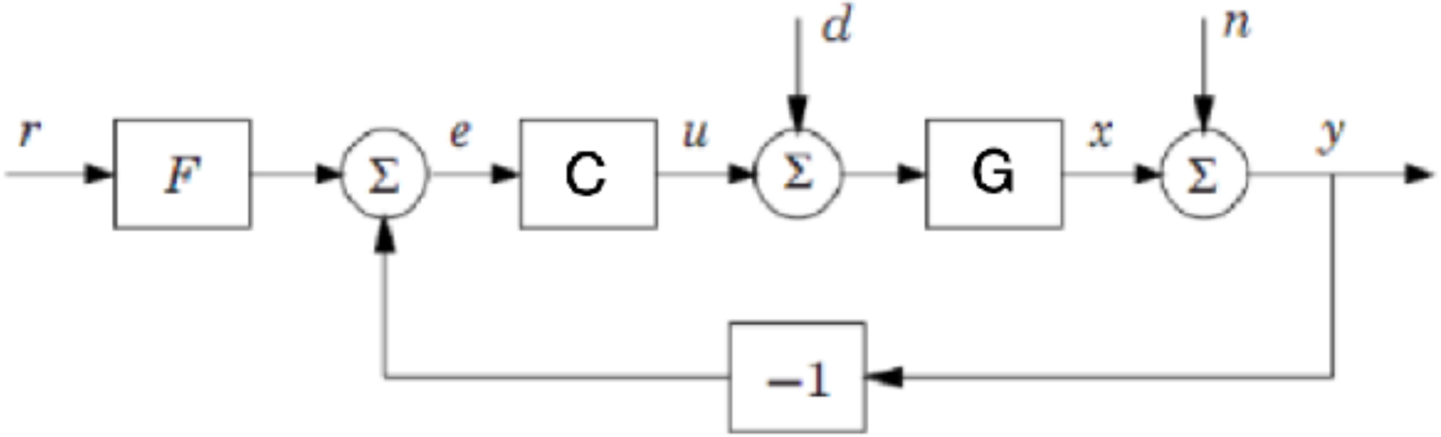
Considerações gerais sobre o projeto do controlador PID via método do lugar das raízes

Análise dos sinais para avaliar o desempenho.



A saída y e o sinal de controle u são utilizados para avaliar o projeto e ajudar nas escolhas e decisões. Para isto, aplica-se a referência em r e depois o distúrbio em d, ou n.

Considerações gerais sobre o projeto do controlador PID via método do lugar das raízes



Considerando F=1, y é obtido de $\dfrac{Y(s)}{R(s)} = \dfrac{C(s)G(s)}{1 + C(s)G(s)}$

O sinal de controle u é obtido de $\dfrac{U(s)}{R(s)} = \dfrac{C(s)}{1 + C(s)G(s)}$

Considerações gerais sobre o projeto do controlador PID via método do lugar das raízes

$$\frac{U(s)}{R(s)} = \frac{C(s)}{1 + C(s)G(s)}$$

Essa FT que fornece u pode conter mais zeros que polos, ou seja, ser não causal.
Neste caso, a simulação não pode ser feita.

Seja $C(s) = \dfrac{B(s)}{A(s)}$ e $G(s) = \dfrac{N(s)}{D(s)}$

Então, $\dfrac{C(s)}{1 + C(s)G(s)} = \dfrac{B/A}{1 + BN/AD} = \dfrac{BD}{AD + BN}$

Caso o grau do polinômio BD tenha grau maior que o grau de AD, teremos um sistema não causal.

Considerações gerais sobre o projeto do controlador PID via método do lugar das raízes

$$\frac{C(s)}{1 + C(s)G(s)} = \frac{B/A}{1 + BN/AD} = \frac{BD}{AD + BN}$$

Seja, por exemplo, $C(s) = K_p \dfrac{s + \frac{K_i}{K_p}}{s}$.

Ao multiplicar BD, a ordem de D é aumentada em 1 devido ao zero em B.

Ao multiplicar AD, a ordem de D é aumentada em 1 devido ao polo em A.

Logo, BD e AD terão a mesma ordem e o sistema será causal.

Assume-se aqui que o numerador de G(s) dado por N(s) tem ordem menor que D(s).
Logo, a ordem de BN é menor que a ordem de AD.

Considerações gerais sobre o projeto do controlador PID via método do lugar das raízes

Seja agora um controlador PID, $C(s) = \dfrac{\beta(s+z_1)(s+z_2)}{s} = \dfrac{K_d s^2 + K_p s + K_i}{s}$.

$$\frac{C(s)}{1+C(s)G(s)} = \frac{B/A}{1+BN/AD} = \frac{BD}{AD+BN}$$

Ao multiplicar BD, a ordem de D é aumentada em 2 devido aos dois zeros em B.

Ao multiplicar AD, a ordem de D é aumentada em 1 devido ao polo em A.

Logo, BD terá ordem maior que AD e o sistema será não causal.

Como simular neste caso?

Considerações gerais sobre o projeto do controlador PID via método do lugar das raízes

A solução é usar um filtro na parte derivativa,

$$C(s) = K_p + \frac{K_i}{s} + K_d \frac{s}{T_f s + 1}$$

O parâmetro T_f atua como um filtro da parte derivativa, e deve ser escolhido de forma a não interferir no desempenho do controlador. O polo do filtro estará em $s = -1/T_f$. Logo, valores pequenos de T_f introduzem polos distantes da origem, afetando pouco o comportamento de C(s).

Com essa estratégia a função de transferência do controlador C(s) tem agora 2 zeros e 2 polos e a FT $\frac{U(s)}{R(s)}$ será causal, permitindo a simulação.

Considerações gerais sobre o projeto do controlador PID via método do lugar das raízes

Exemplo: $G(s) = \dfrac{4}{s^2 + 6s + 5}$

C=pidtuning(g,2)

$$C(s) = 0.75 + \frac{0.625}{s} + 0.125s$$

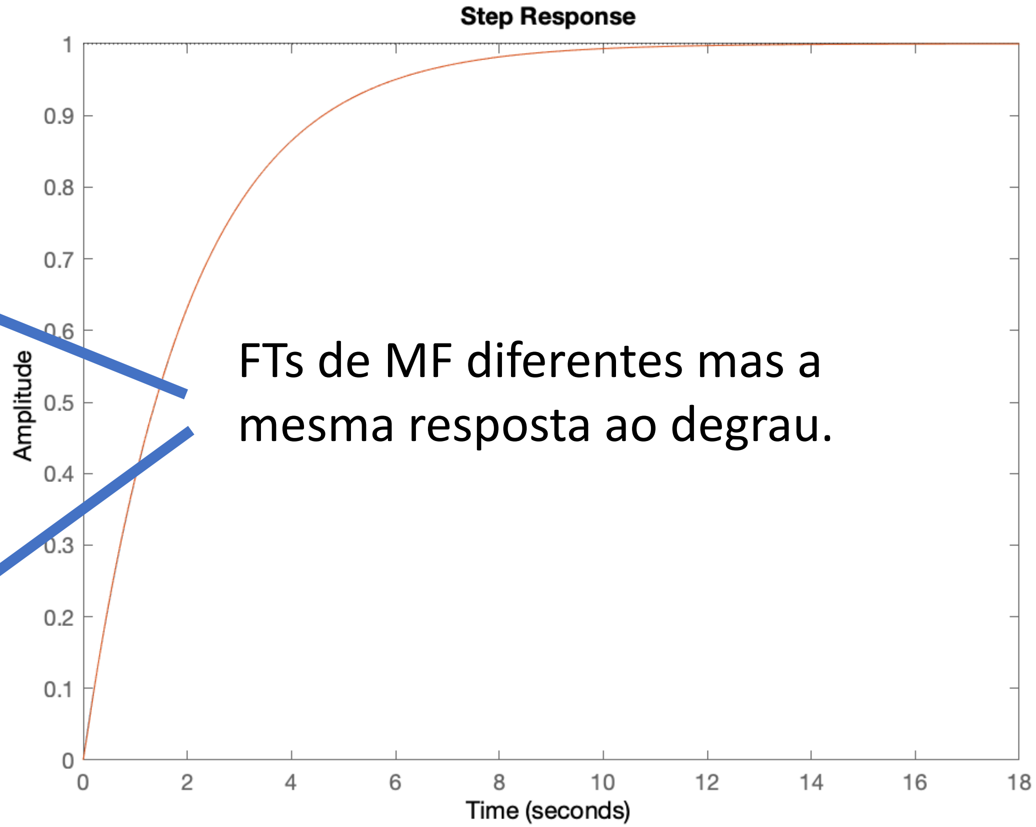
Usando $T_f = 0.01$

C.Tf=0.01 (comando no Matlab usando a variável C obtida via comando pidtuning ou comando pid.

