

Aula dia 18/10 Controladores PID

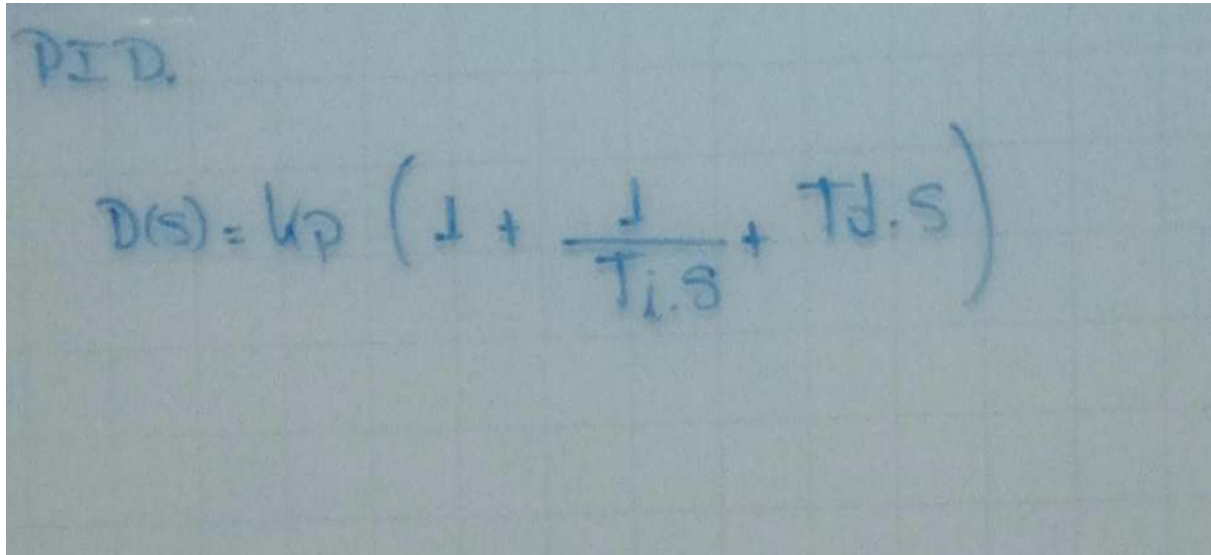
Nomes:

P: Controlador Proporcional

PI: Controlador Proporcional Integral

PID: Controlador Proporcional Integral Derivativo

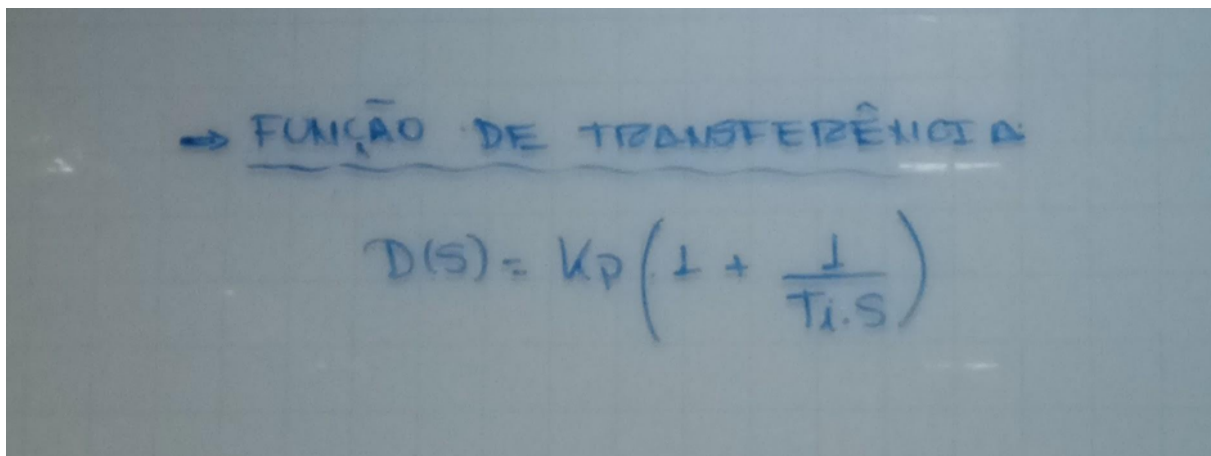
Lembrar a equação da função de transferência PID



PID.

$$D(s) = K_p \left(1 + \frac{1}{T_i \cdot s} + T_d \cdot s \right)$$

Função da transferência PI



→ FUNÇÃO DE TRANSFERÊNCIA

$$D(s) = K_p \left(1 + \frac{1}{T_i \cdot s} \right)$$

Exercício: em um controlador PID, o ganho proporcional $K_p = 20$, o tempo da parte integral $T_i = 0,1s$ e o tempo da parte derivativa é $T_d = 0,01$. Determinar a Função de Transferência desse controlador PID.

