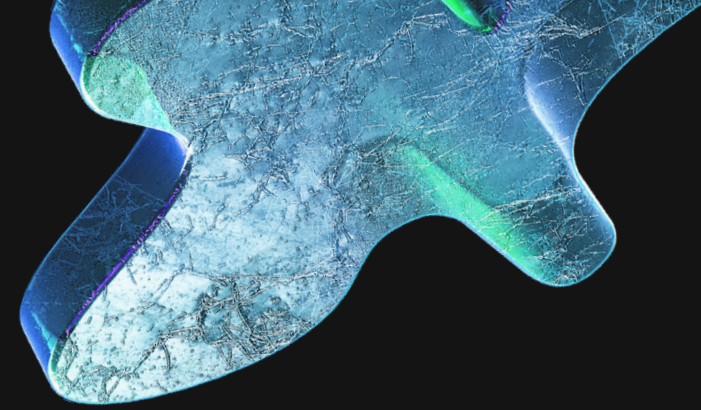


C213 – Sistemas Embarcados



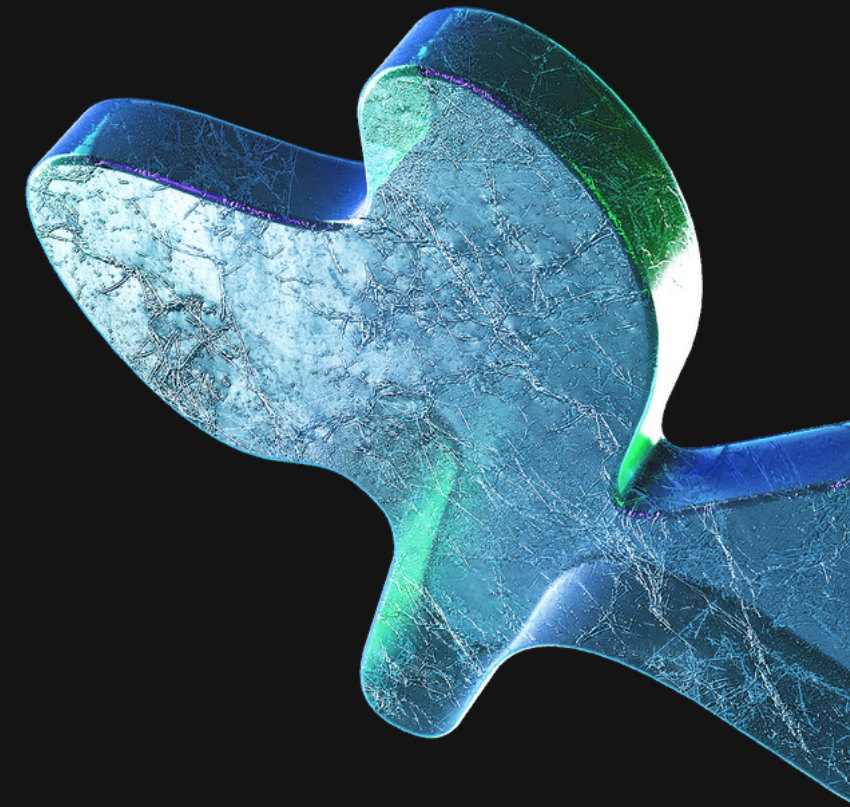
Controladores Clássicos

Grupo 13

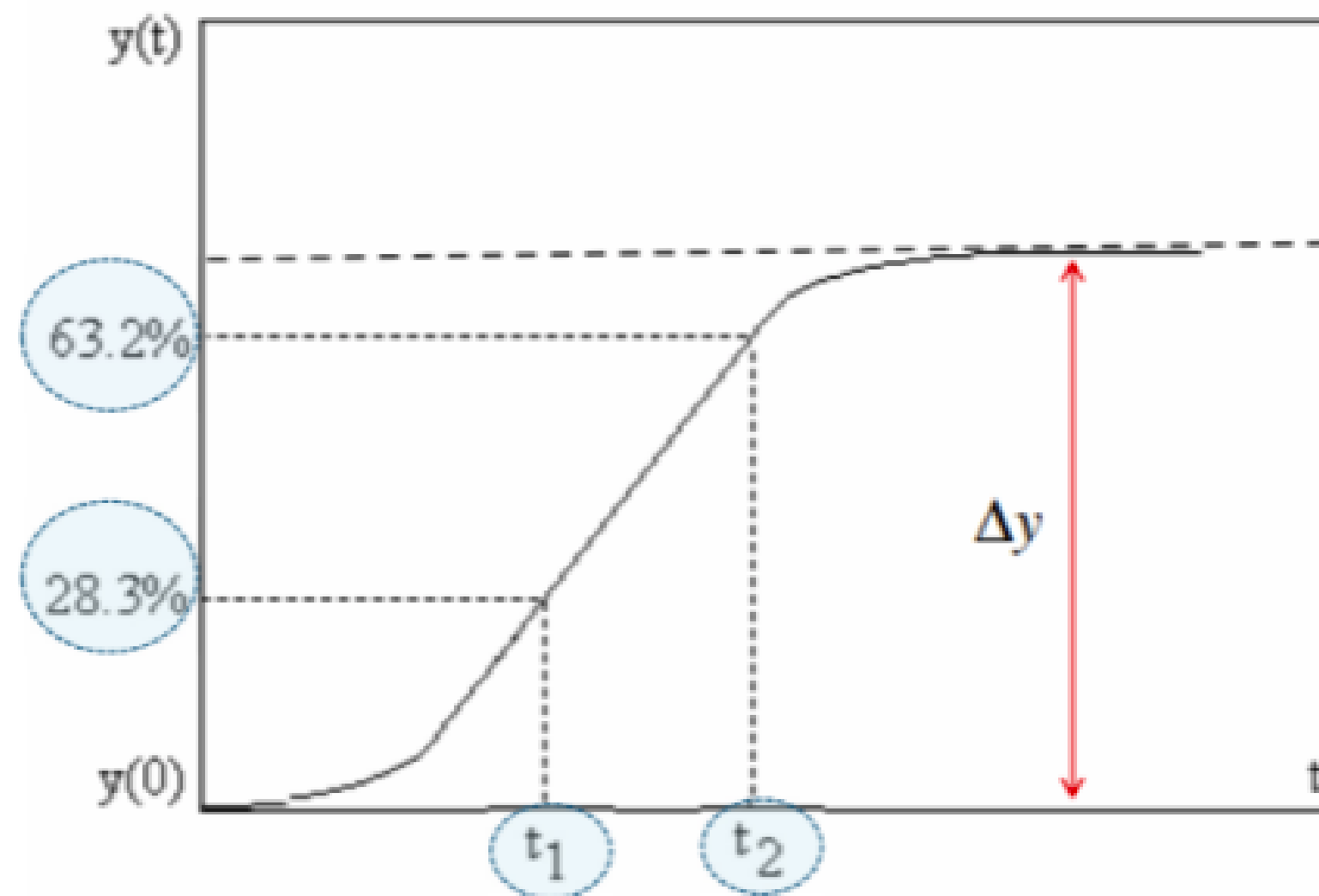
Fernanda Nagata Ito – 1759

João Henrique Silva Delfino – 1662

Paulo Otavio Luczensky de Souza – 1732



Método de Smith



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

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

$$\theta = t_2 - \tau$$

Aplicando o método de Smith

```
% defining transfer function using smith method
delta_output = saida(end) - saida(1);
delta_input = degrau(1);

% getting k
k = delta_output / delta_input;

% getting output values for t1 and t2
v_t1 = 0.283 * delta_output;
v_t2 = 0.632 * delta_output;

% getting t1 and t2 values using array values
t1 = t(16);
t2 = t(36);

% getting tal value
tal = 1.5 * (t2 - t1);

% getting theta value
theta = t2 - tal;
```