$$C_1(s) = 0.75 + \frac{0.625}{s} + 0.125\frac{s}{0.01s + 1}$$

Considerações gerais sobre o projeto do controlador PID via método do lugar das raízes

```
>> m=feedback(C*g,1)

m =

      0.5 s^2 + 3 s + 2.5
  -------------------------
  s^3 + 6.5 s^2 + 8 s + 2.5

Continuous-time transfer function.

>> m1=feedback(C1*g,1)

m1 =

              53 s^2 + 302.5 s + 250
  -----------------------------------
  s^4 + 106 s^3 + 658 s^2 + 802.5 s +

Continuous-time transfer function.

>> step(m,m1);
```



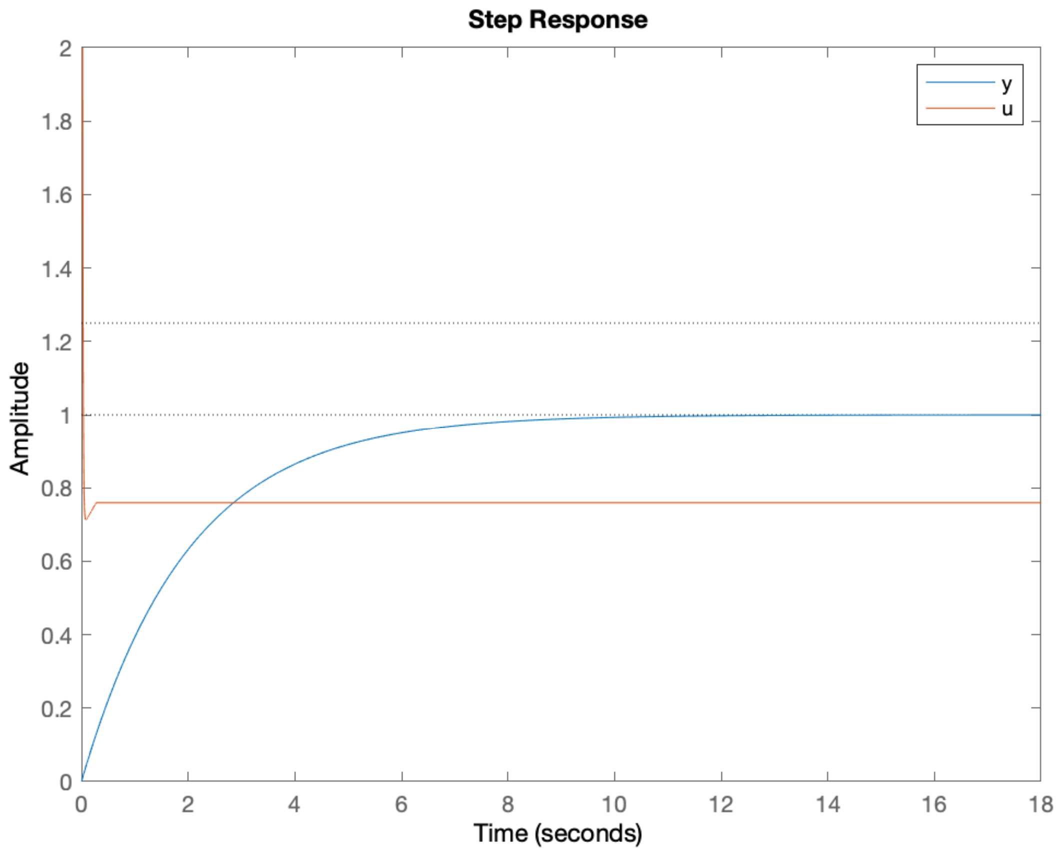
FTs de MF diferentes mas a mesma resposta ao degrau.

Considerações gerais sobre o projeto do controlador PID via método do lugar das raízes

FTs para my=y/r e mu=u/r:

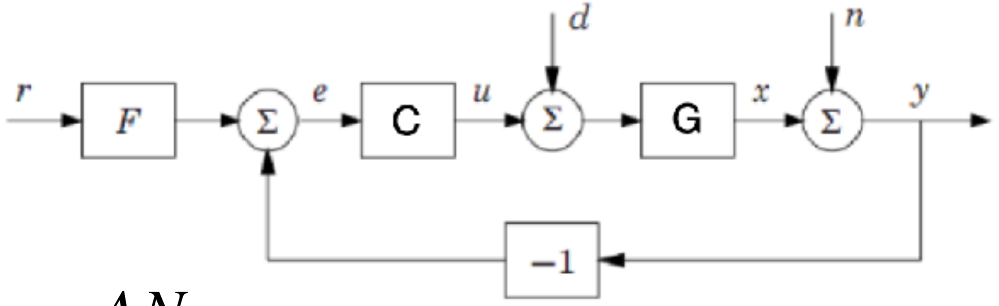
```
my=feedback(C*g,1)
mu=feedback(C1,g)
step(my,mu);
```

Agora tanto y quanto u podem ser obtidos via simulaçãoem malha fechada.



Considerações gerais sobre o projeto do controlador PID via método do lugar das raízes

Resposta ao distúrbio em U:



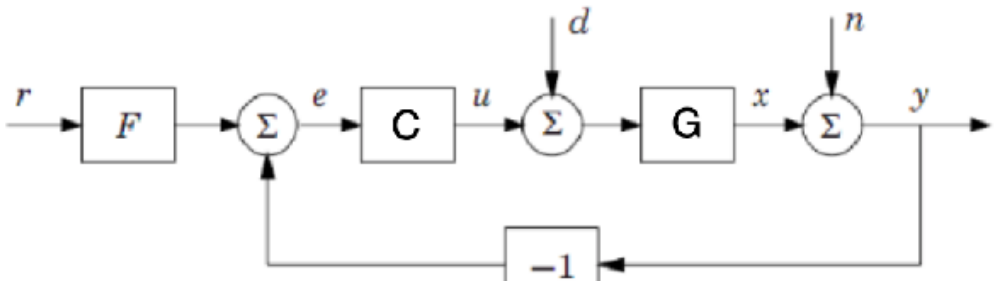
$$\frac{G(s)}{1 + C(s)G(s)} = \frac{N/D}{1 + BN/AD} = \frac{AN}{AD + BN}$$

O grau do numerador AN será sempre menor que o do denominador AD+BN, pois grau de N é menor que grau de D, e esta FT será causal.

```
md=feedback(g,C)
step(md);
```

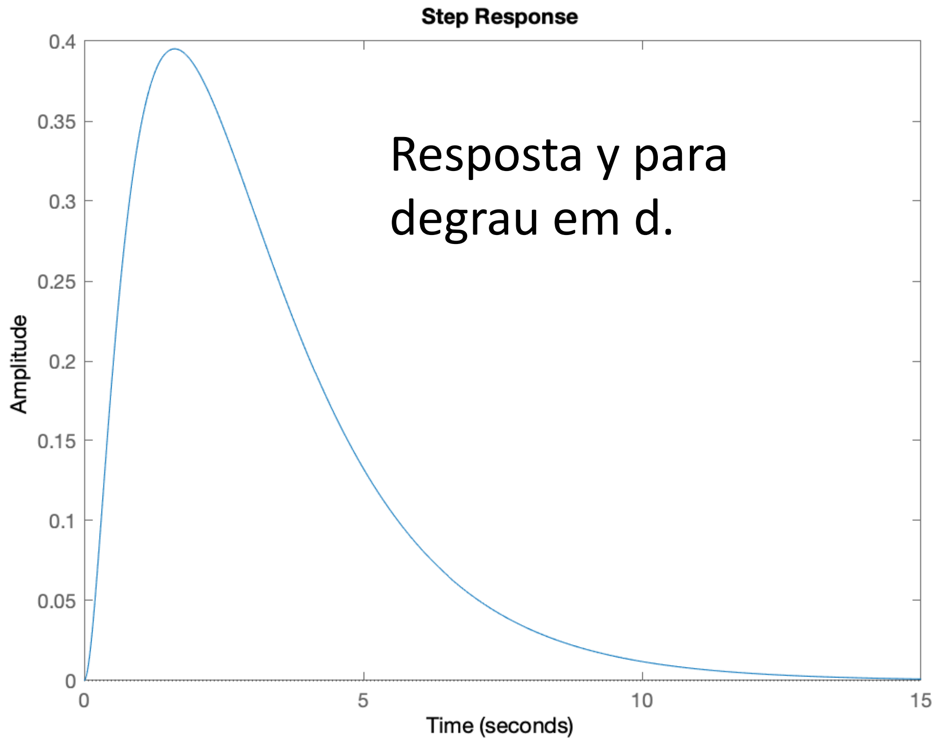
Considerações gerais sobre o projeto do controlador PID via método do lugar das raízes

Resposta ao distúrbio em U:



md=feedback(g,C)
step(md);

Não há problemas para a simulação ao distúrbio.



