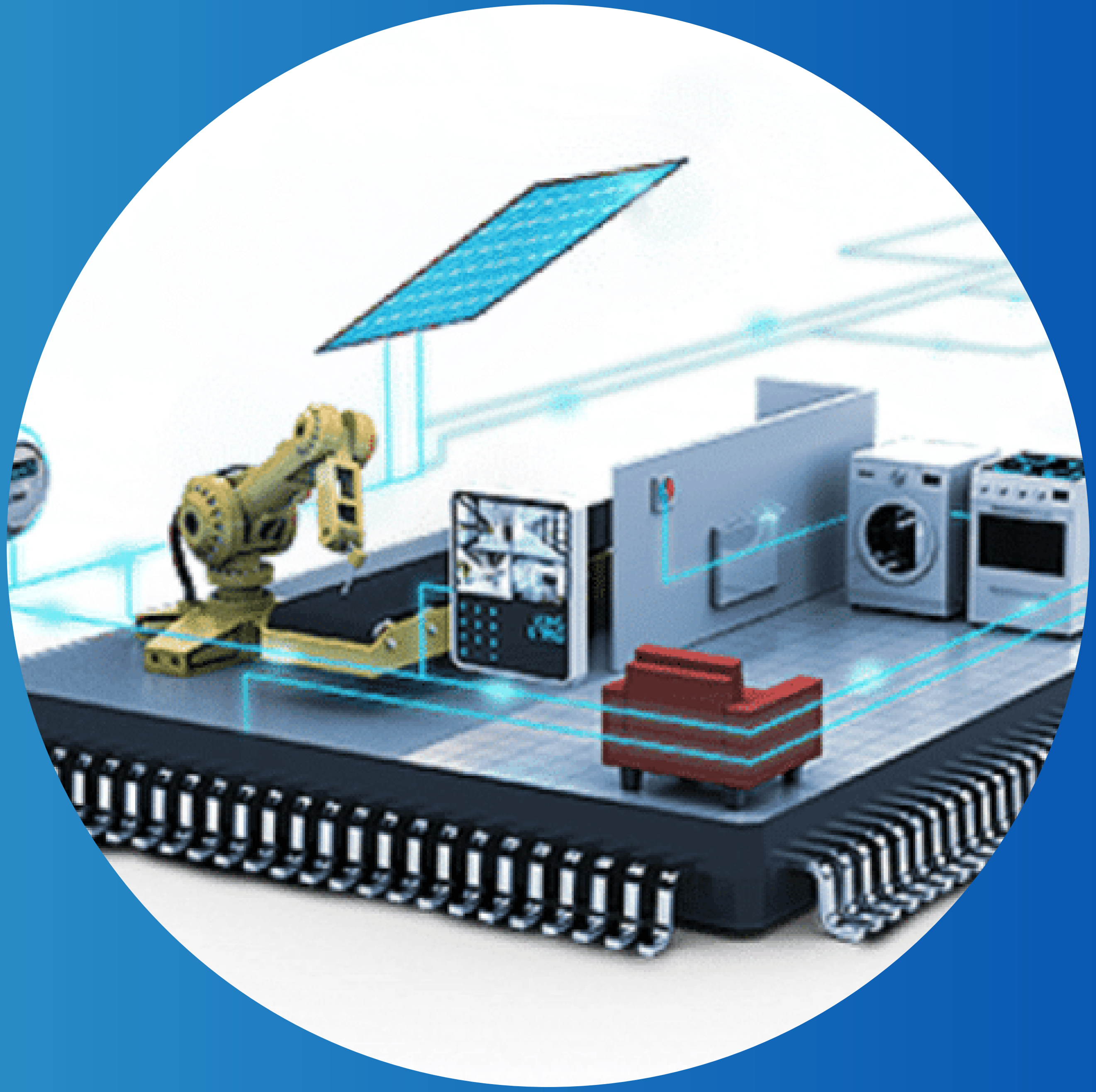
Trabalho Controle Clássico

Grupo 9

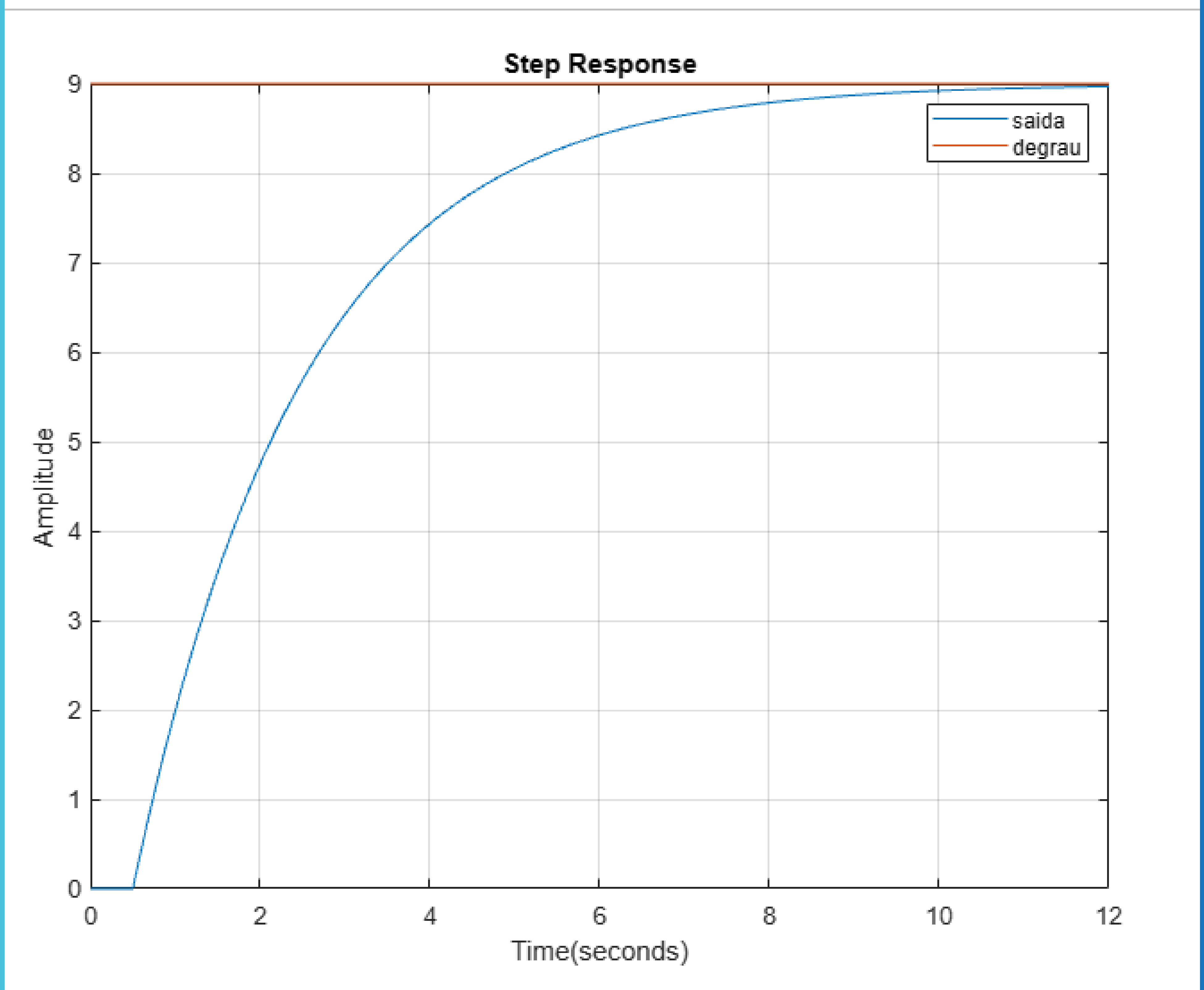