$$\Delta u = 13$$

$$\Delta y = 19.443$$

$$k = 1.4956$$

$$\theta = 0.5 \text{ segundos}$$

$$\tau = 3 \text{ segundos}$$

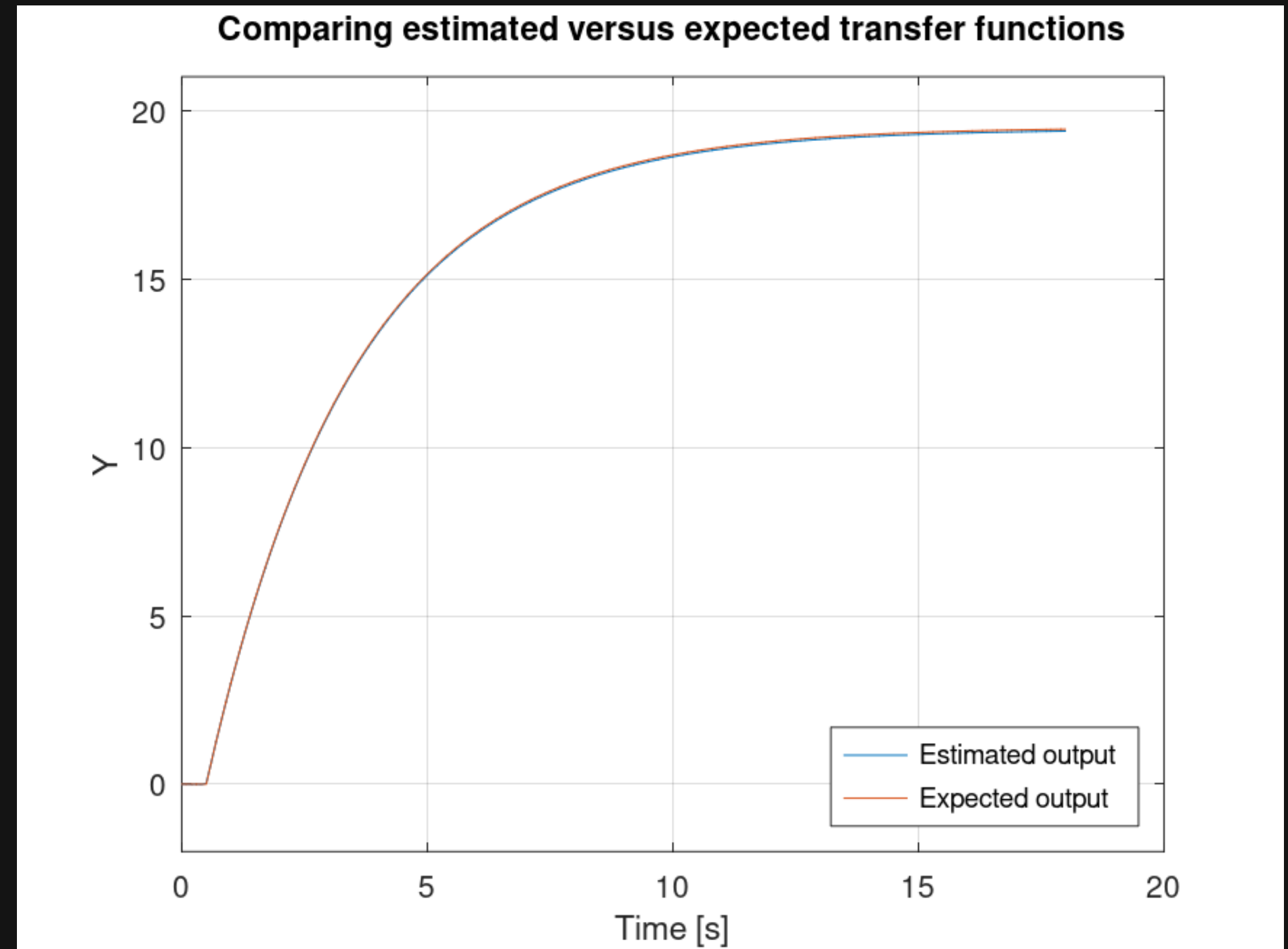
Estimando a função de transferência do sistema

```
% system without delay
sys_without_delay = tf([k], [tau 1]);

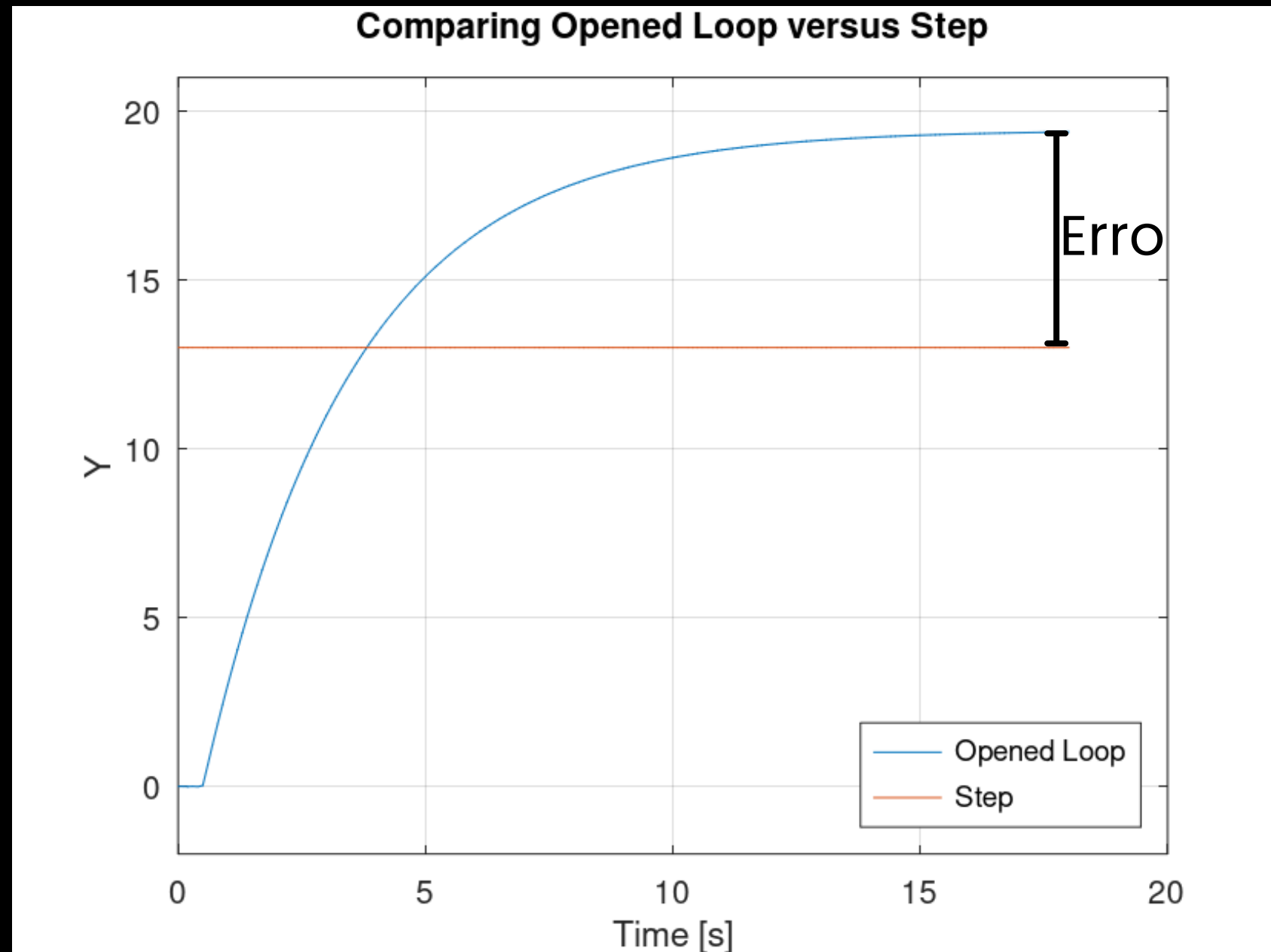