Resposta y para degrau em d.

Considerações gerais sobre o projeto do controlador PID via método do lugar das raízes

O método do lugar das raízes é a forma mais geral para o projeto de controladores PID, pois permite total flexibilidade na escolha dos parâmetros do controlador.

Entretanto, essa flexibilidade pode resultar em uma maior dificuldade nas escolhas de projeto.

No caso do PI, onde alocar o zero? $C(s) = K_p \dfrac{s + \frac{K_i}{K_p}}{s}$

Pode ser útil observar onde o zero é alocado para os métodos de sintonia.

Considerações gerais sobre o projeto do controlador PID via método do lugar das raízes

Seja $G(s) = \dfrac{K}{\tau s + 1}$

$$C(s)G(s) = K_p \frac{s + \frac{K_i}{K_p}}{s} \frac{K}{\tau s + 1}$$

Pelo método de síntese direta,

$$C(s) = K_p(1 + \frac{K_i}{s}) = K_p \frac{s + K_i}{s}, \text{ com } K_p = \frac{\tau}{K\lambda} \text{ e } K_i = \frac{1}{\tau}$$

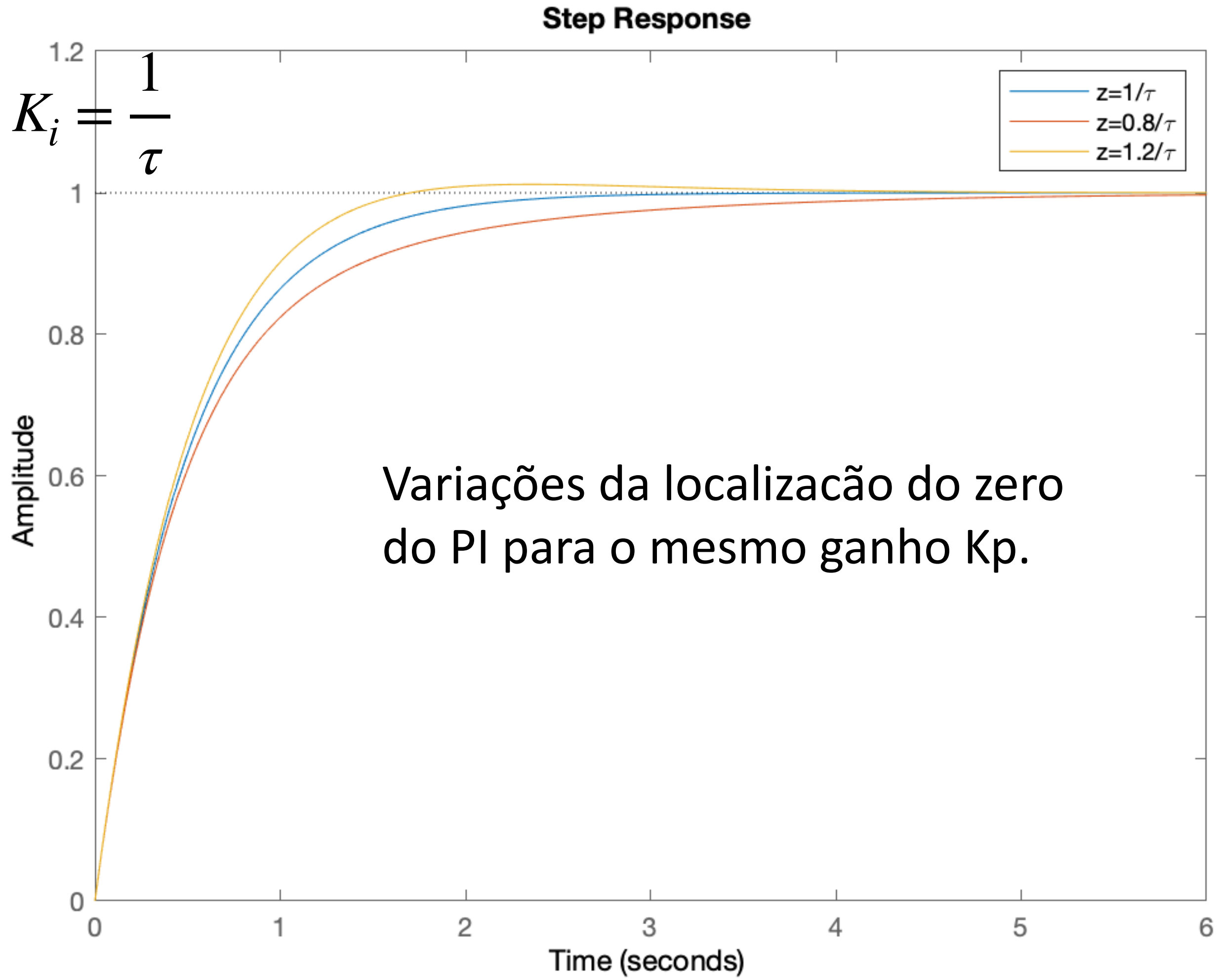
Logo, o zero é colocado em $1/\tau$.

Considerações gerais sobre o projeto do controlador PID via método do lugar das raízes

$$C(s) = K_p(1 + \frac{K_i}{s}) = K_p\frac{s + K_i}{s}, \text{ com } K_p = \frac{\tau}{K\lambda} \text{ e } K_i = \frac{1}{\tau}$$

Logo, o zero é colocado em $1/\tau$.

Portanto, essa é uma boa escolha inicial para o zero no projeto via lugar das raízes.
Variações em torno deste valor podem melhorar o desempenho.



Variações da localizacão do zero do PI para o mesmo ganho Kp.

Considerações gerais sobre o projeto do controlador PID via método do lugar das raízes

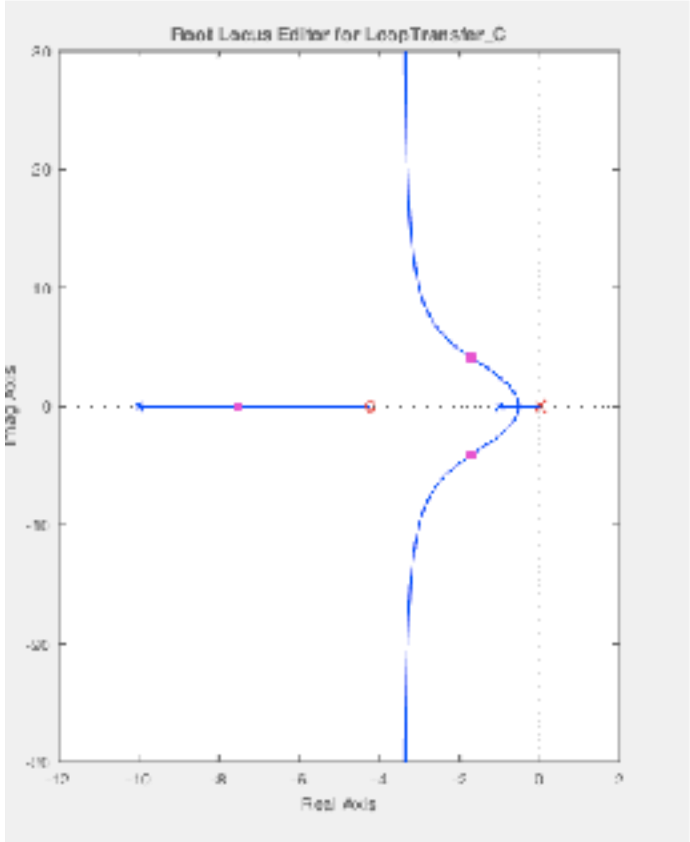
Para o projeto do PID, os métodos de síntese alocam os dois zeros do controlador PID sobre os polos da FT de MA G(s), neste caso de ordem 2 ou maior.