Resolução do professor:

Resolvendo o controlador PID

EXERCÍCIOS:

EM UM CONTROLADOR PID, O GANHO PROPORCIONAL $K_P = 20$, O TEMPO DA PARTE INTEGRAL $T_I = 0,1s$ E O TEMPO DA PARTE DERIVATIVA É $T_D = 0,01s$. DETERMINAR A FUNÇÃO DE TRANSFERÊNCIA DESSE CONTROLADOR PID.

$$D(s) = K_P \left(1 + \frac{1}{T_I \cdot s} + T_D \cdot s \right)$$

$$D(s) = K_P + \frac{K_P}{T_I \cdot s} + K_P \cdot T_D \cdot s$$

$$D(s) = 20 + \frac{20}{0,1 \cdot s} + 20 \cdot 0,01 \cdot s$$

$$D(s) = 20 + \frac{200}{s} + 0,2 \cdot s$$

$$D(s) = \frac{20s + 200 + 0,2 \cdot s^2}{s}$$

$$D(s) = \frac{0,2 \cdot s^2 + 20s + 200}{s}$$

Exercício 2:

Um controlador PI possui um ganho proporcional $K_P = 30$ e o tempo de ação da parte integral $T_I = 0,01s$. Determinar a Função de Transferência desse controlador PI.

Resolução

→ FUNÇÃO DE TRANSFERÊNCIA

$$D(s) = K_p \left(1 + \frac{1}{T_i \cdot s} \right)$$

$$D(s) = 30 \cdot \left(1 + \frac{1}{0,01 \cdot s} \right) \Rightarrow D(s) = 30 + \frac{30}{0,01 \cdot s}$$

$$D(s) = 30 + \frac{30 \times 100}{s} \Rightarrow D(s) = 30 + \frac{3000}{s}$$

$$D(s) = \frac{30s + 3000}{s}$$

Conteúdo

Método de Ziegler-Nichols

Método de Ziegler-Nichols

- Um controlador PID pode ser sintonizado pela seleção de valores apropriados dos parâmetros K_p , T_i , T_d .
- Um dos métodos mais simples de seleção dos valores K_p , T_i e T_d é a utilização do método de Ziegler-Nichols.
- Ziegler-Nichols sugerem que os parâmetros do controlador PID podem ser obtidos pelo teste da planta em malha aberta ou malha fechada.

Qualquer planta pode ser aproximada a essa Função de Transferência

Fator de atraso é uma função exponencial ou do primeiro grau

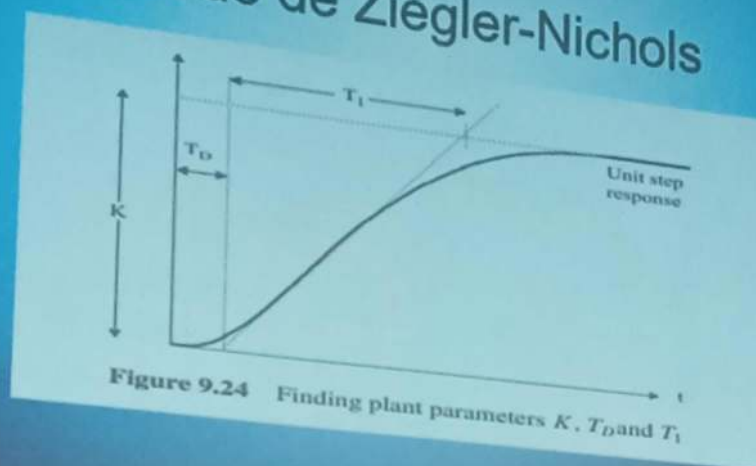
Método de Ziegler-Nichols

$$G(s) = \frac{K e^{-sT_D}}{1 + sT_1},$$

- De acordo com Ziegler-Nichols, a função de transferência de um sistema pode ser aproximada com um atraso de tempo e um sistema de 1ª ordem:
- TD = atraso de tempo do sistema (tempo de transporte);
- T1 = constante de tempo do sistema;

Curva de resposta da planta quando aplicado a uma função degrau unitário

Método de Ziegler-Nichols



Atenção: "... Determinar os parâmetros TD, T1 e K"

Após encontrar o TD, T1 e K

Método de Ziegler-Nichols

Table 9.1 Open-loop Ziegler-Nichols settings

Controller	K_p	T_i	T_d
Proportional	$\frac{T_i}{K T_D}$		
PI	$\frac{0.9 T_i}{K T_D}$	$3.3 T_D$	
PID	$\frac{1.2 T_i}{K T_D}$	$2 T_D$	$0.5 T_D$

... dados pela tabela abaixo.