% creating delay
[num, den] = padcoef(theta, 20);
h_pade = tf(num, den);

% inserting delay on system
sys_opened_loop = sys_without_delay * h_pade;

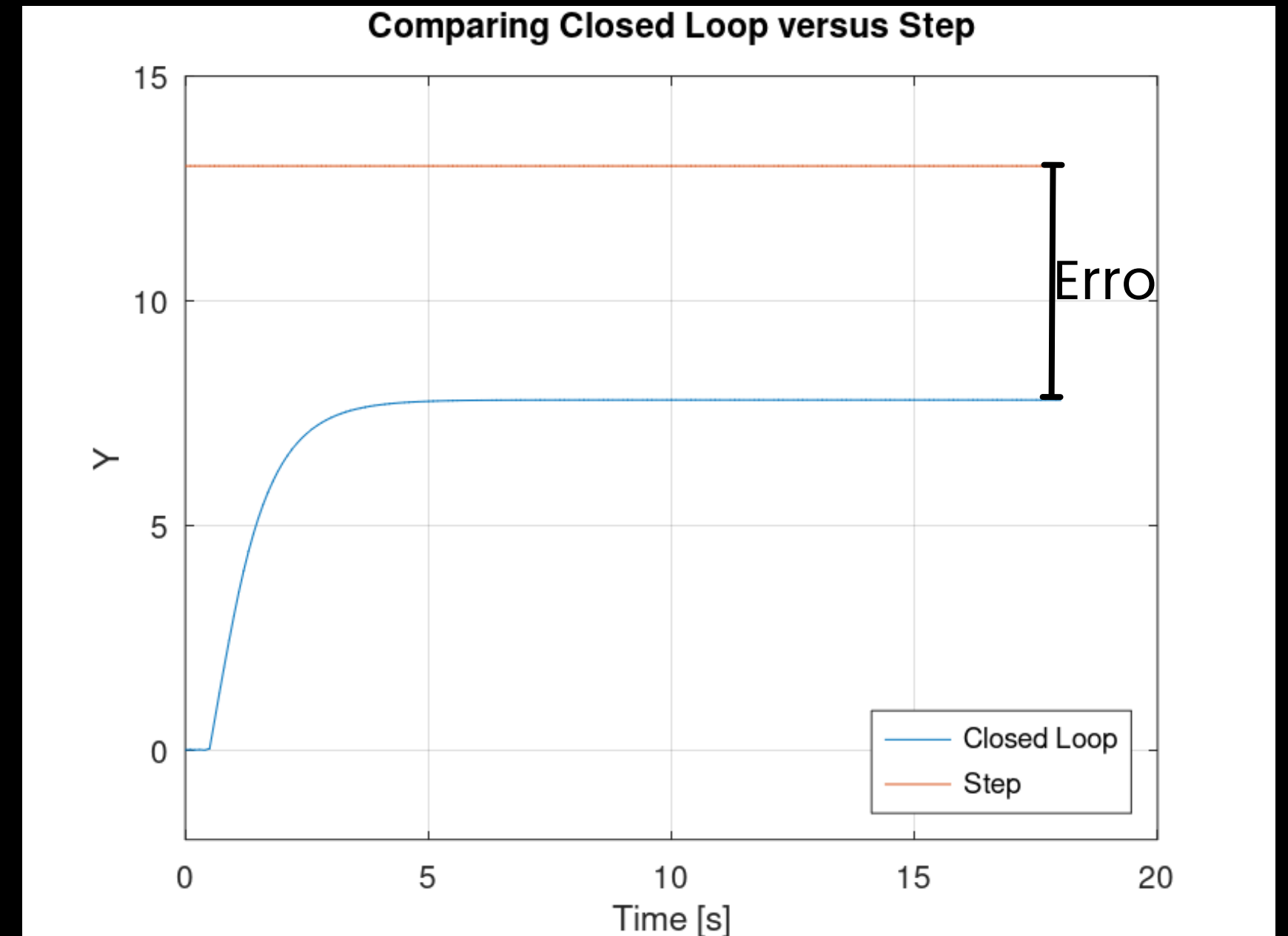
% using step function in order to get output estimated values
[output_opened_loop, t, x] = step(delta_input * sys_opened_loop, t);
```