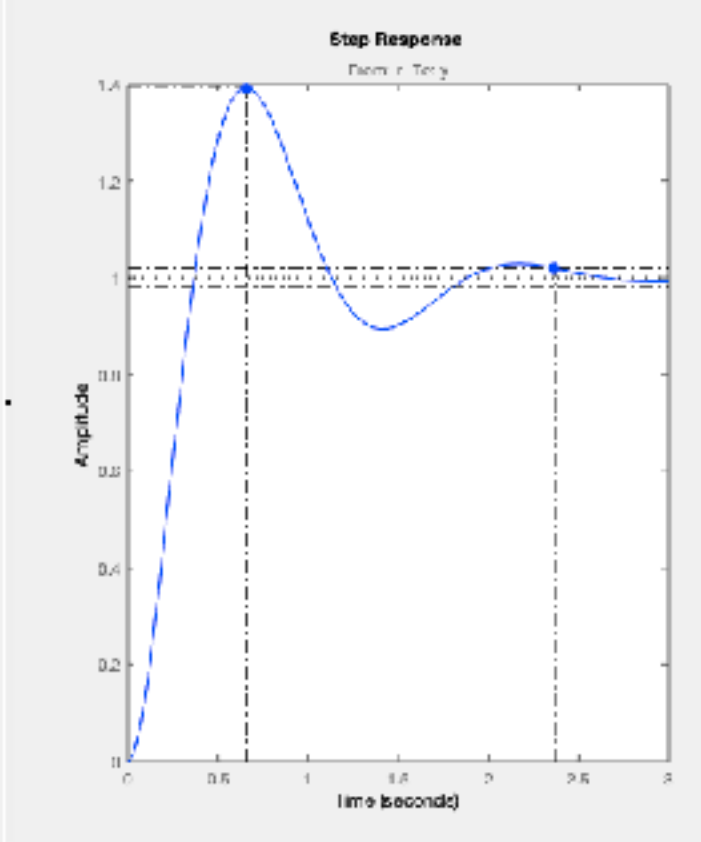
A mesma estratégia pode ser usada para as escolhas iniciais do projeto do PID.

Considerações gerais sobre o projeto do controlador PID via método do lugar das raízes

O projeto do PID a partir do projeto do PI.

Caso o projeto do controlador PI não atenda as especificações de sobreelevação, neste caso o efeito derivativo deve ser incluído, resultando em um controlador PID.





Seja $G(s) = \dfrac{5}{s^2 + 11s + 10}$

O ganho Kp foi aumentado para reduzir o tempo de estabelecimento, porém aumentando a sobreelevação.

$$C(s) = 7.5\frac{(s+4)}{s}$$

Considerações gerais sobre o projeto do controlador PID via método do lugar das raízes

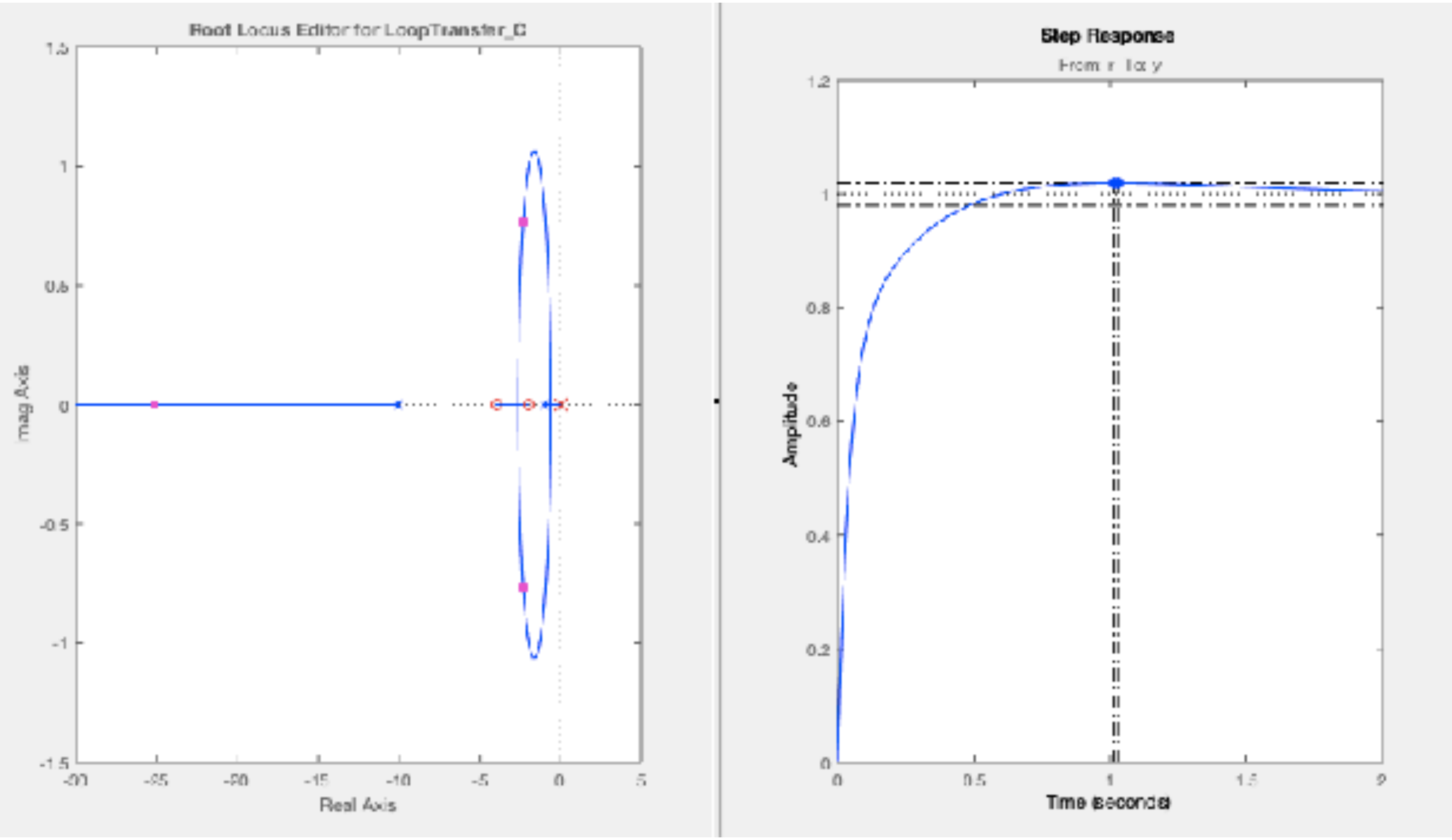
O zero do PD é adicionado no eixo real, e movimentando observando a resposta ao degrau.
A figura mostra o resultado para s=-2.

Neste caso, o ganho Kp pode ser aumentado para deixar a resposta ainda mais rápida.