Igor Luiz Rodrigues

Rafaela Fernandes Machado

Rodrigo Paiva de Oliveira

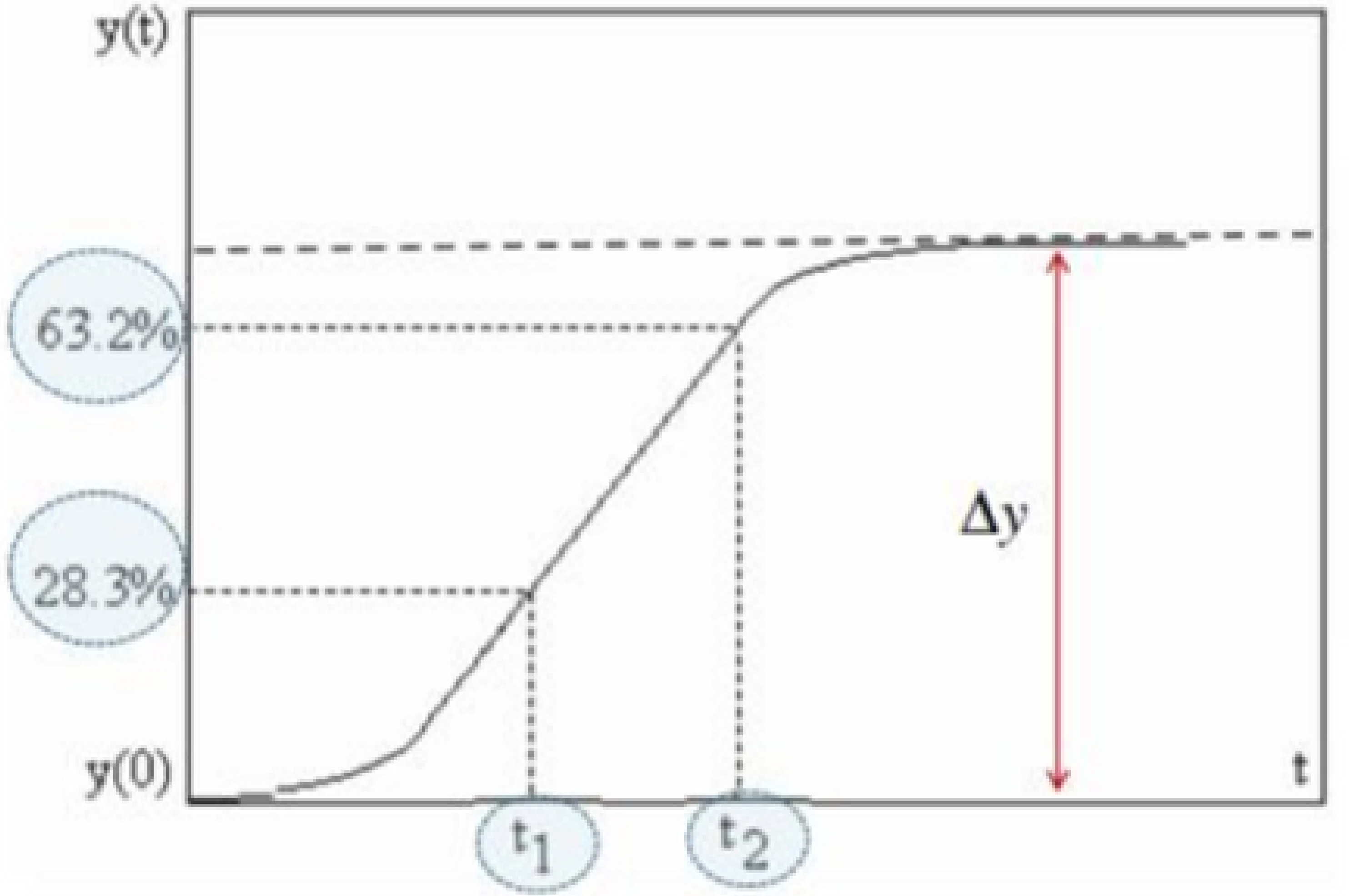


Mostrando a saida:



Smith:

Método de Smith

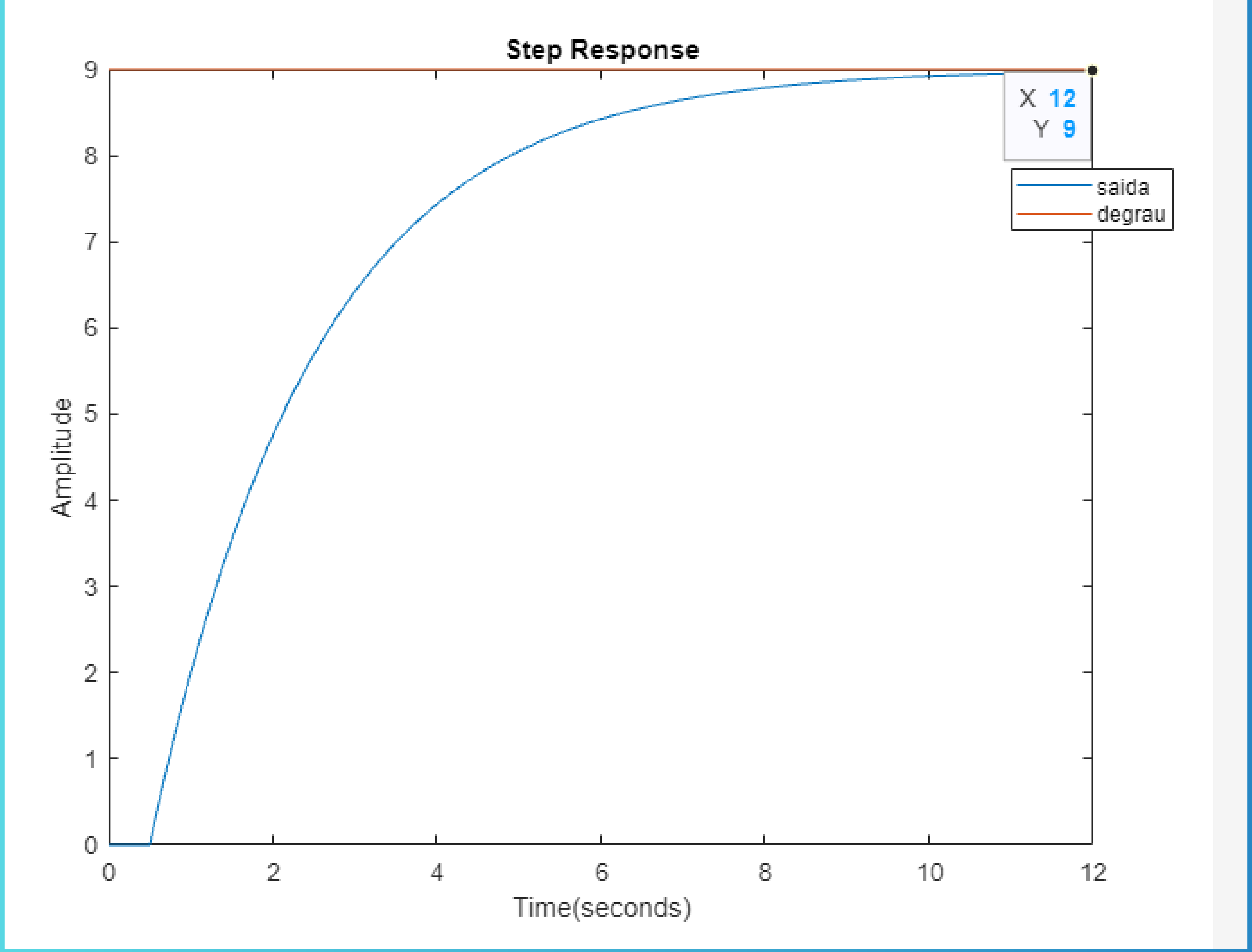


$$K = \frac{\Delta y}{\Delta u}$$

$$\tau = 1.5(t_2 - t_1)$$

$$\theta = t_2 - \tau$$

Smith:



K = Amplitude da Saída / Amplitude da Entrada

K = 9/9 = 1

τ = 1.55(t2-t1)