Escrito no quadro

	K_p	t_i	t_d
P \Rightarrow	$\frac{T_i}{K \cdot T_D}$		
PI \Rightarrow	$\frac{0.9 \cdot T_i}{K \cdot T_D}$	$3.3 \cdot T_D$	
PID \Rightarrow	$\frac{1.2 \cdot T_i}{K \cdot T_D}$	$2 \cdot T_D$	$0.5 \cdot T_D$

Exemplo:

Método de Ziegler-Nichols

Exemplo:

From Figure 9.25, the system parameters are obtained as $K = 40^\circ\text{C}$, $T_D = 5\text{ s}$ and $T_I = 20\text{ s}$, and the transfer function of the plant is

$$G(s) = \frac{40e^{-5s}}{1 + 20s}$$

Proportional controller: According to Table 9.1, the Ziegler-Nichols settings for a proportional controller are:

$$K_P = \frac{T_I}{K T_D}$$

Thus,

$$K_P = \frac{20}{40 \times 5} = 0.1$$

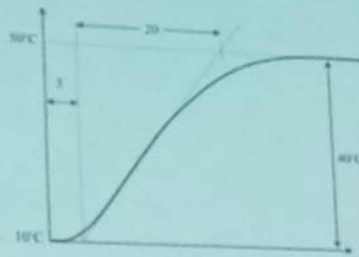


Figure 9.25 Unit step response of the system for Example 9.7

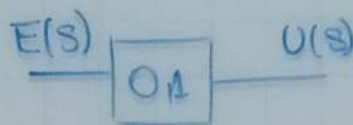
A partir do gráfico iremos determinar a Função de Transferência P, PI e PID

Resolução:

P:

⇒ PROJETAR O CONTROLADOR PROPORCIONAL:

$$K_P = \frac{T_I}{K \cdot T_D} \Rightarrow K_P = \frac{20}{40 \times 5} \Rightarrow \boxed{K_P = 0.1}$$



PI:

⇒ PROJETAR UM CONTROLADOR
PROPORCIONAL - INTEGRAL:

$$K_p = \frac{0,9 \times T_d}{k \times T_D} \Rightarrow K_p = \frac{0,9 \times 20}{40 \times 5}$$

$$K_p = \frac{0,9 \times 20}{200 \times 10} \Rightarrow \boxed{K_p = 0,09}$$

$$T_i = 3,3 \times T_D \Rightarrow T_i = 3,3 \times 5$$

$$\boxed{T_i = 16,5 \text{ s}}$$

$$D(s) = K_p + \frac{K_p}{T_i \cdot s}$$

$$D(s) = 0,09 + \frac{0,09}{16,5 \cdot s}$$

$$D(s) = \frac{0,09 \times 16,5 \text{ s} + 0,09}{16,5 \cdot s}$$

$$\boxed{D(s) = \frac{1,48 \cdot s + 0,09}{16,5 \cdot s}}$$

PID:

⇒ PROJETAR O
CONTROLADOR PID:

$$K_P = \frac{1,2 \times T_1}{K \cdot T_D}$$

$$K_P = \frac{1,2 \times 20}{40 \times 5}$$

$$K_P = 0,12$$

$$T_i = 2 \times T_D$$

$$T_i = 2 \times 5 \Rightarrow T_i = 10s$$

$$T_d = 0,5 \times 5 \Rightarrow T_d = 2,5s$$

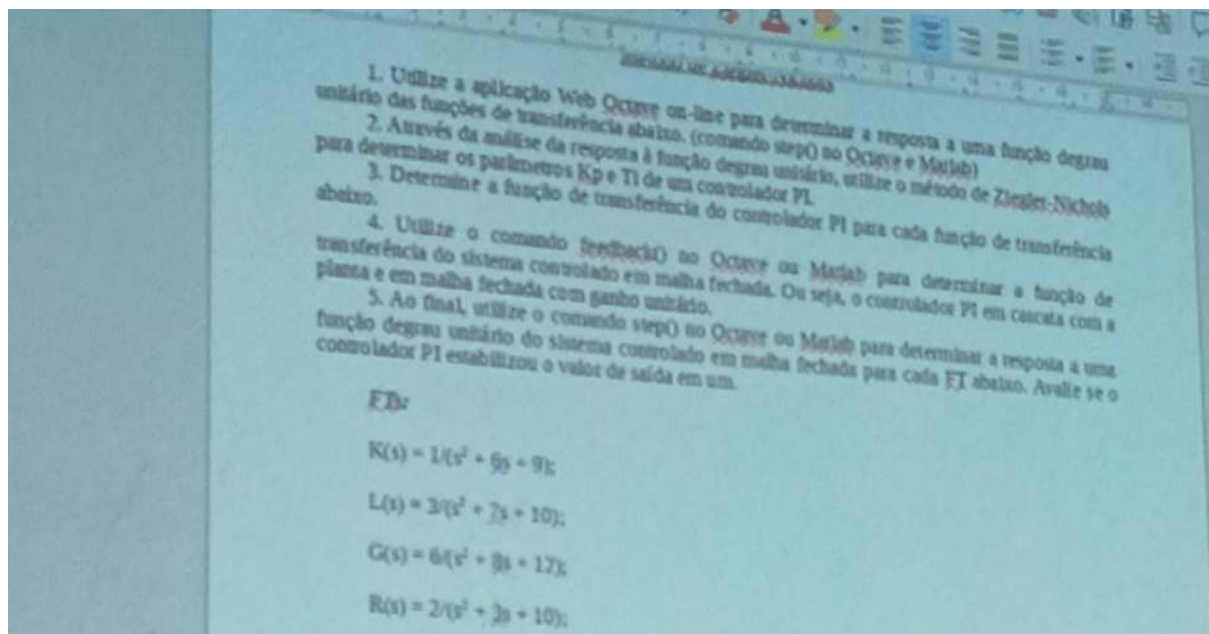
$$K_P = 0,12; \quad T_i = 10s; \quad T_d = 2,5s.$$

$$D(s) = K_P + \frac{K_P}{T_i \cdot s} + K_P \cdot T_d \cdot s$$

$$D(s) = 0,12 + \frac{0,12}{10 \cdot s} + 0,12 \times 2,5 \cdot s$$

$$D(s) = \frac{1,2s + 0,1 + 1,2 \times 2,5 \cdot s^2}{10s}$$

$$D(s) = \frac{3 \cdot s^2 + 1,2s + 0,1}{10s}$$



Temos as funções de transferência da letra A ($K(s)$) e B ($L(s)$)