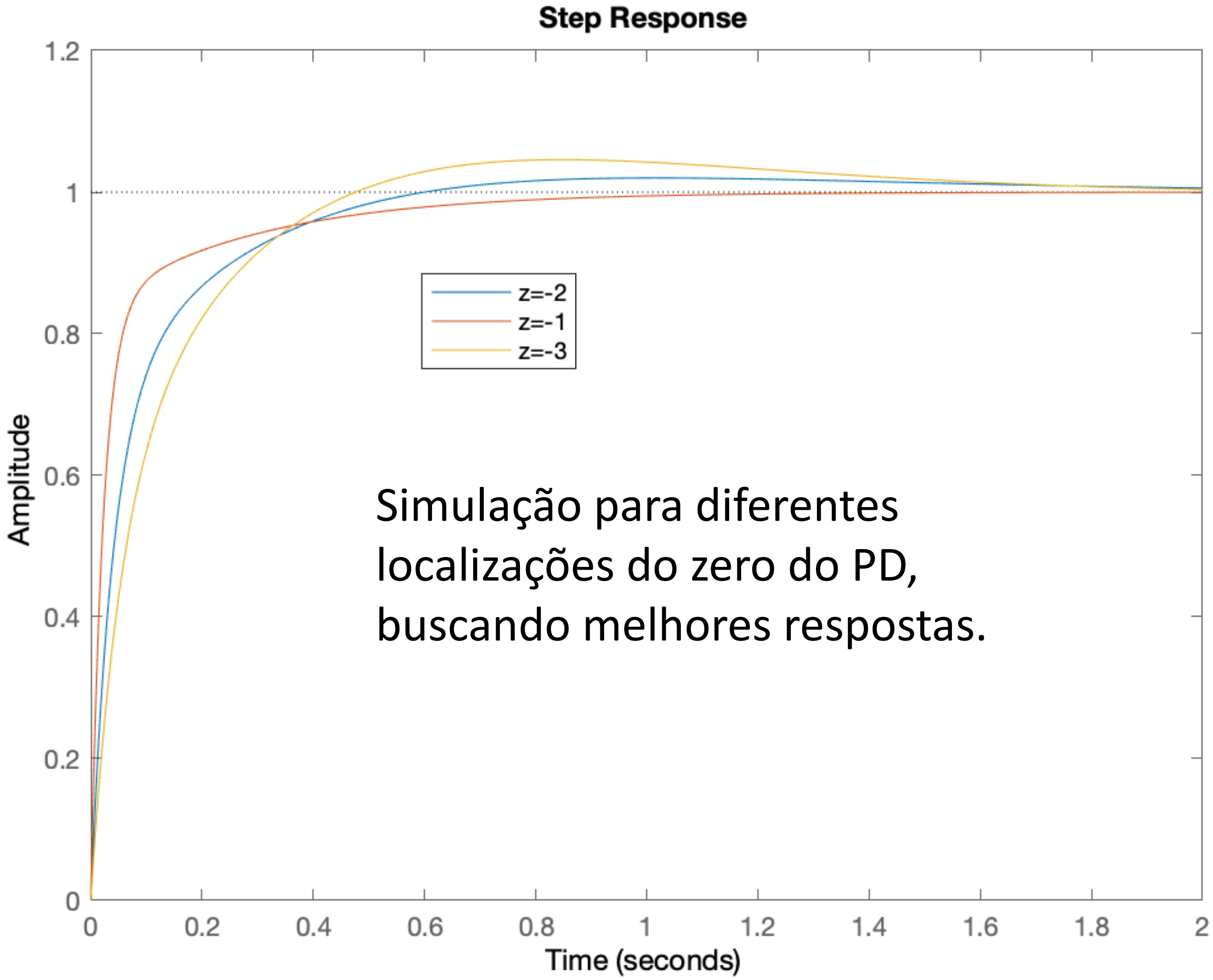
$$C(s) = 3.75\frac{(s+4)(s+2)}{s}$$

Observe que nos dois projetos o ganho foi 30.

$$C(s) = 30\frac{(0.25s+1)(0.5s+1)}{s}$$



Considerações gerais sobre o projeto do controlador PID via método do lugar das raízes



Simulação para diferentes localizações do zero do PD, buscando melhores respostas.

Considerações gerais sobre o projeto do controlador PID via método do lugar das raízes

Métricas para avaliar o desempenho

- Referência
- Distúrbio

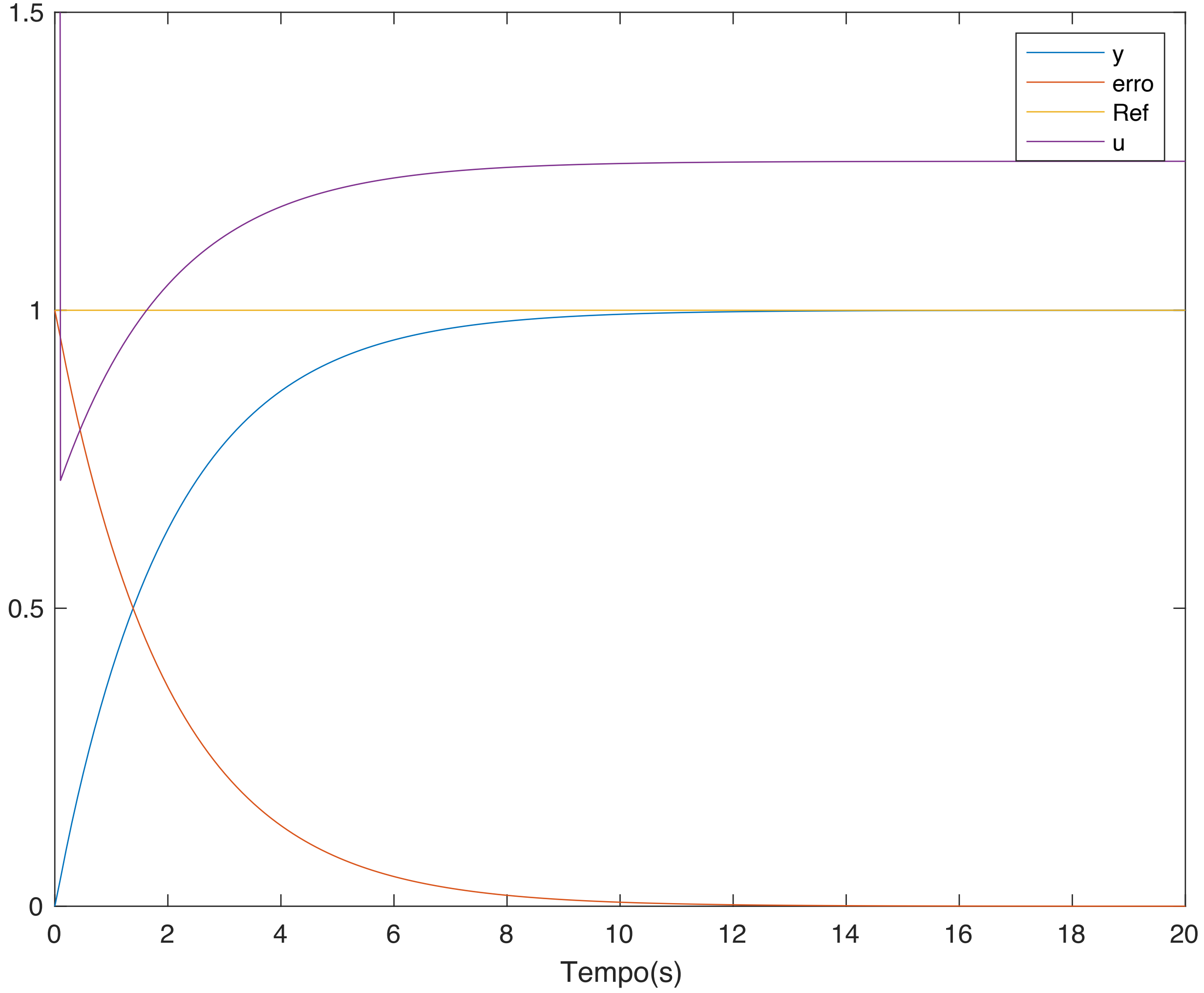
Seja novamente $G(s) = \dfrac{4}{s^2 + 6s + 5}$ e o controlador PID projetado via síntese dierta:

C=pidtuning(g,2)

$$C(s) = 0.75 + \frac{0.625}{s} + 0.125\frac{s}{0.01s + 1} \qquad T_f = 0.01$$

Considerações gerais sobre o projeto do controlador PID via método do lugar das raízes

Analisando a resposta ao degrau unitário temos:




```
>> S=stepinfo(m)

S =

  struct with fields:

        RiseTime: 4.3933
    SettlingTime: 7.8241
     SettlingMin: 0.9043
     SettlingMax: 1.0000
       Overshoot: 0
      Undershoot: 0
            Peak: 1.0000
        PeakTime: 21.0799
```

IAE=trapz(t,abs(e))
IAE=2

cef=sum(abs(diff(u)))
cef=13.07

Considerações gerais sobre o projeto do controlador PID via método do lugar das raízes

O erro em regime, a sobreelevação, o tempo de estabelecimento são normalmente utilizados para avaliar a resposta no tempo.
Entretanto, também é importante verificar se o sinal de controle atende condições reais da aplicação.

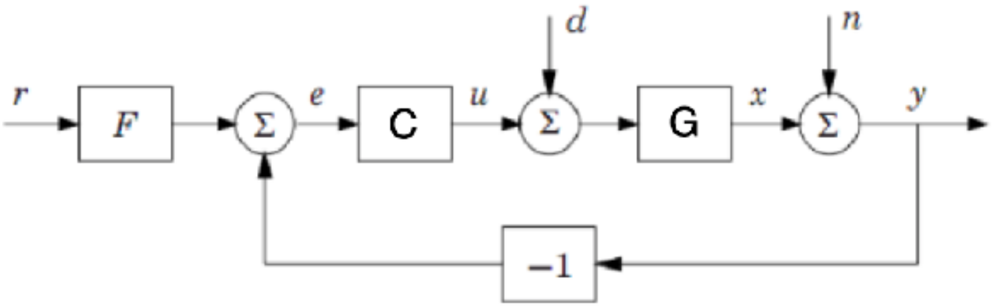
O sinal de controle sempre é limitado. Por exemplo, um motor DC de 12V não pode receber um sinal de controle maior que 12V, embora isso possa ser feito em simulação.
A limitação do sinal de controle limita o tempo de resposta.