Malha aberta e malha fechada



$$\text{Erro} = -6.386$$



$$\text{Erro} = 5.2092$$

Métodos de sintonia de PID

CHR – sem Sobrevalor

- Proposto por Chien, Hron e Reswick, em 1952.
- Esse critério visa minimizar o tempo de subida (tempo que o sistema leva para atingir o valor desejado sem ultrapassar).
- Em termos práticos, para sistemas de controle, isso significa que o controlador ajustado deve fazer com que o sistema atinja o valor desejado o mais rápido possível, sem criar uma sobrevalorização significativa na resposta.

Integral do Erro Absoluto

- O objetivo do Método IAE é minimizar a integral do erro absoluto ao longo do tempo.
- No contexto de controle PID, o "erro" refere-se à diferença entre o valor desejado (setpoint) e o valor atual medido do processo controlado.
- A sintonia de um controlador PID usando o Método IAE envolve experimentação e ajustes iterativos dos parâmetros até que o desempenho desejado seja alcançado.

CHR sem sobrevalor

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

```
% P controller
kp = (0.3 * tal) / (k * theta);

sys_p = pidstd(kp);

sys_chr_p = feedback(sys_opened_loop*sys_p, 1);
```

P

```
% PI controller
kp = (0.35 * tal) / (k * theta);
ti = 1.16 * tal;

sys_pi = pidstd(kp, ti);

sys_chr_pi = feedback(sys_opened_loop*sys_pi, 1);
```

PI

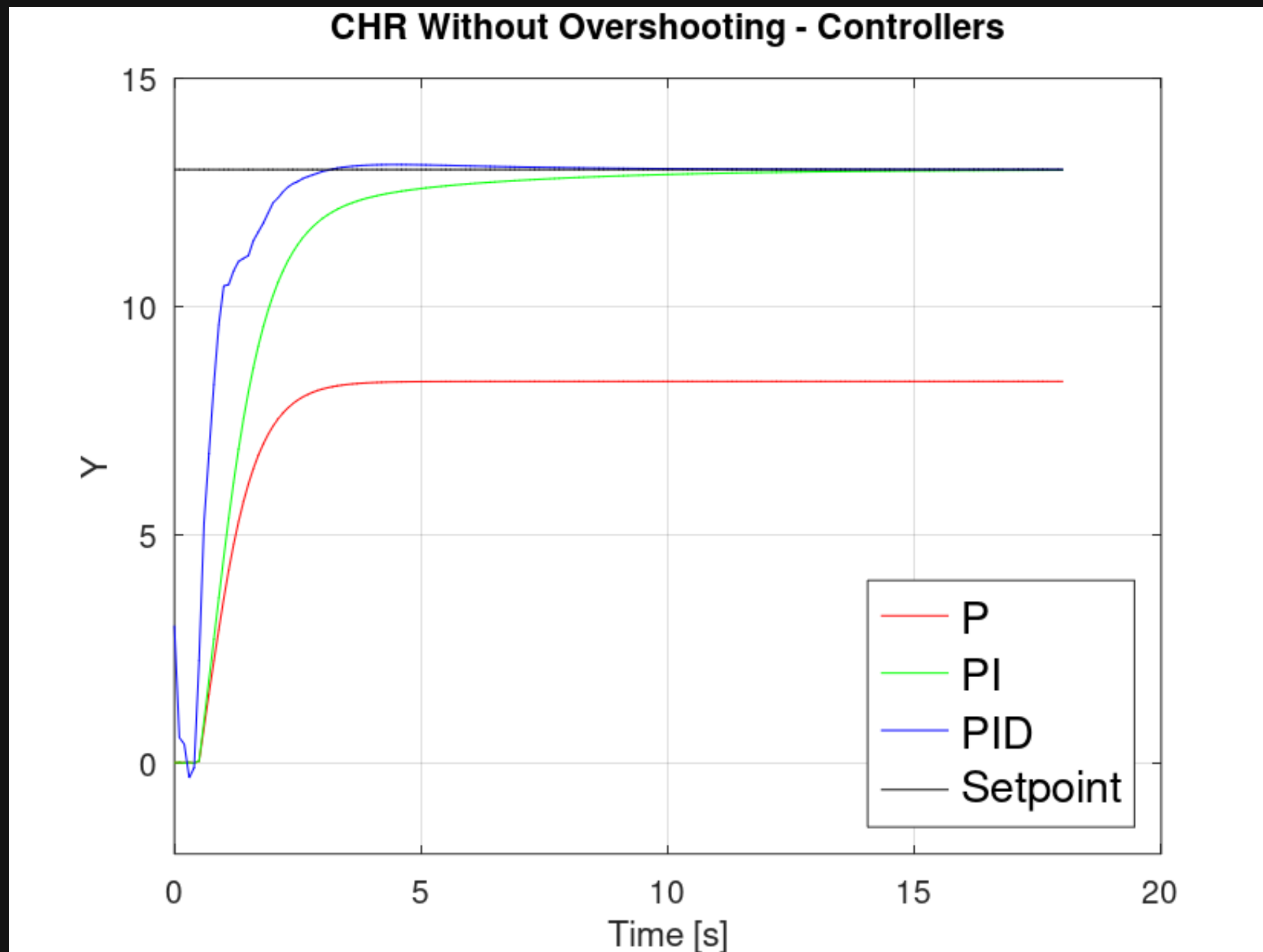
```
% PID controller
kp = (0.6 * tal) / (k * theta);
ti = 1 * tal;
td = 0.5 * theta;

sys_pid = pidstd(kp, ti, td);

sys_chr_pid = feedback(sys_opened_loop*sys_pid, 1);
```

PID

Resultados do método CHR sem sobrevalor



PID

$$k_p = 2.407$$

$$t_i = 3$$

$$t_d = 0.25$$

$$\text{Setpoint} = 13$$

Integral do Erro absoluto (IAE)