$$K(s) = \frac{1}{s^2 + 6s + 9}$$

$$L(s) = \frac{3}{s^2 + 7s + 10}$$

```
Student License -- for use by stu
and perform academic research at I

>> syms s
>> numG = [0 0 1];
>> denG = [1 6 9];
>> G = tf(numG,denG)

G =

      1
-----
fx
```

```
>> denG = [1 6 9];
>> G = tf(numG,denG)

G =

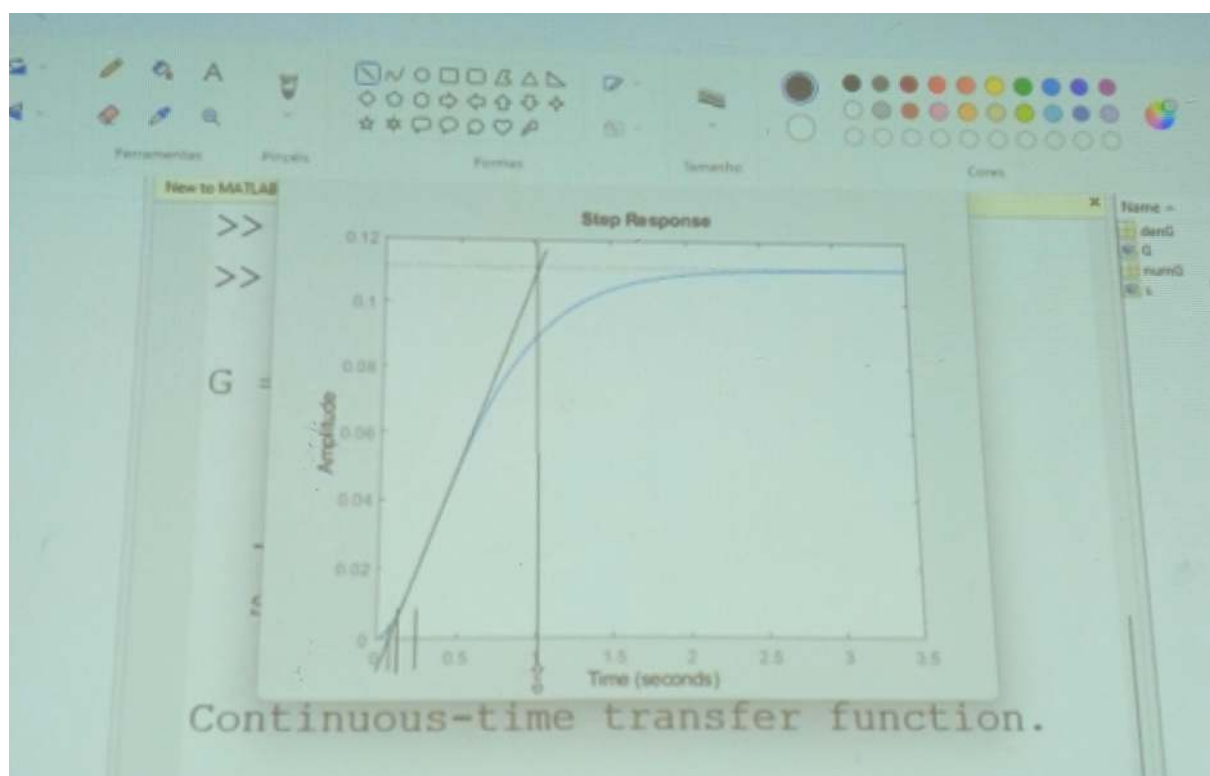
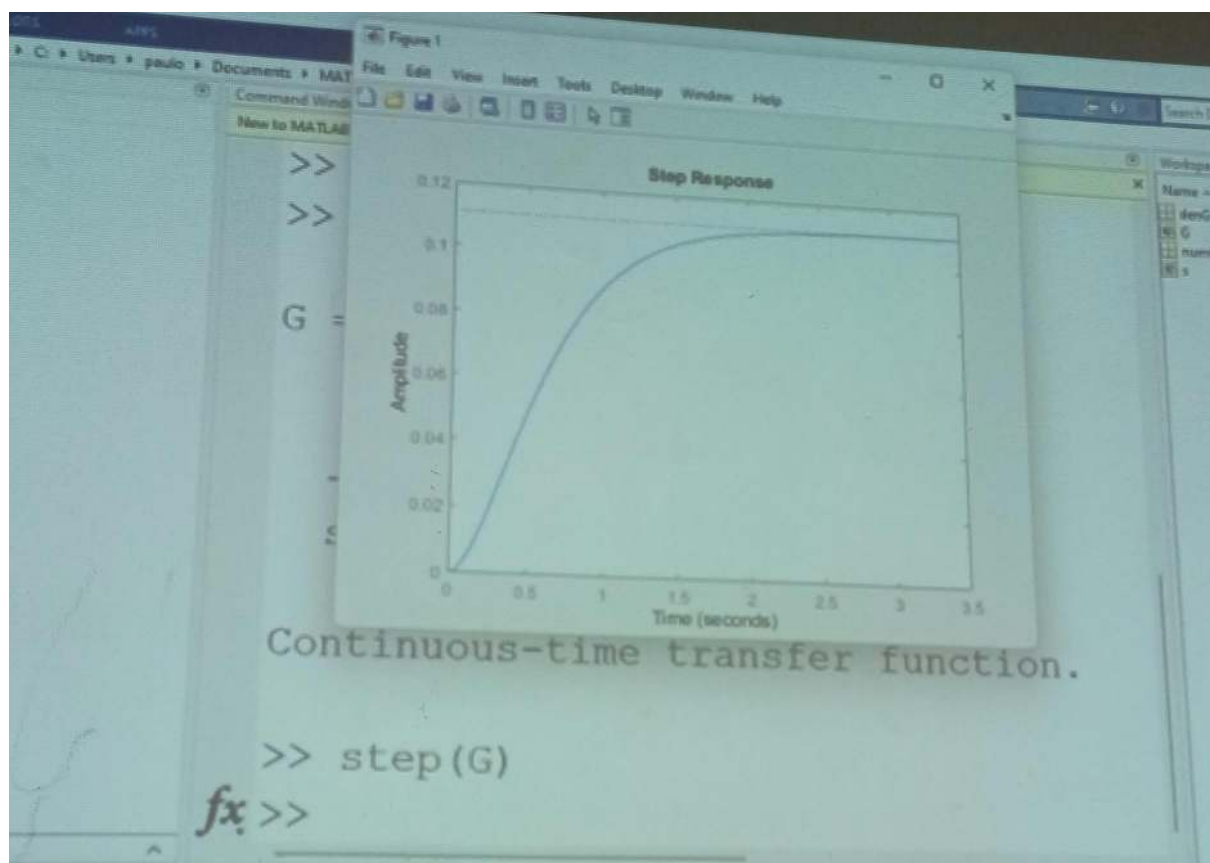
      1
-----
s^2 + 6 s + 9

Continuous-time transfer function.

>> step(G)

fx
```

