

Sistemas Realimentados - 2023/2

Nome: (coloque seu nome aqui)

Data limite para entrega: 4/10, 6h da manhã.

Trabalho 3 - Projeto de controladores PID via método do lugar das raízes.

```
I=8; % Seu valor de I
[G1,G2,G3, iae_G1, iae_G3, ts_G2]=ini_t3(I);
datetime('now')
```

```
ans = datetime
      02-Oct-2023 07:54:50
```

Atividade 1: Projeto de um controlador PI para sistema de primeira ordem + tempo morto (G1(s)).

Projetar um controlador PI via método do lugar das raízes usando a FT G1. O controlador resultante C1 deve resultar em um valor de $IAE \leq iae_{G1}$ e erro nulo para entrada degrau.

Mostrar o controlador e o LR utilizado, explicar as escolhas da localização do zero do controlador para atender a especificação e a obtenção de Kp e Ki do LR.

G1

G1 =

$$\exp(-7s) * \frac{9}{28s + 1}$$

Continuous-time transfer function.
Model Properties

iae_G1

iae_G1 = 15.6732

O controlador Proporcional-Integral (PI) é um tipo de controlador que combina as ações proporcional e integral. A ação proporcional ajusta o ganho do sistema, enquanto a ação integral elimina o erro estático do sistema.

O primeiro passo para criar um controlador PI com tempo morto via Lugar das Raízes é utilizar a aproximação de Pade:

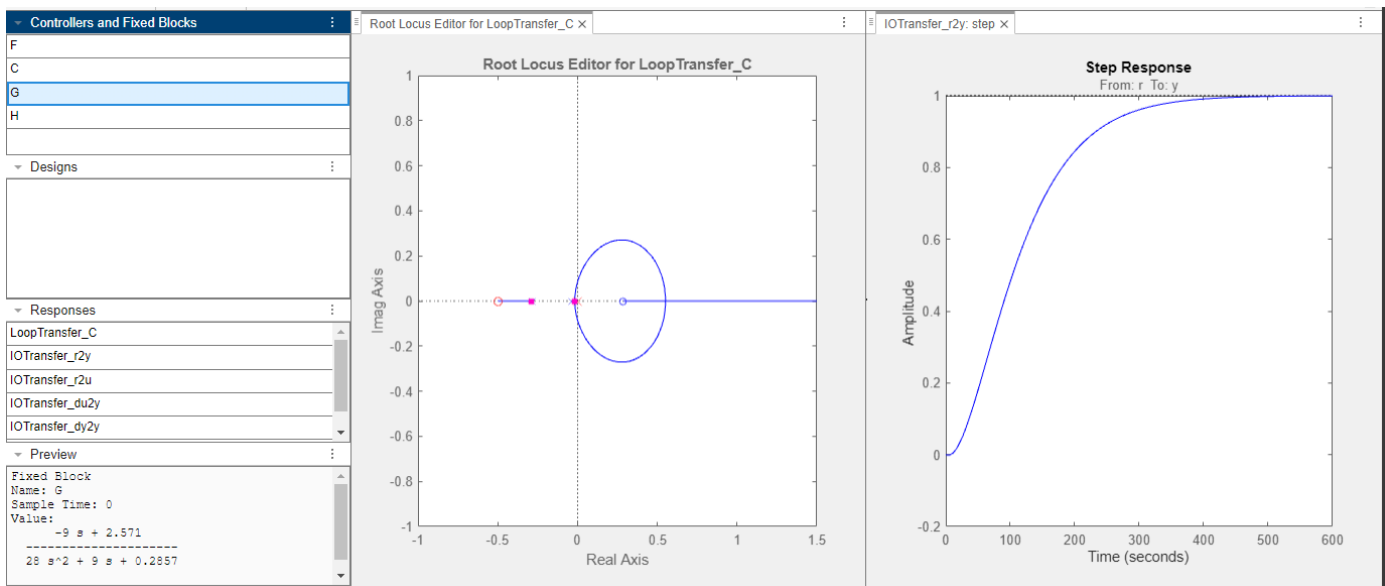
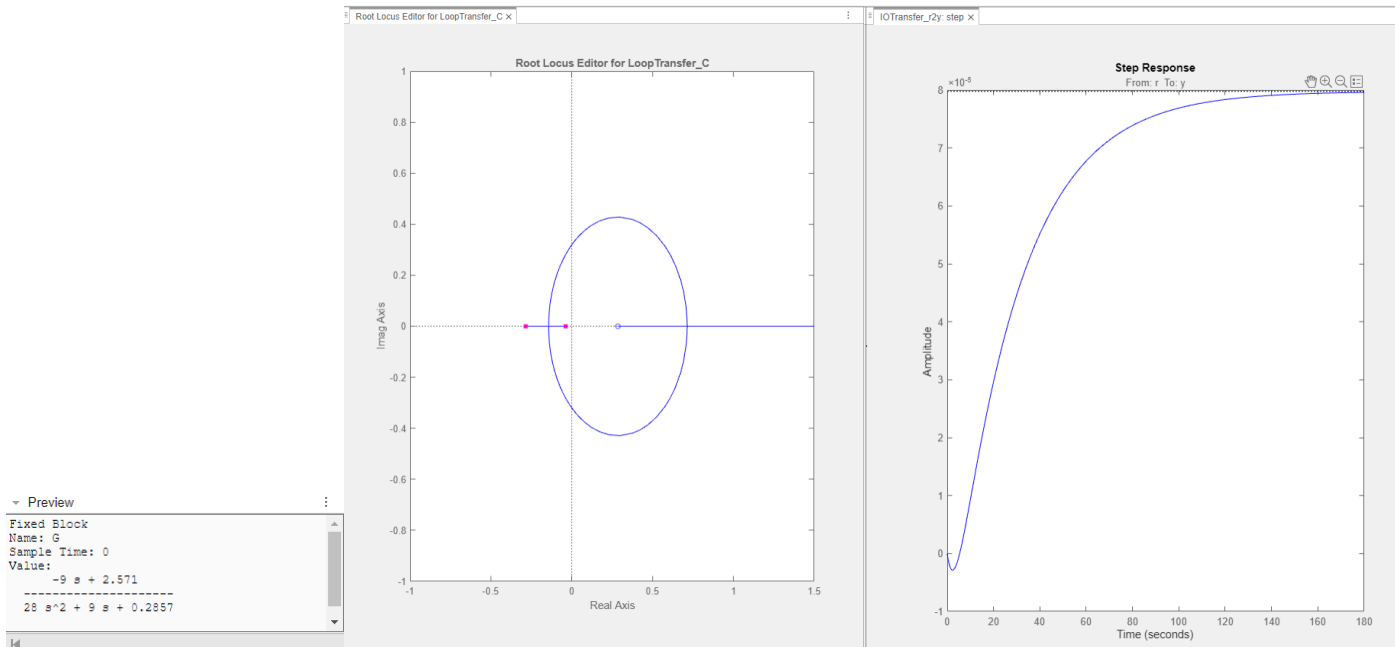
```
G_MODEL=pade(G1,1)
```

G_MODEL =

$$\frac{-9s + 2.571}{\dots}$$

$$28 s^2 + 9 s + 0.2857$$

Continuous-time transfer function.
Model Properties



O segundo passo é abrir o lugar das raízes (rltool), adicionar um polo na origem + um zero ajustável.

O polo do controlador PI está localizado em $s = 0$ (devido ao termo integral), e o zero está localizado em $-\frac{K_p}{K_i}$.

```
%Pontos retirados do rltool
s=tf('s');
```

```
control_PI= (s+0.5)/s
```

```
control_PI =
```

$$\frac{s + 0.5}{s}$$

Continuous-time transfer function.
Model Properties

```
%Controlador PI final (após adicionar um polo em 0 e um zero em -0.5  
C1= control_PI*G_MODEL
```