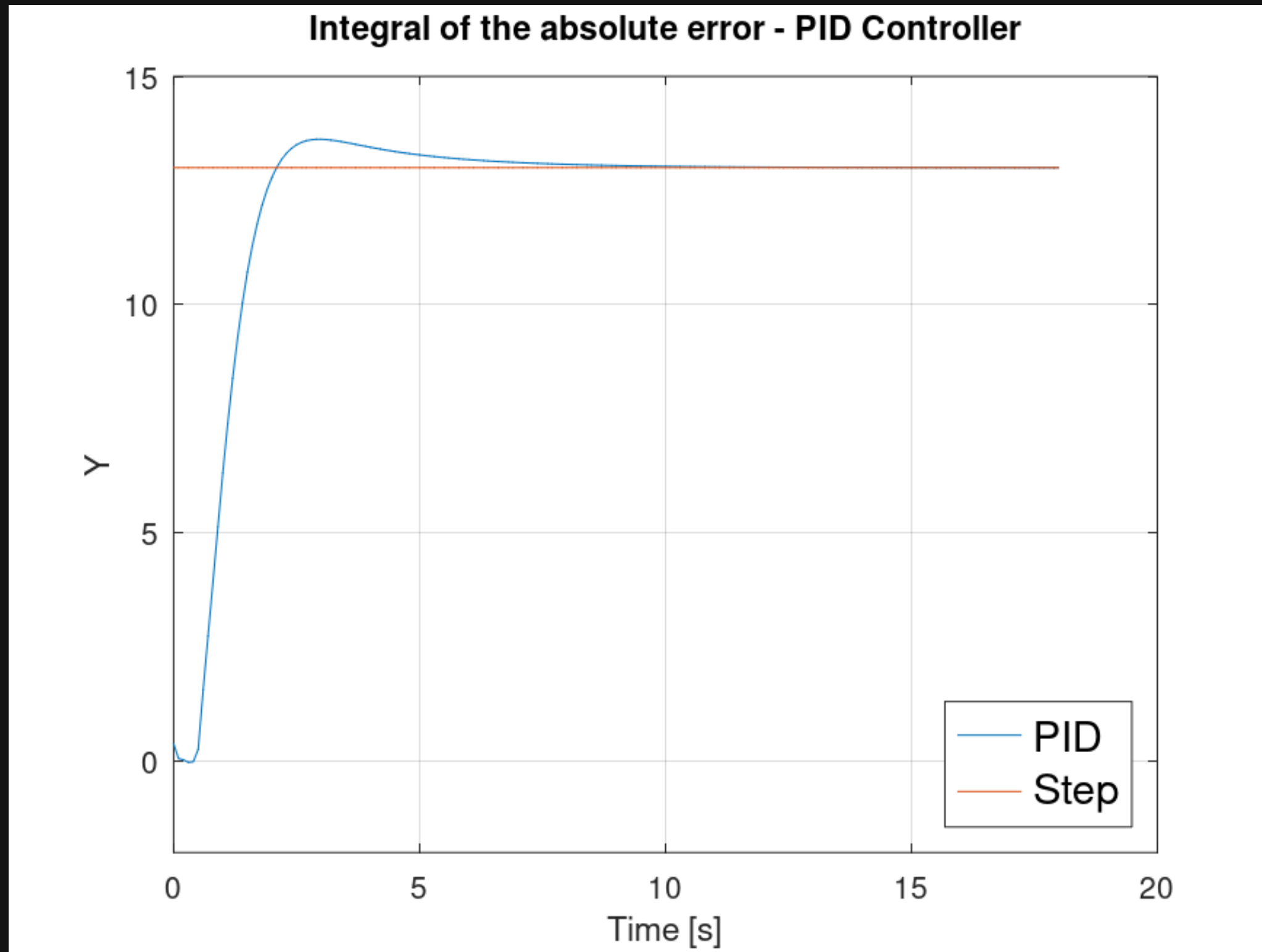
Fator Adimensional	IAE
$K_p \times K$	$1/((\theta/\tau) + 0.2)$
T_i/θ	$(0.3 \times (\theta/\tau) + 1.2) / ((\theta/\tau) + 0.08)$
T_d/θ	$1 / (90 \times (\theta/\tau))$

```
% PID Controller
kp = (1 / ((theta / tal) + 0.2)) / k;
ti = ((0.3 * (theta / tal) + 1.2) / ((theta / tal) + 0.08)) * theta;
td = (1 / (90 * (theta / tal))) * theta;

sys_pid = pidstd(kp, ti, td);

sys_integral_pid = feedback(sys_opened_loop*sys_pid, 1);
```