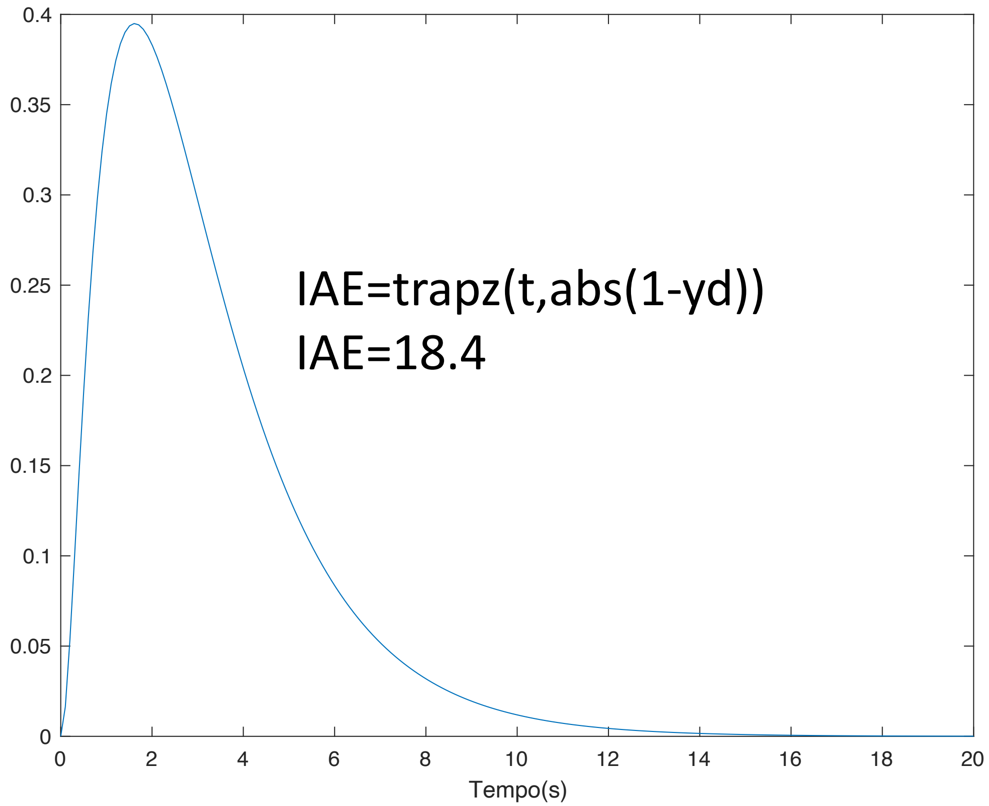
O esforço do controlador pode ser medido pela métrica cef=sum(abs(diff(u))), que soma a diferenca do sinal de controle entreo instante atual e o anterior a cada instante de amostragem.

A integral absoluta do erro (IAE) encapsula as medidas de erro em regime, sobreelevação, tempos de resposta.
Quando maior forem os tempos de resposta e a sobreelevação, maior o IAE.
O IAE é uma forma muito adequada de comparar o desempenho de diferentes controladores usando apenas uar
métrica.

Considerações gerais sobre o projeto do controlador PID via método do lugar das raízes

Analisando agora a resposta ao distúrbio temos:





Neste caso não faz sentido utilizar tempos de resposta e sobreelevação, que foram definidos para a reposta ao degrau.

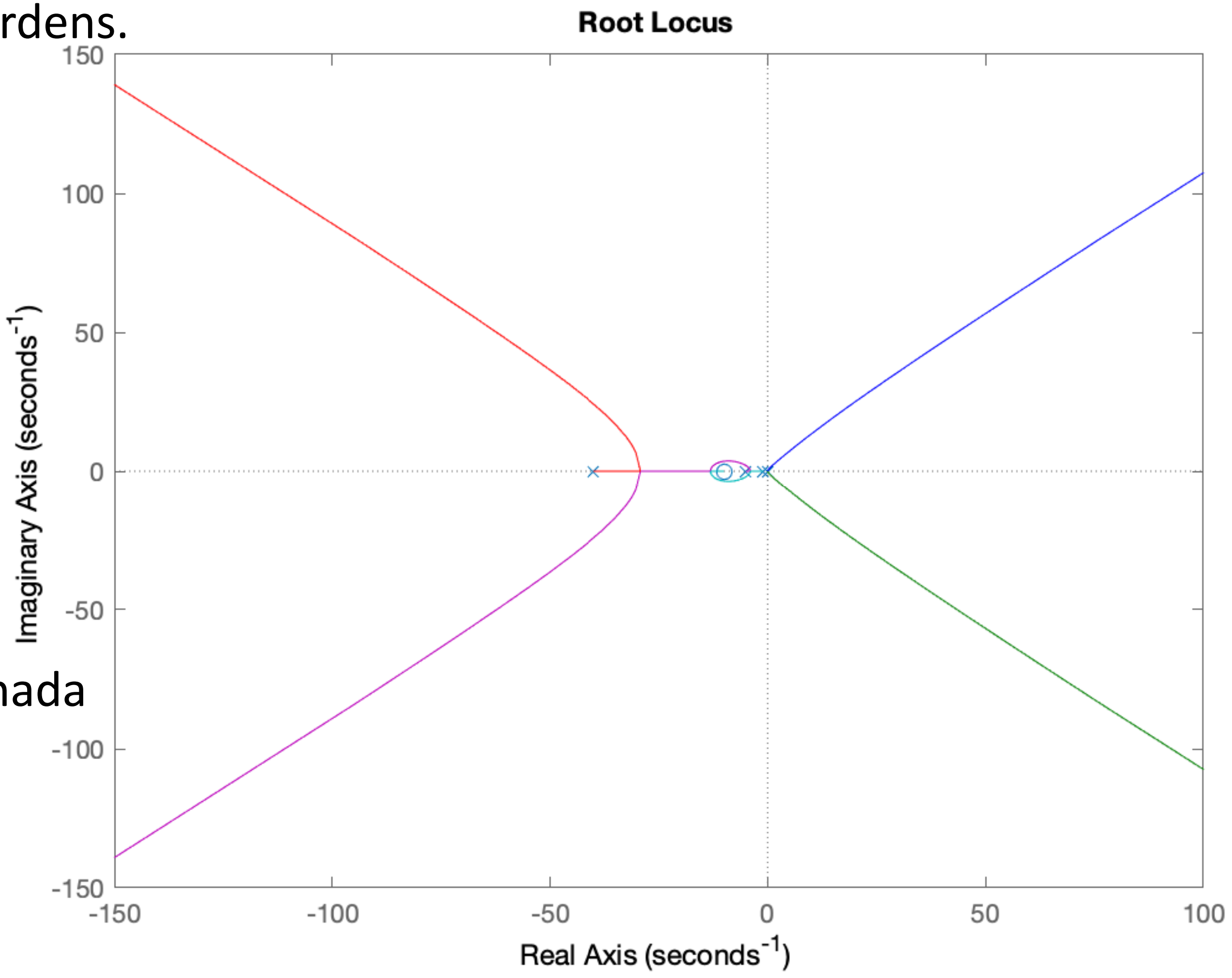
O que se espera:
Que o distúrbio desapareça em regime: pode-se medir o erro originado pelo distúrbio.
Na figura ao lado é nulo.

O IAE é uma forma adequada de medir o impacto do distúrbio no transitório, neste caso, simplemente integrando a saída, pois a referência é zero.

Considerações gerais sobre o projeto do controlador PID via método do lugar das raízes

Projeto do PID para modelos de maiores ordens.

Seja $G(s) = \dfrac{100}{s(s+1)(s+5)(s+40)}$



A análise do LR tende a ficar mais confusa.
No LR ao lado, foram adicionados um polo na origem e um zero em s=-1.