Curva de saída



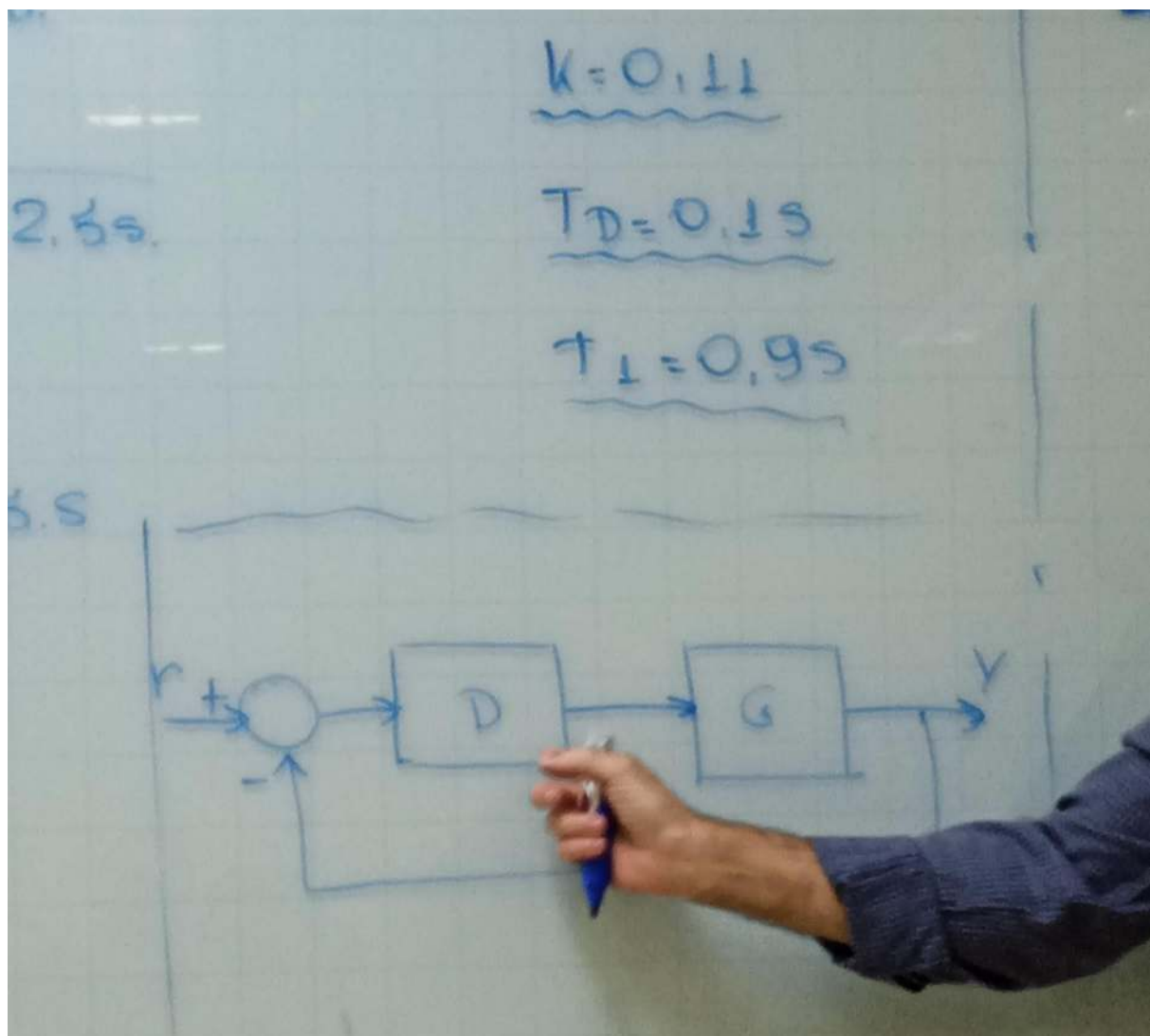
$K = 0,11$
 $T_D = 0,1 \text{ s}$
 $T_i = 0,9 \text{ s}$