Resultados do método da Integral do Erro Absoluto – IAE



PID

$$k_p = 1.8235$$

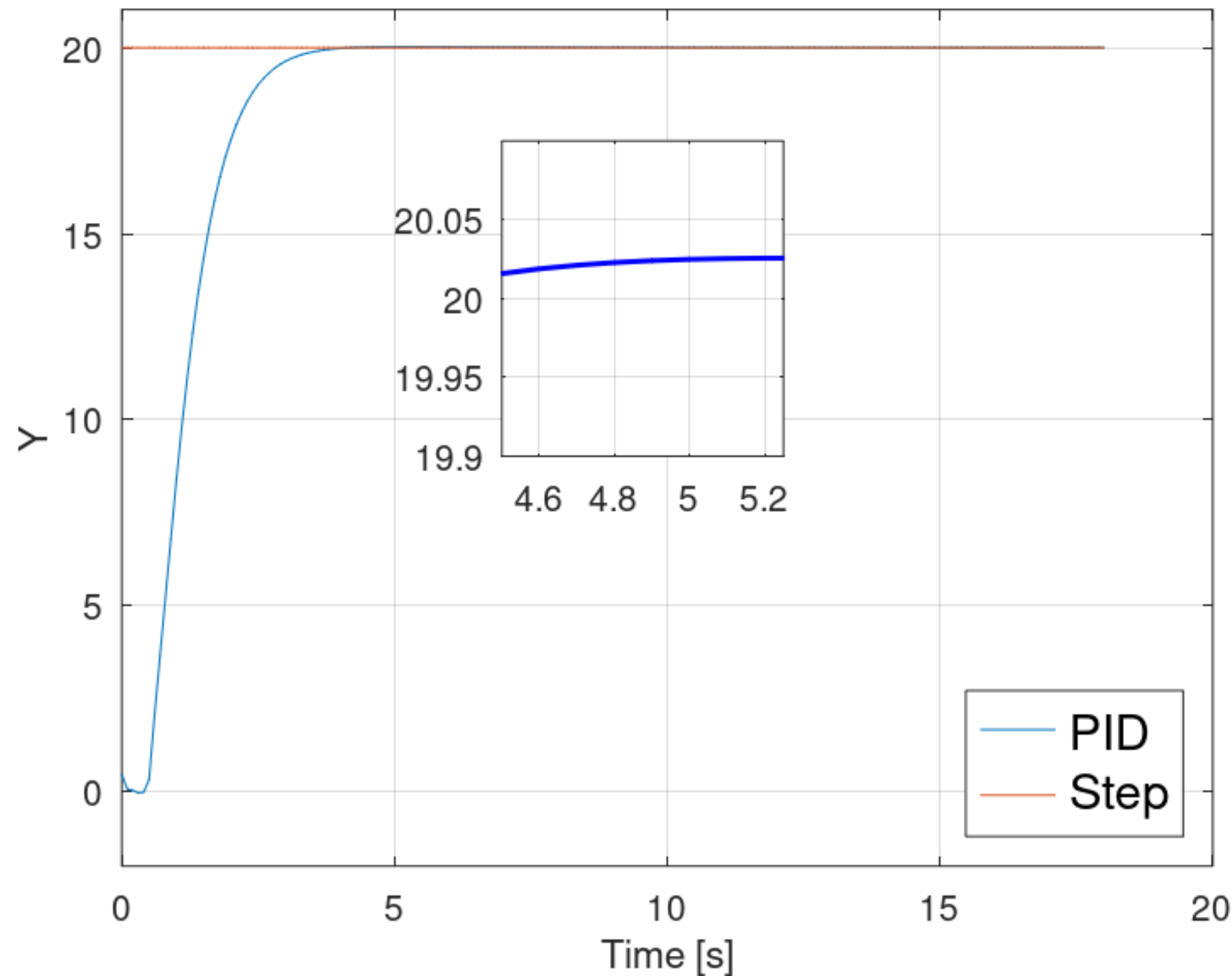
$$t_i = 2.5338$$

$$t_d = 0.033$$

$$\text{Setpoint} = 13$$

Menu do usuário

Constants from user input - PID Controller



```
% Entry values
kp = input("Kp value: ");
ti = input("Ti value: ");
td = input("Td value: ");
setpoint = input("Setpoint value: ");

sys_pid = pidstd(kp, ti, td);
sys_series_pid = feedback(sys_opened_loop*sys_pid, 1);
```