t1 = 28.3%*9 -> 1.2

t2 = 63.2%*9 -> 2.5

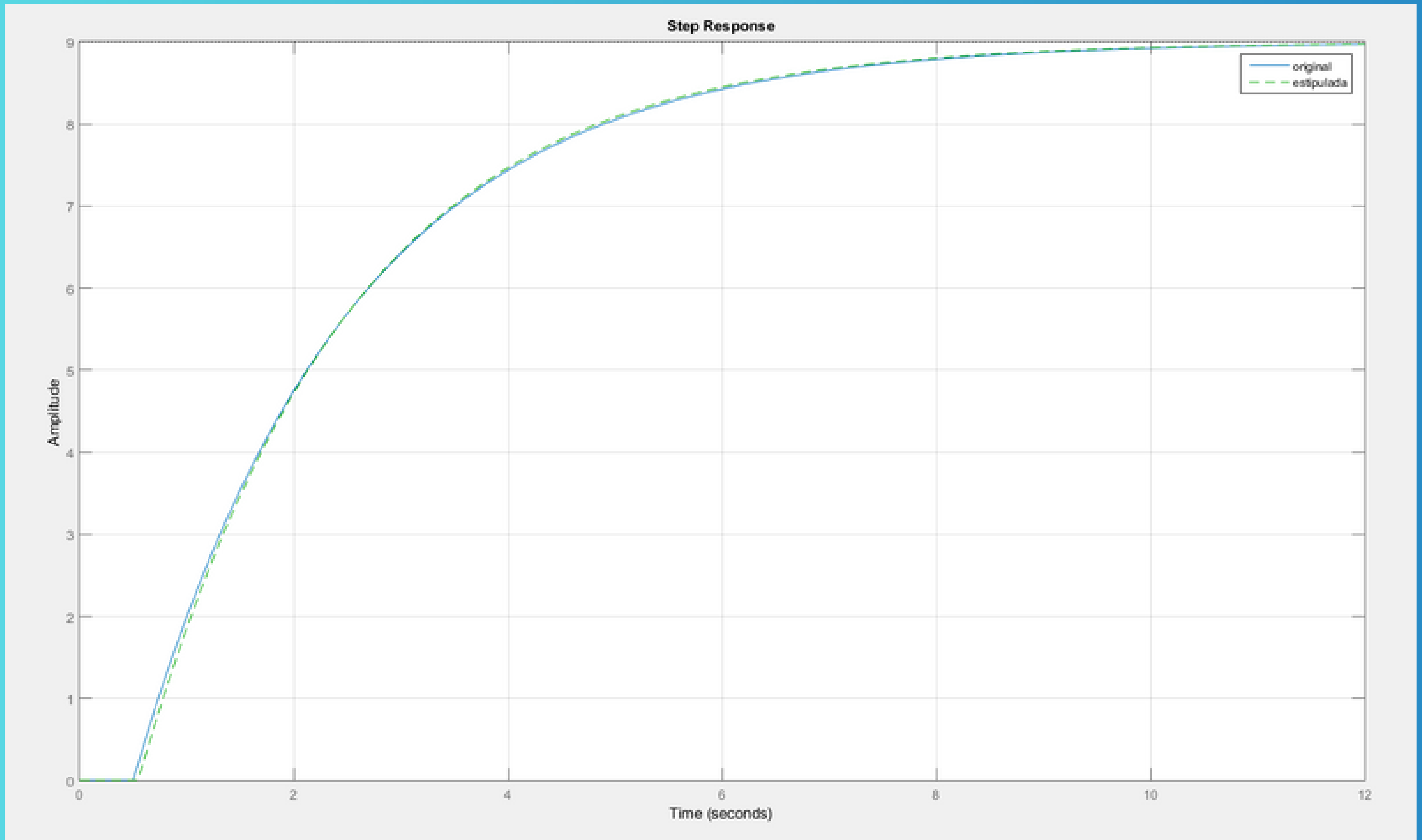
τ = 1,95 s

ϴ = t2 - τ

ϴ = 0.55 s

Original em relação a estimada:

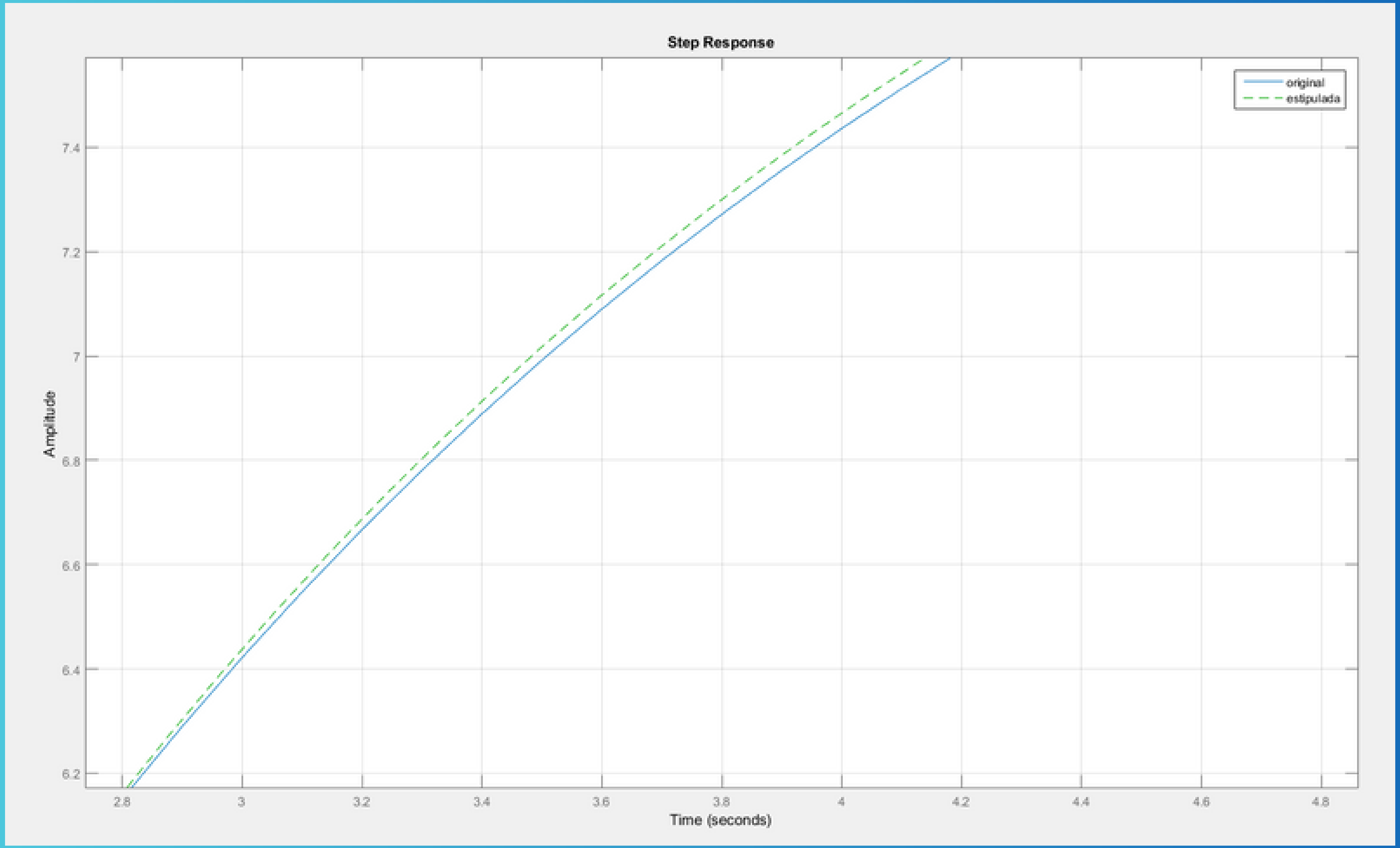
Plotando os gráficos:



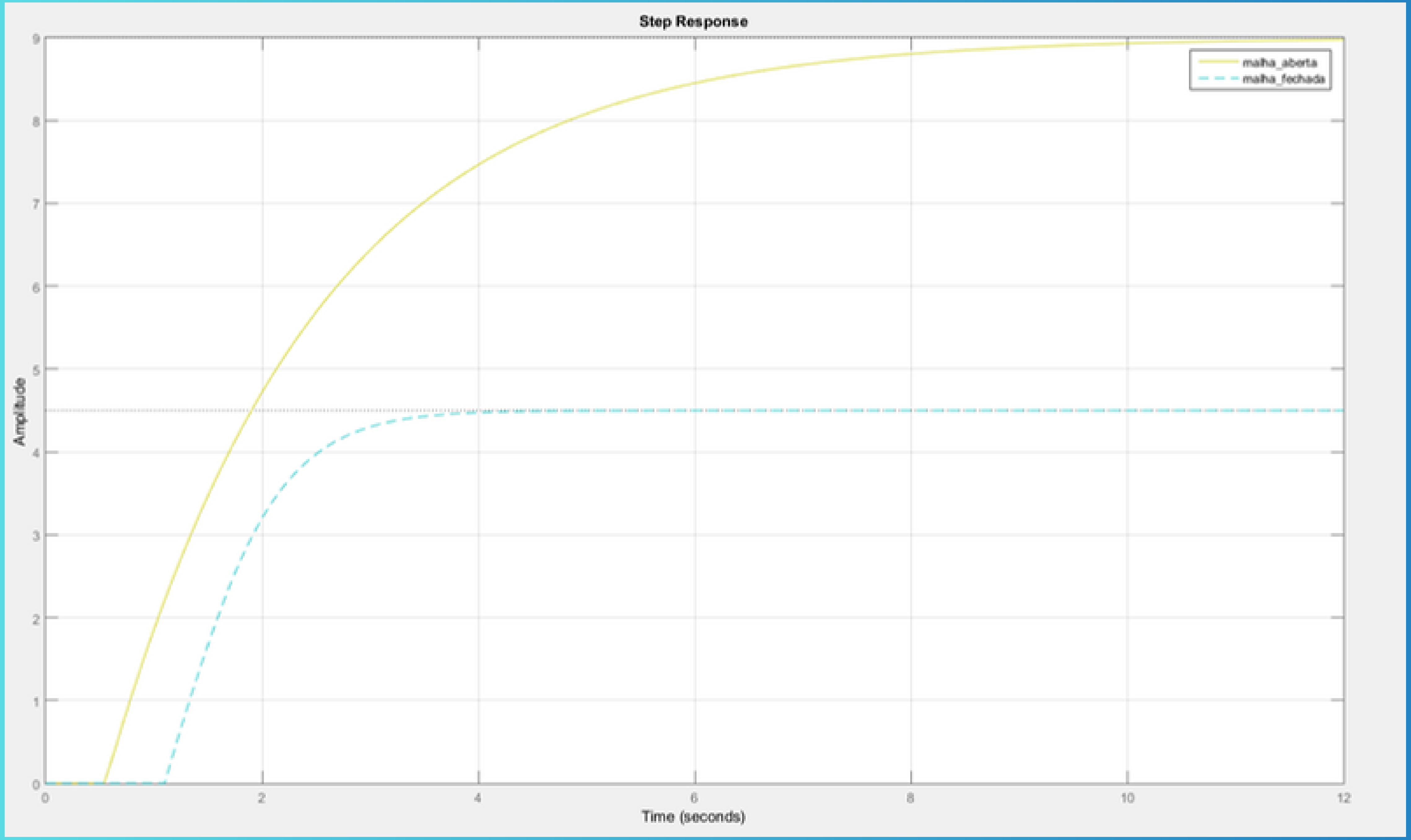

```
>> plot(t,saida)
title('Step Response')
xlabel('Time(seconds)')
ylabel('Amplitude')
hold on
sys = tf([1], [1.95 1])
sys_aberta = sys
set(sys_aberta, 'InputDelay',0.55)
step(sys_aberta*9,'g--')
grid on
legend('original','estipulada')
```

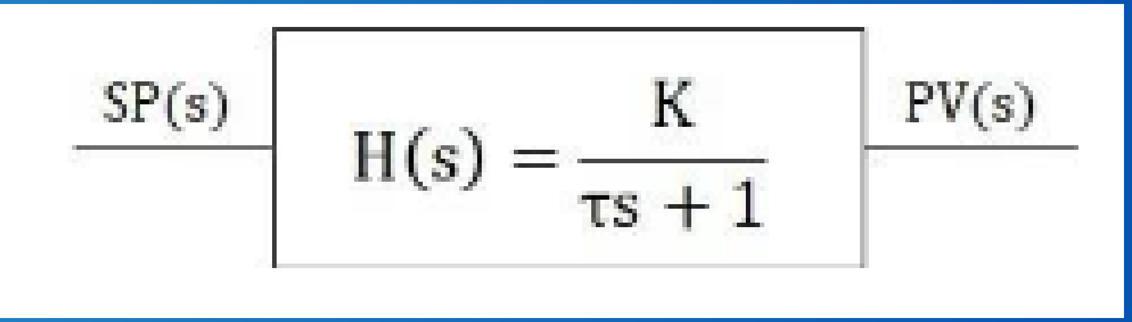
Original em relação a estimada:

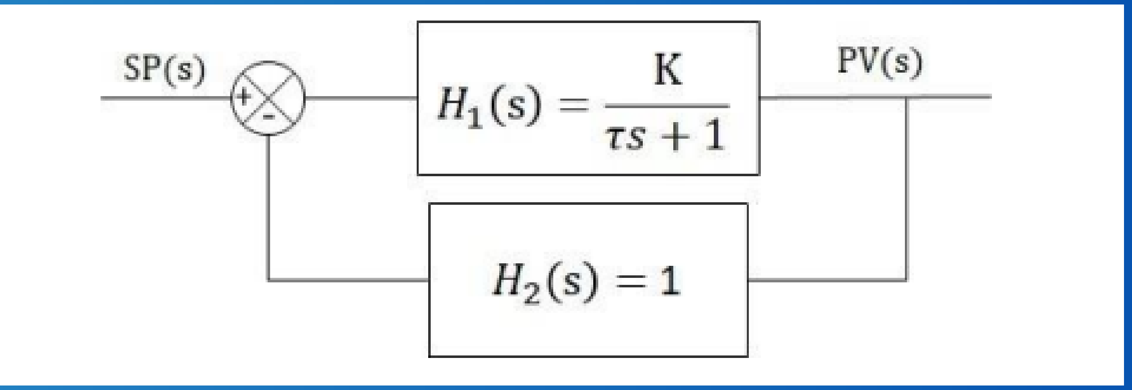
Plotando os gráficos:



Valores de erro planta aberta e fechada:








```
>> sys = tf([1], [1.95 1])
sys_aberta = sys
set(sys_aberta, 'InputDelay',0.55)
sys_fechada = feedback(sys_aberta,1)
set(sys_fechada, 'InputDelay',0.55)
step(sys_aberta*9, 'y-',sys_fechada*9,'c--')
hold on
legend('malha_aberta','malha_fechada')
grid on
```

Método Antigo CHR 1:

- O método CHR (Chien, Hrones, e Reswick) ,foi proposto por Y.C. Chien, A.E. Hrones e S.R. Reswick em 1952.

- O primeiro critério do método CHR é buscar uma resposta mais rápida do sistema. A resposta mais rápida refere-se ao tempo necessário para que o sistema atinja o valor final desejado a partir do momento em que a entrada é aplicada.

- As sintonias são obtidas tanto para o problema servo (mudança de valor do setpoint) como para o problema regulatório (perturbação de carga com setpoint constante).

- Em sistemas de controle, especialmente em aplicações onde a precisão é crítica, o sobrevalor é geralmente indesejado, pois pode levar a oscilações e instabilidade no sistema.

Método Novo Cohen e Coon:

- Em 1953, Cohen e Coon propuseram ajustes aos parâmetros do PID baseados na curva de reação. Apresentando bons resultados para sistemas com tempo morto elevado.
- As equações para a sintonia são relacionadaspara sistemas aproximados de primeira ordem

Calculando Controlador CHR PID

→ CHR sem Sobrevalor:

$Kp = (0.6*\tau)/(K*\Theta)$

$Kp = (0.6*1.95)/(1*0.55)$

$Kp = 2.127$

$Ti = \tau$

$\tau = 1.95$

$Ti = 1.95$

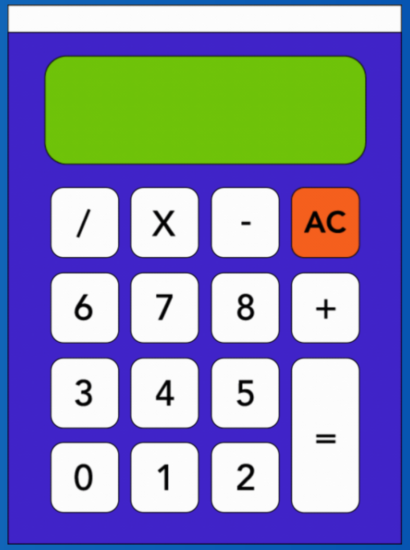
$Td = 0.5*\Theta$

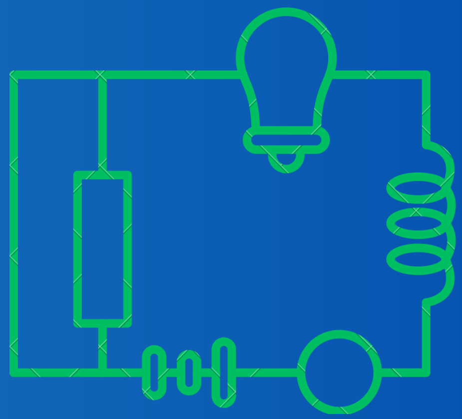
$Td = 0.5*0.55$

$Td = 0.275$

Tabela 3 – CHR sem Sobrevalor (Problema Servo)

Controlador	K_p	T_i	T_d
P	$\frac{0.3\tau}{K\theta}$	-	-
PI	$\frac{0.35\tau}{K\theta}$	1.16τ	-
PID	$\frac{0.6\tau}{K\theta}$	τ	0.5θ





Calculando Controlador CHR PID

→ Método Cohen e Coon para Curva de Reação:

$Kc = (1/K)*(\tau/\theta)*[(4/3)+(1/4)*(\theta/\tau)]$

$Kc = (1/1)*(1.95/0.55)*[(4/3)+(1/4)*(0.55/1.95)]$

$Kc = 4.977$

$Ti = \theta*[(32+6*(\theta/\tau))/(13+8*(\theta/\tau))]$

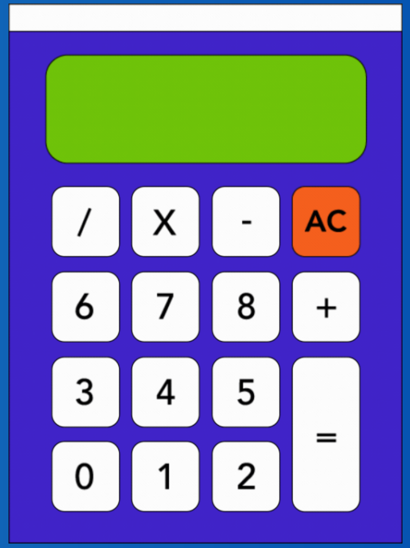
$Ti = 0,55*[(32+6*(0.55/1.95))/(13+8*(0.55/1.95))]$

$Ti = 1.214$

$Td = \theta*[(4)/(11+2*(\theta/\tau))]$

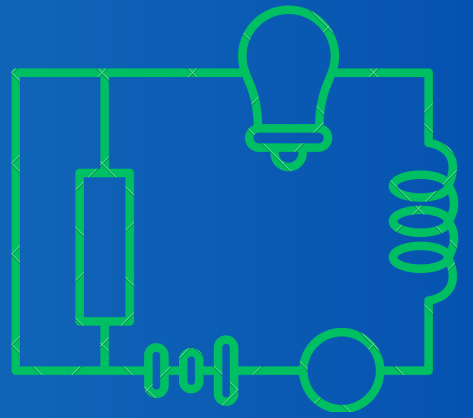
$Td = 0.55*[(4)/(11+2*(0.55/1.95))]$

$Td = 0.19$

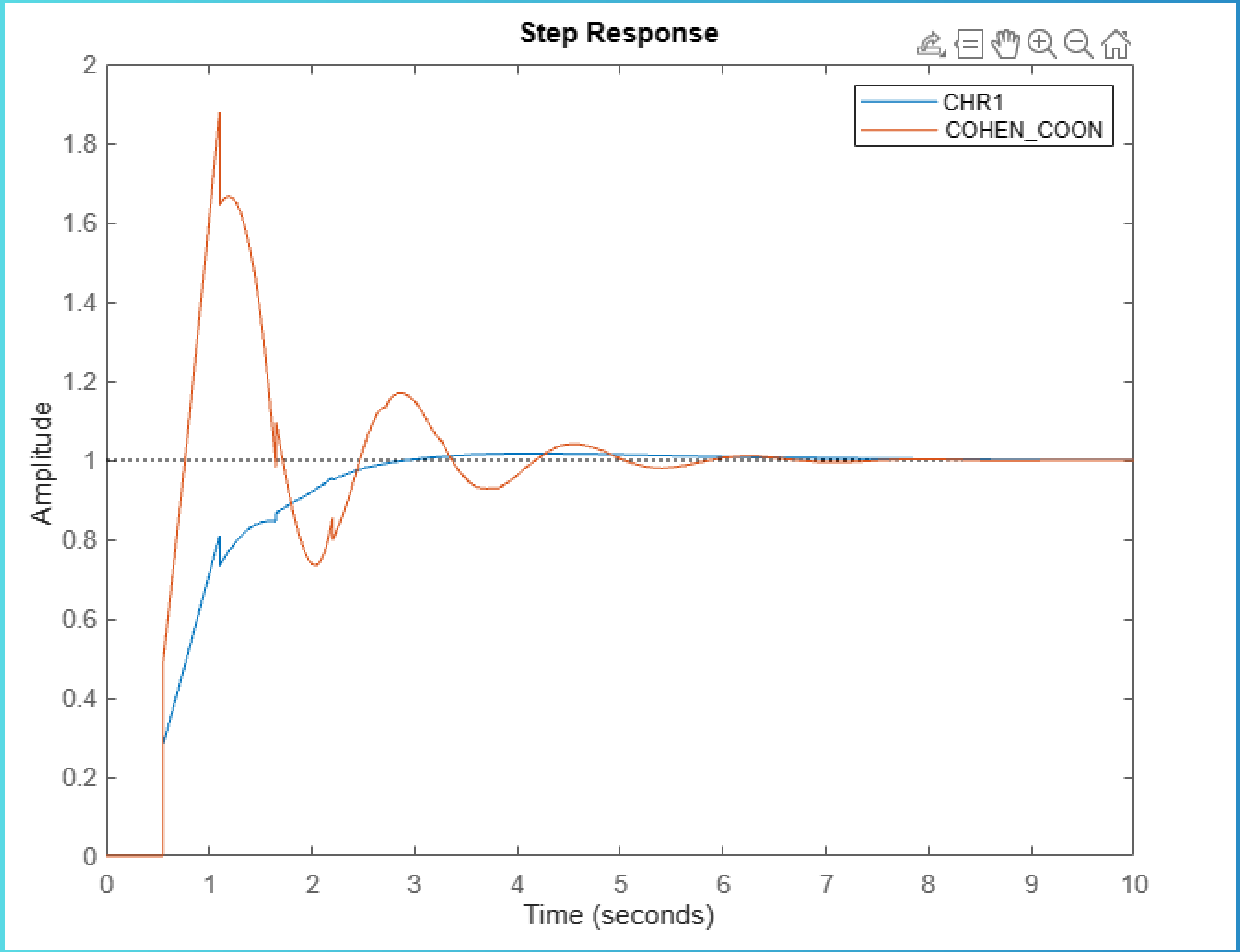


Método Cohen e Coon para Curva de Reação

Tipo de Controlador	K_c	τ_I	τ_D
P	$\frac{1}{k}(\frac{\tau}{\theta})[1+\frac{1}{3}(\frac{\theta}{\tau})]$		
PI	$\frac{1}{k}(\frac{\tau}{\theta})[.9+\frac{1}{12}(\frac{\theta}{\tau})]$	$\theta\left[\frac{30+3(\frac{\theta}{\tau})}{9+20(\frac{\theta}{\tau})}\right]$	
PID	$\frac{1}{k}(\frac{\tau}{\theta})[\frac{4}{3}+\frac{1}{4}(\frac{\theta}{\tau})]$	$\theta\left[\frac{32+6(\frac{\theta}{\tau})}{13+8(\frac{\theta}{\tau})}\right]$	$\theta\left[\frac{4}{11+2(\frac{\theta}{\tau})}\right]$