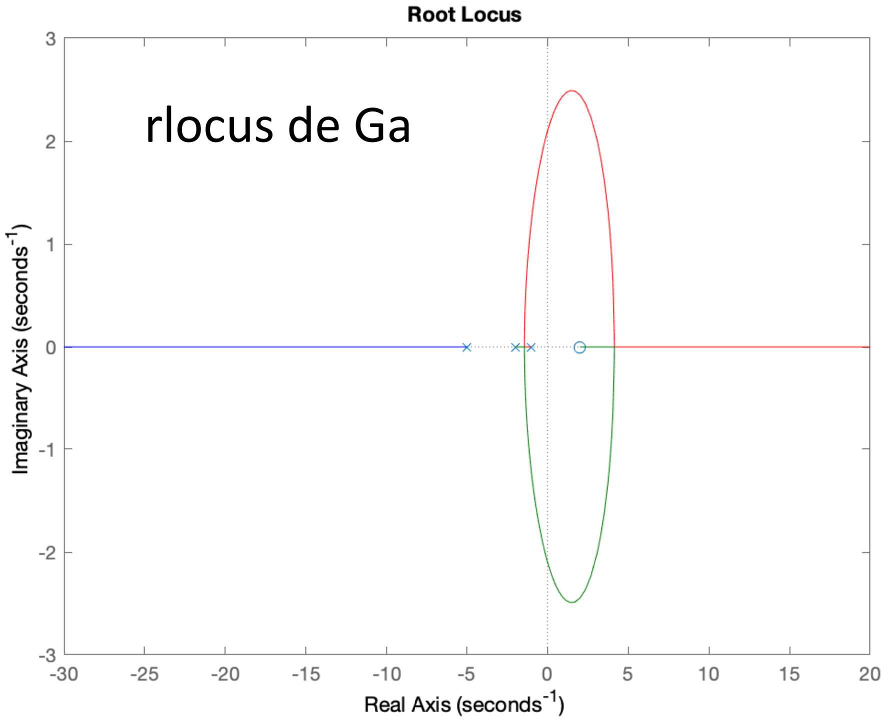
Atenção deve ser dada aos polos de malha fechada que levam à instabilidade ou a um mau desempenho.
Polos que se afastam do eixo imaginário com grande amortecimento não preocupam.

Considerações gerais sobre o projeto do controlador PID via método do lugar das raízes

Sistemas com tempo morto. Seja $G(s) = \dfrac{10e^{-s}}{(s+1)(s+5)}$

Deve-se usar a aproximação de Pade para o projeto: ga=pade(g,1)

$$G_a(s) = (-10)\frac{s-2}{(s+1)(s+5)(s+2)}$$

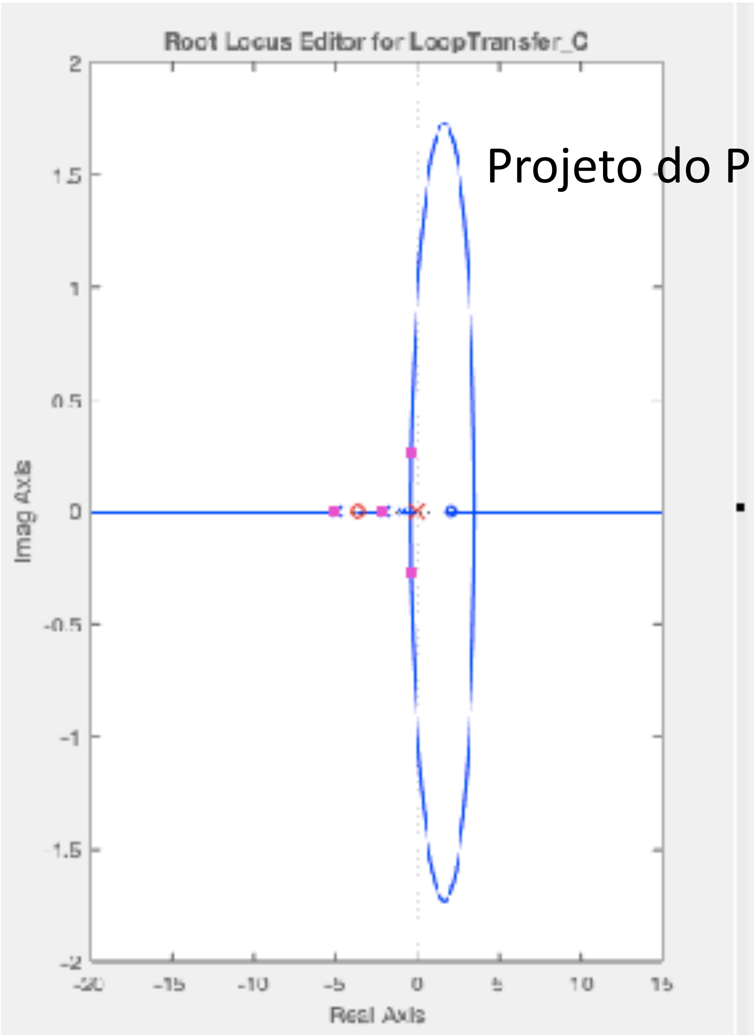


Considerações gerais sobre o projeto do controlador PID via método do lugar das raízes

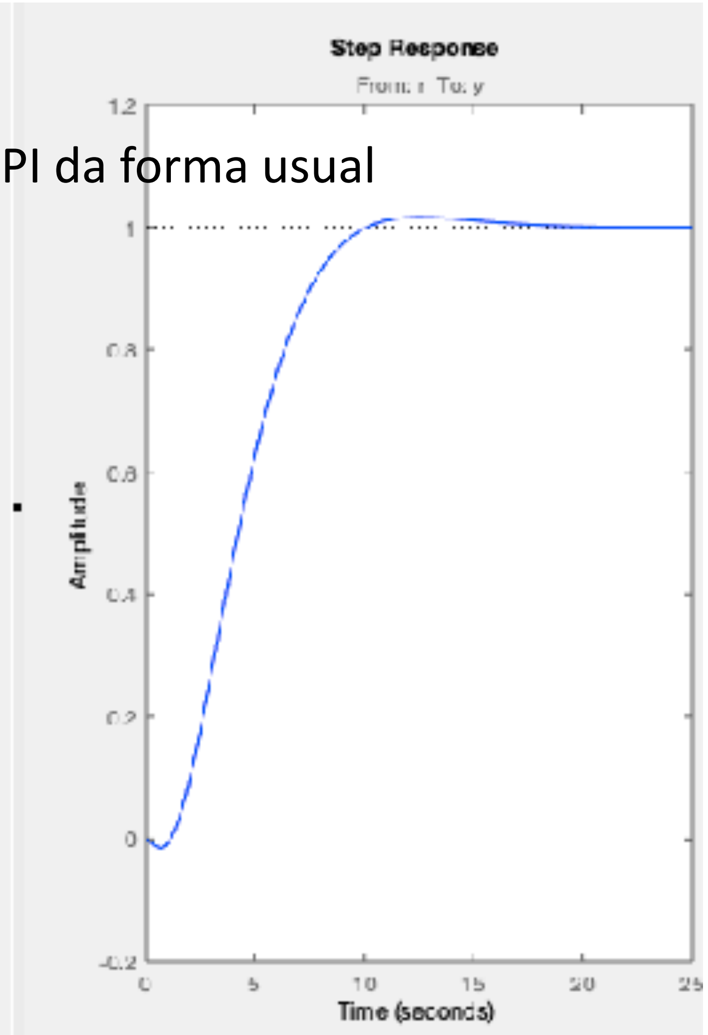
Sistemas com tempo morto. Seja $G(s) = \dfrac{10e^{-s}}{(s+1)(s+5)}$

$$G_a(s) = (-10)\frac{s-2}{(s+1)(s+5)(s+2)}$$

O projeto do PI, PD e PID segue da forma usual após a aproximação.



Projeto do PI da forma usual



Considerações gerais sobre o projeto do controlador PID via método do lugar das raízes

Análise para o caso discreto.
Quando o tempo de amostragem é pequeno comparado com a dinâmica do processo, o projeto é feito consider
o modelo contínuo e implementado em um controlador discreto.

Entretanto, quando esse não é o caso, melhor fazer análise com modelo discreto.