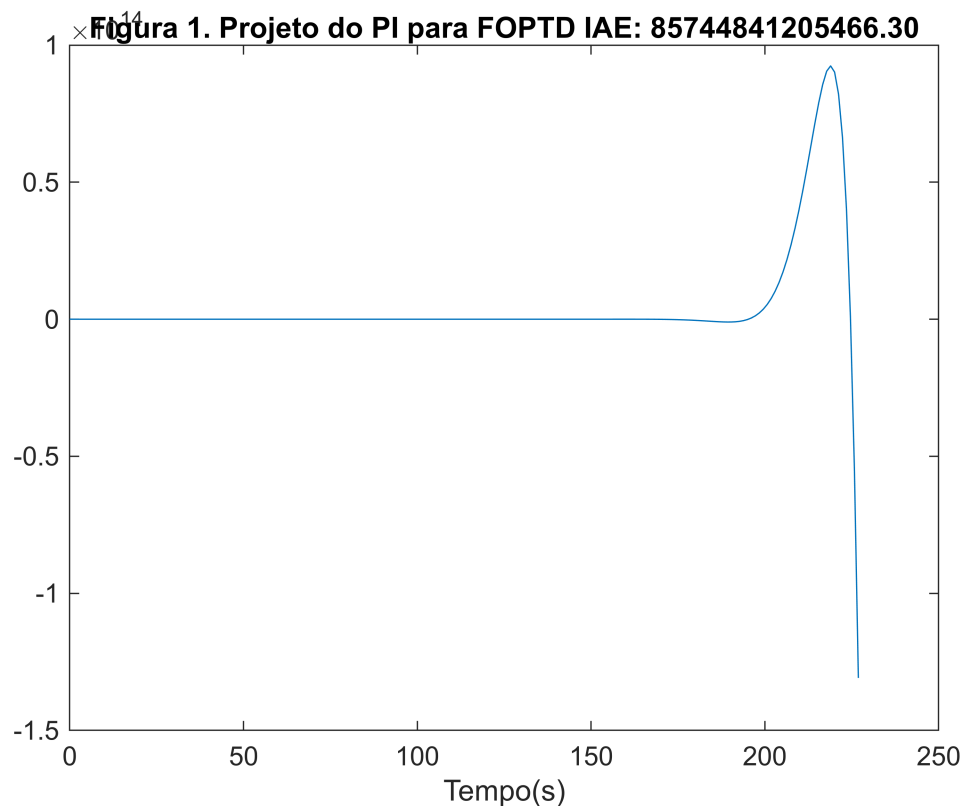
```
C1 =
```

$$\frac{-9 s^2 - 1.929 s + 1.286}{28 s^3 + 9 s^2 + 0.2857 s}$$

Continuous-time transfer function.
Model Properties

Apresentação do resultado: simulação e o cálculo do IAE usando o controlador C1 projetado.

```
M1=feedback(C1*G1,1);  
[y,t]=step(M1);  
t=linspace(0,max(t),200);  
y=step(M1,t);  
plot(t,y);xlabel('Tempo(s)');  
iae1=trapz(t,abs(1-y));  
ss=sprintf('Figura 1. Projeto do PI para FOPTD IAE: %3.2f', iae1/iae_G1);  
title(ss);
```



Atividade 2: Projeto de um controlador C2 tipo PD via método do LR para o modelo de ordem 2 $G_2(s)$ que permita obter o tempo de estabilização $t_s \leq t_{sG2}$ e com sobrelevação menor que 1%.

Mostrar o controlador e o LR utilizado, explicar as escolhas da localização do zero do controlador para atender a especificação e a obtenção de K_p e K_d do LR.

G_2
 t_{s_G2}

Abaixo a simulação com o controlador C2 projetado.

```
M2=feedback(C2*G2,1);
[y,t]=step(M2);
t=linspace(0,max(t),200);
y=step(M2,t);
plot(t,y);xline(ts_G2);
xlabel('Tempo(s)');title('Figura 2. Projeto do PD para G2')
```

Atividade 3: Projeto de um controlador C3 PI ou PID para o modelo de ordem 4 $G_3(s)$ tal que se tenha $IAE \leq iae_{G3}$.

Mostrar o(s) LR utilizado(s), explicar as escolhas para obter o controlador e atender a especificação e a obtenção dos ganhos do PID no LR.

G_3
 iae_{G3}

Abaixo a simulação e o cálculo do IAE usando o controlador C3 projetado.

```
M3=feedback(C3*G3,1);  
[y,t]=step(M3);  
t=linspace(0,max(t),200);  
y=step(M3,t);  
plot(t,y);xlabel('Tempo(s)');  
iae3=trapz(t,abs(1-y));  
ss=sprintf('Figura 3. Projeto do PID para G3 IAE: %3.2f', iae3/iae_G3);  
title(ss);
```