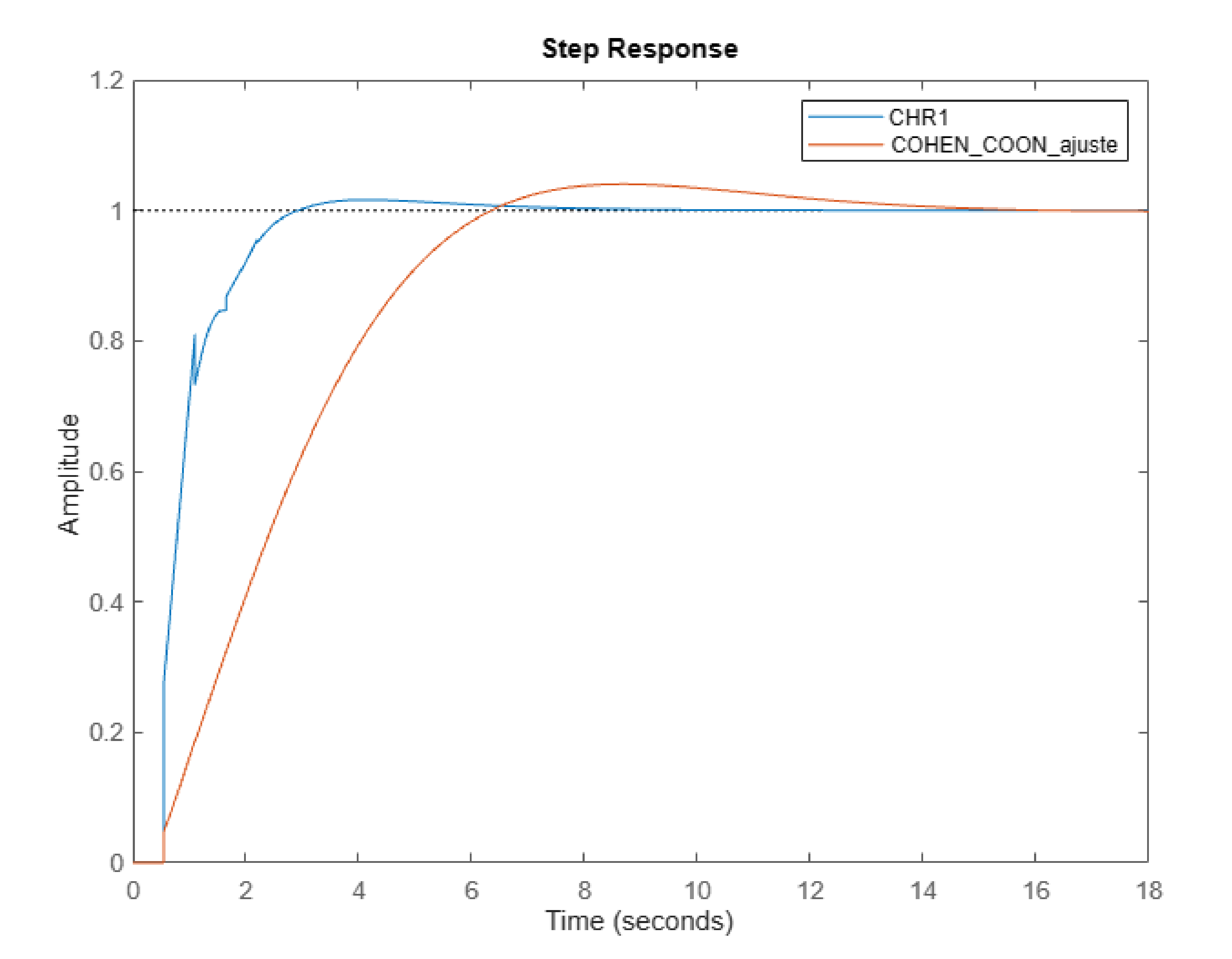
Plotando o gráfico do controlador




```
>> func = tf([1], [1.95 1], 'InputDelay', 0.55);
PIDCHR1 = pidstd(2.127, 1.95, 0.255);
PIDCOHEN_COON = pidstd(4.977, 1.214, 0.19);
resposta_CHR1 = feedback(func * PIDCHR1, 1);
resposta_COON = feedback(func * PIDCOHEN_COON, 1);
hold on
grid on
step(resposta_CHR1, resposta_COON)
legend('CHR1', 'COHEN_COON')
```

Plotando o gráfico com ajuste fino

Como o Cohen_Coon está com o overshoot muito alto (próximo de 90%) diminuímos o Kc do Cohen_Coon em 10 vezes (Kc = 0.4977).




```
>> func = tf([1], [1.95 1], 'InputDelay', 0.55);
PIDCHR1 = pidstd(2.127, 1.95, 0.255);
PIDCOHEN_COON = pidstd(0.4977, 1.214, 0.19);
resposta_CHR1 = feedback(func * PIDCHR1, 1);
resposta_COON = feedback(func * PIDCOHEN_COON, 1);
hold on
grid on
step(resposta_CHR1, resposta_COON)
legend('CHR1', 'COHEN_COON_ajuste')
```

Considerações dos métodos