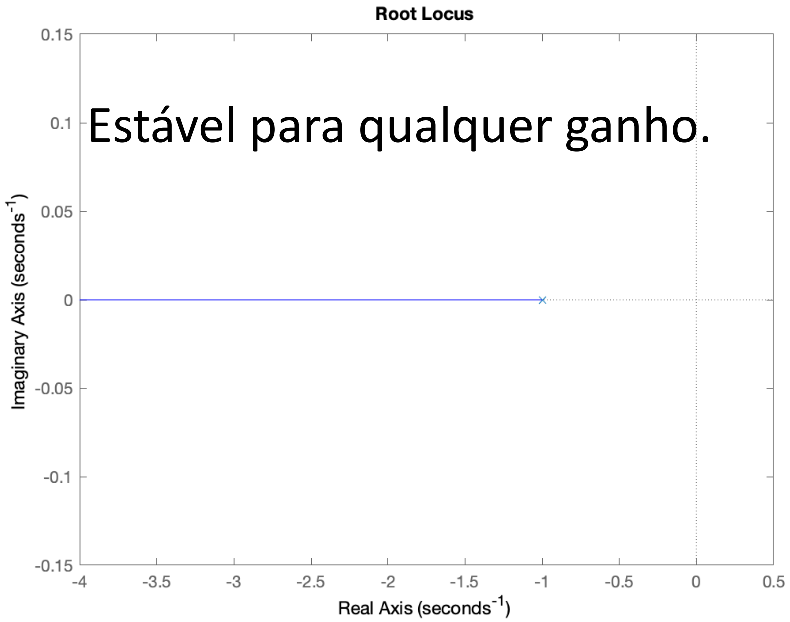
Exemplo: $G(s) = \dfrac{2}{s+1}$

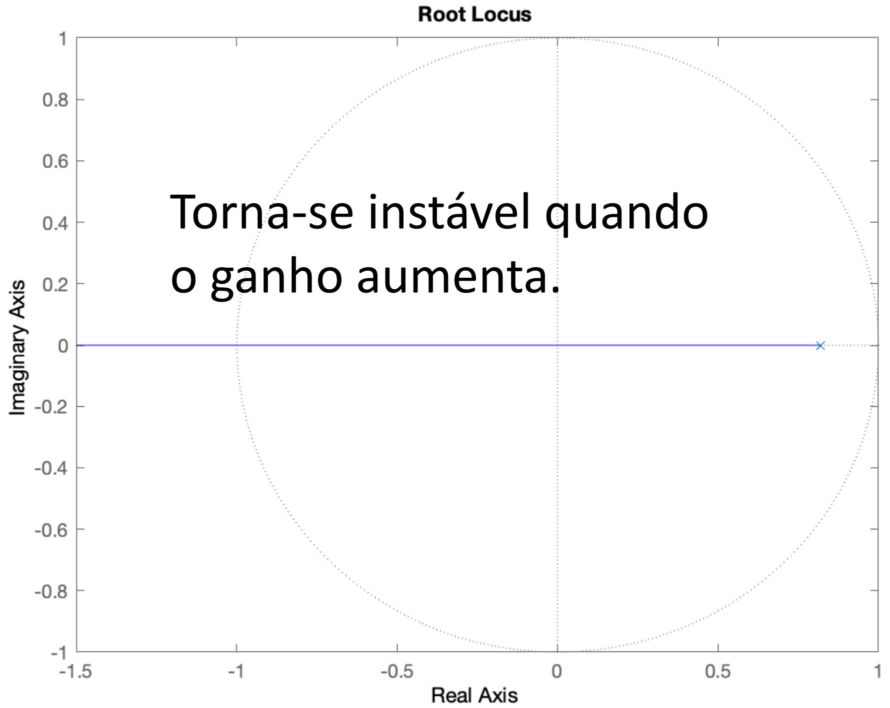
Discretizando com Ts=0.2s, $G(z) = \dfrac{0.3625}{z - 0.8187}$

Considerações gerais sobre o projeto do controlador PID via método do lugar das raízes

$$G(s) = \frac{2}{s+1}$$

$$G(z) = \frac{0.3625}{z - 0.8187}$$





Considerações gerais sobre o projeto do controlador PID via método do lugar das raízes

Seja agora o projeto de um controlador PI para G(s)

c=pidtuning(g,1) $$C(s) = 0.5 + \frac{0.5}{s}$$

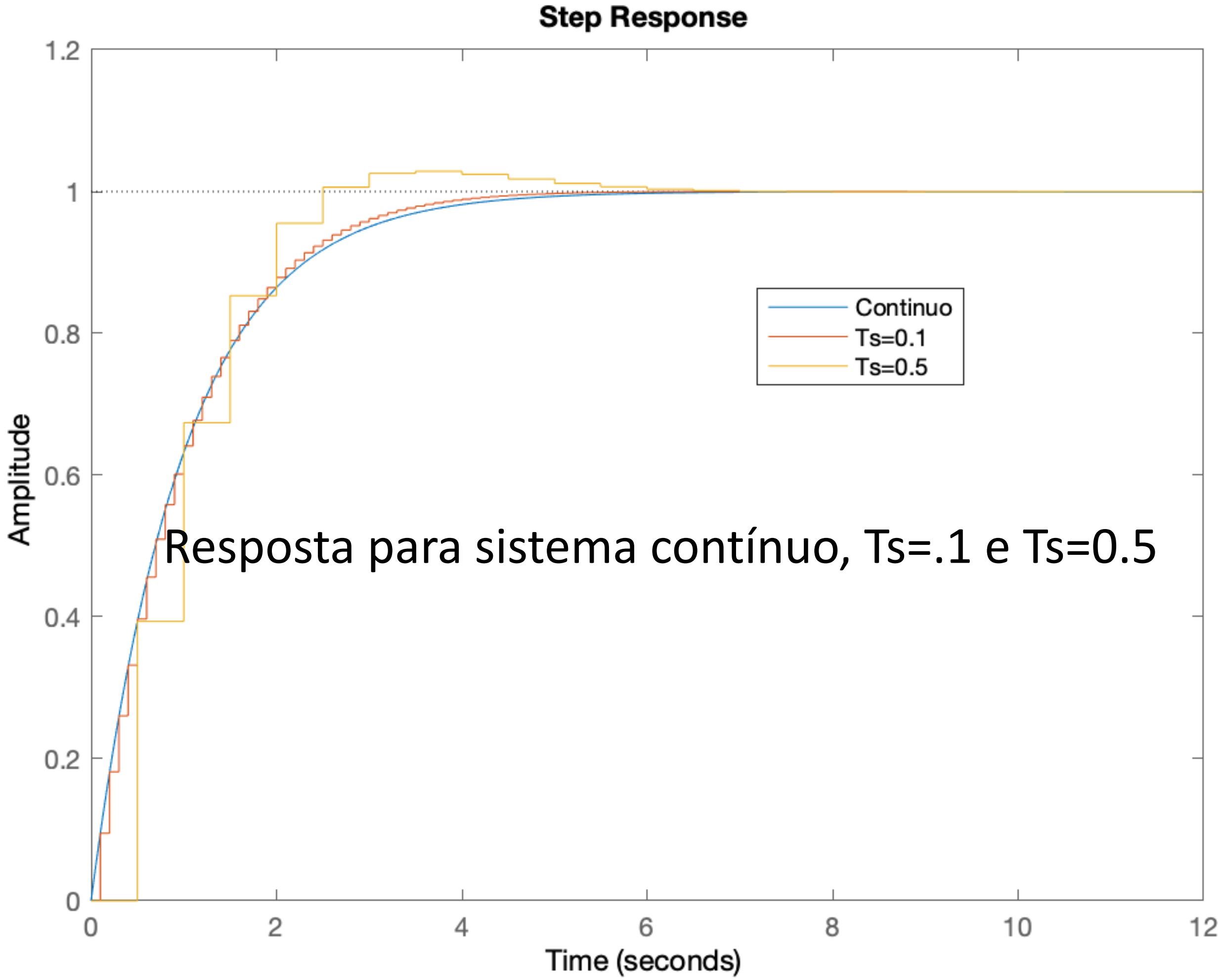
Para obter a FT de MF, $$M(z) = \frac{C(z)G(z)}{1 + C(z)G(z)}$$

com C(z) obtido de c2d(C,Ts) e G(z) obtido de c2d(G,Ts), onde Ts é o tempo de amostragem.

M(z) pode ser usada para testar o controlado contínuo C(s) implementado discretamente como C(z).

Considerações gerais sobre o projeto do controlador PID via método do lugar das raízes

Exemplo:



Resposta para sistema contínuo, Ts=.1 e Ts=0.5