Com base nisso,

Encontrando o controlador PI:

$$\begin{array}{l}
 \underline{K = 0,11} \\
 \underline{T_D = 0,15} \\
 \underline{T_I = 0,95}
 \end{array}
 \quad
 \begin{array}{l}
 \Rightarrow \text{CONTROLADOR PI:} \\
 K_P = \frac{0,9 \times 0,9}{0,11 \times 0,1} \Rightarrow \underline{K_P = 73,63} \\
 T_I = 3,3 \times T_D \Rightarrow T_I = 3,3 \times 0,1 \Rightarrow \underline{T_I = 0,33s} \\
 D(s) = K_P + \frac{K_P}{s} \quad D(s) = 73,63 + \frac{73,63}{s}
 \end{array}$$

Colocando as duas funções em Cascata



Inserindo D no MatLab

```
>> denG = [1 6 9];
>> G = tf(numG,denG)

G =

      1
-----
s^2 + 6 s + 9

Continuous-time transfer function.

>> step(G)
fx>> numD = [0 24.3 73.63]
```



```

>> numD = [0 24.3 73.63];
>> denD = [0 0.33 0];
>> D = tf(numD,denD)

D =

    24.3 s + 73.63
    -----
    0.33 s

Continuous-time transfer function.

fx >>

```

Após isso, verificar o sistema completo= SysCL.

Temos

```

Continuous-time transfer function.

>> SysCL = feedback(D*G,1)

SysCL =

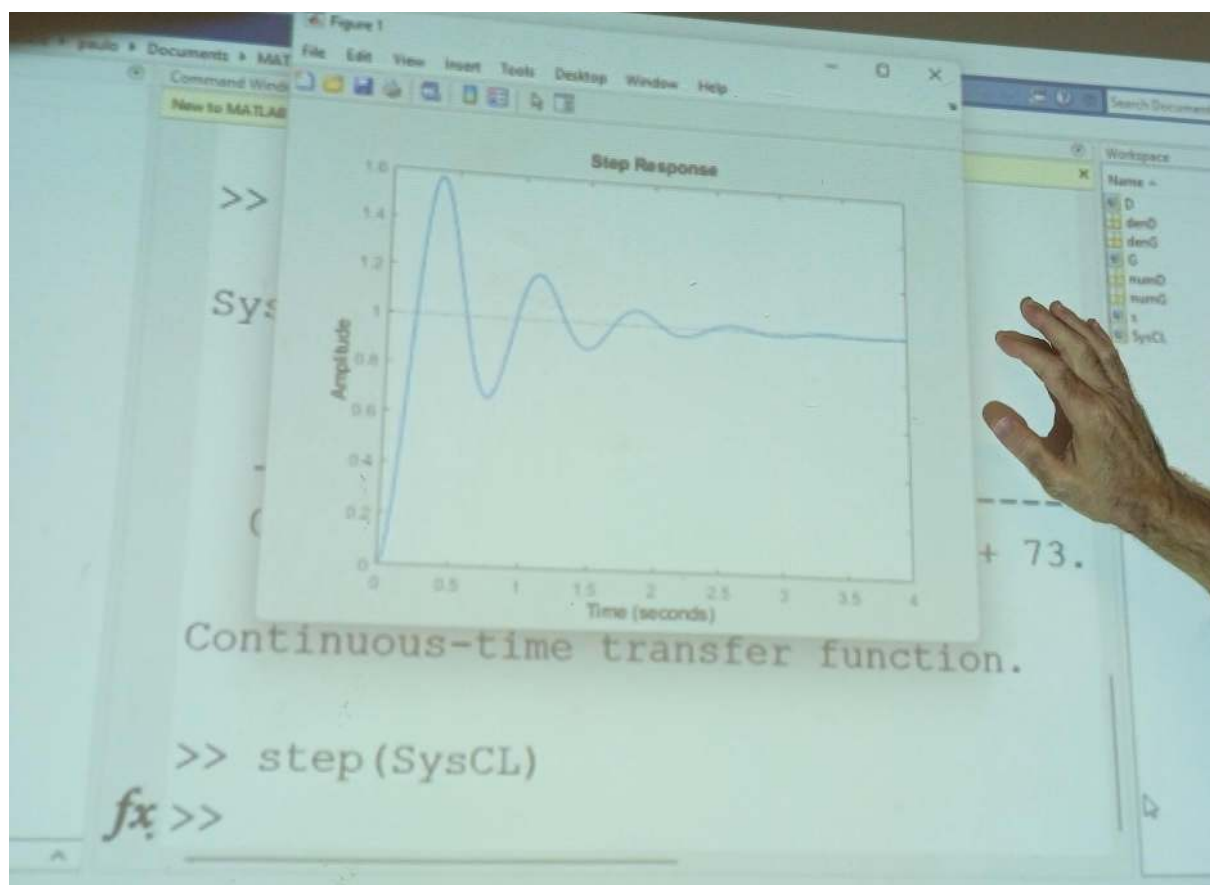
    24.3 s + 73.63
    -----
    0.33 s^3 + 1.98 s^2 + 27.27 s + 73.

Continuous-time transfer function.

fx >>