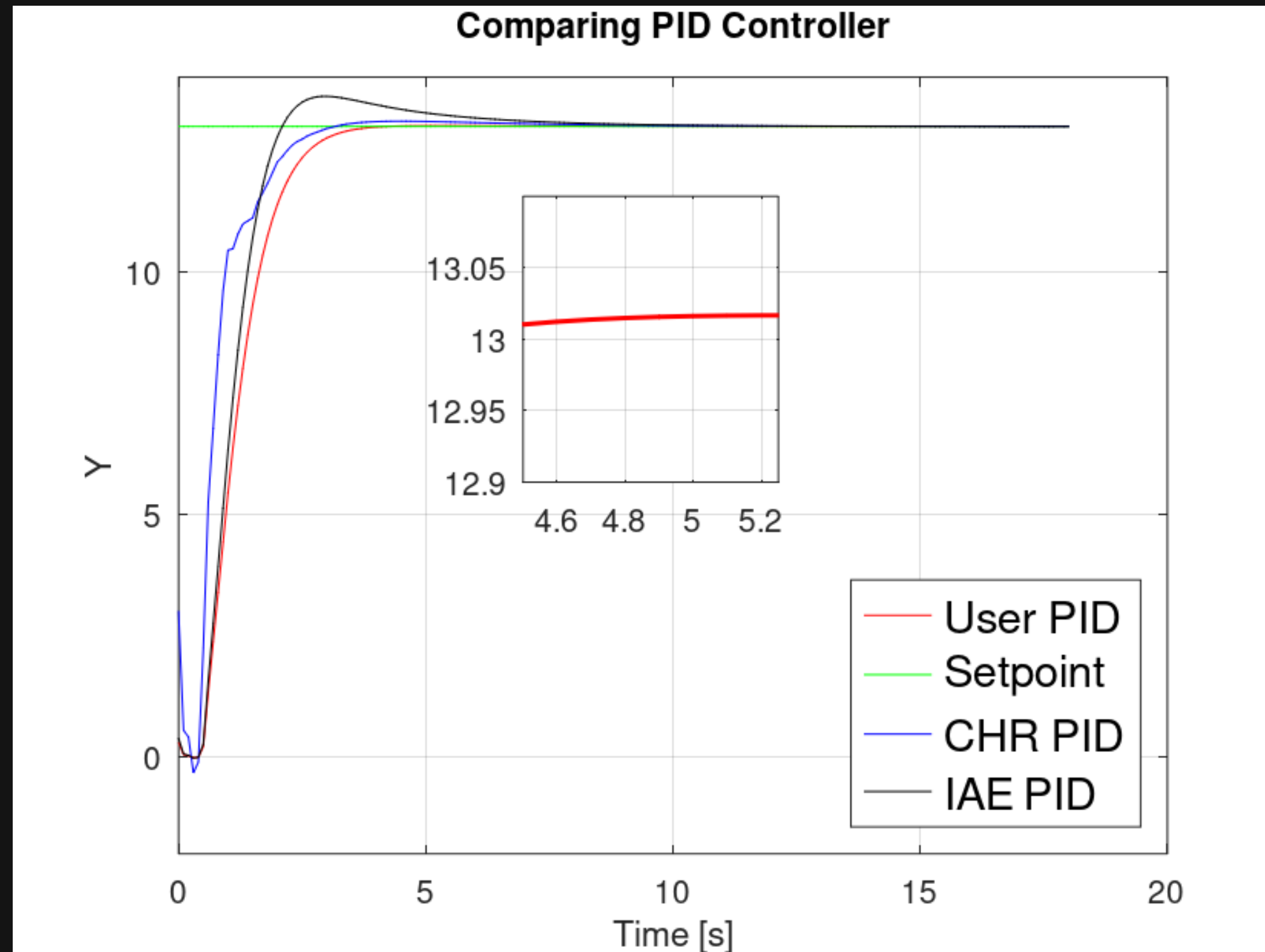
$kp = 1.6$

$ti = 3$

$td = 0.03$

Setpoint = 20

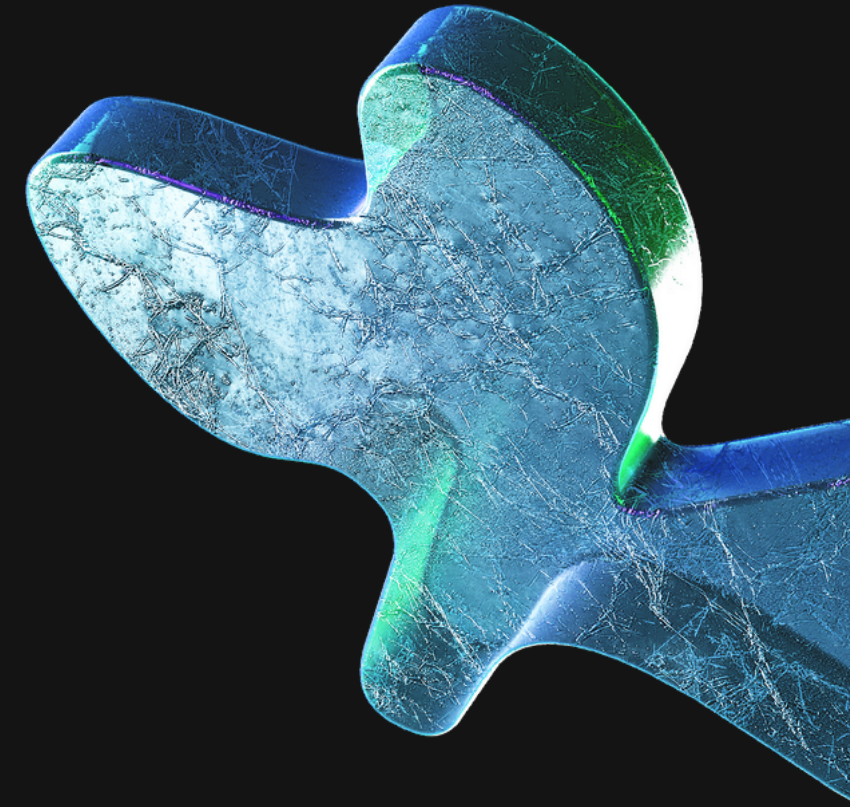
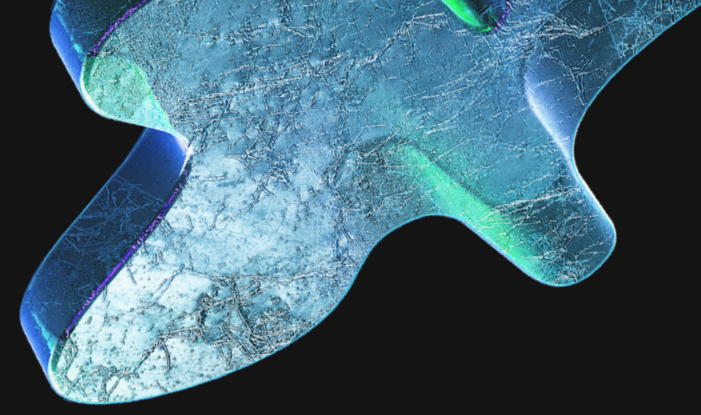
Comparativo entre os métodos



Conclusões

- Os métodos servem como apoio para encontrar os valores das constantes;
- O método IAE apresentou um overshooting maior que o método clássico CHR sem sobrevalor (comportamento esperado);
- O método CHR apresentou uma certa oscilação durante a subida, mas atingiu a estabilidade antes do método IAE;
- As constantes fornecidas pelo usuários foram as que apresentaram um menor tempo para estabilidade, mas bem próximo aos dos métodos;
- É possível realizar o autotuning do processo utilizando ferramentas de IA, por exemplo;

Obrigado!





Link do GitHub:

<https://github.com/Joaohsd/classic-control-pid>