```

Verificando o gráfico
Usando o Step(SysCL)



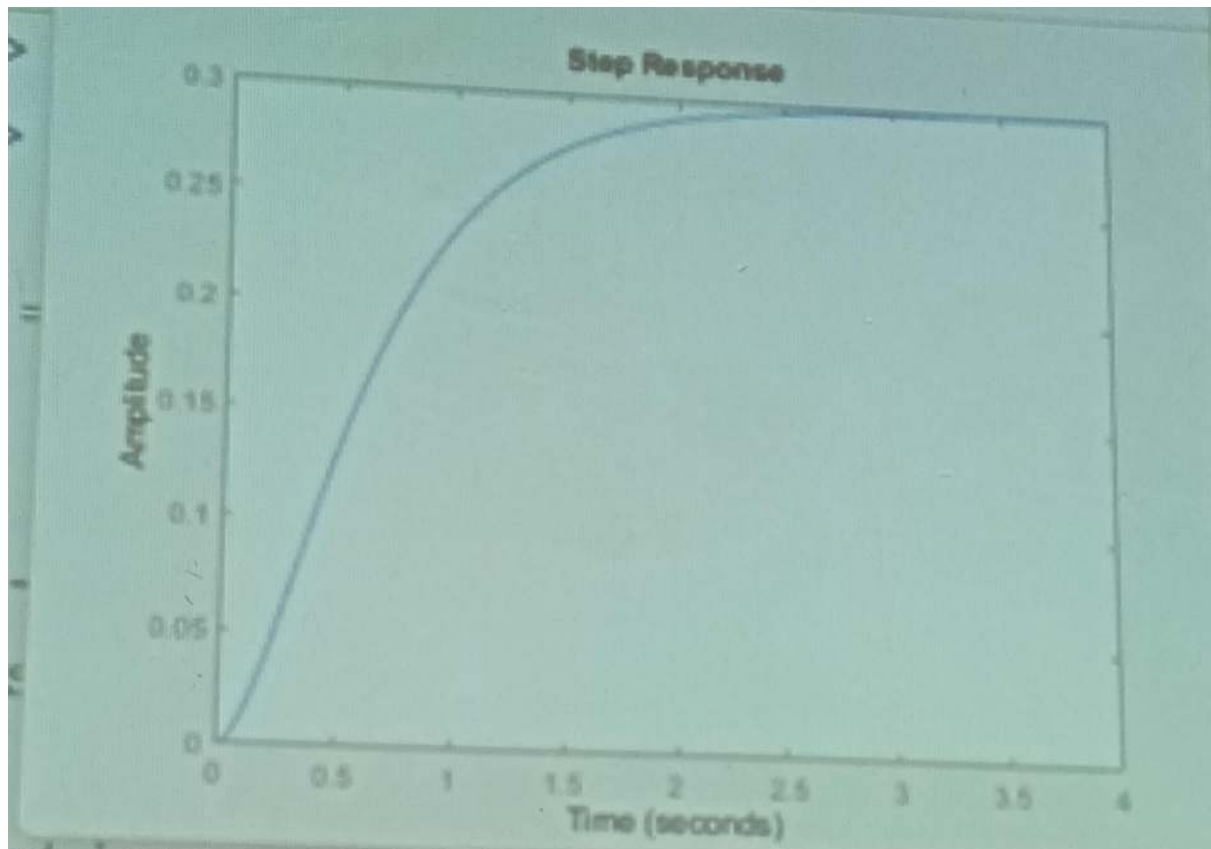
Com isso, conseguimos estabilizar em 1

Exemplo 2

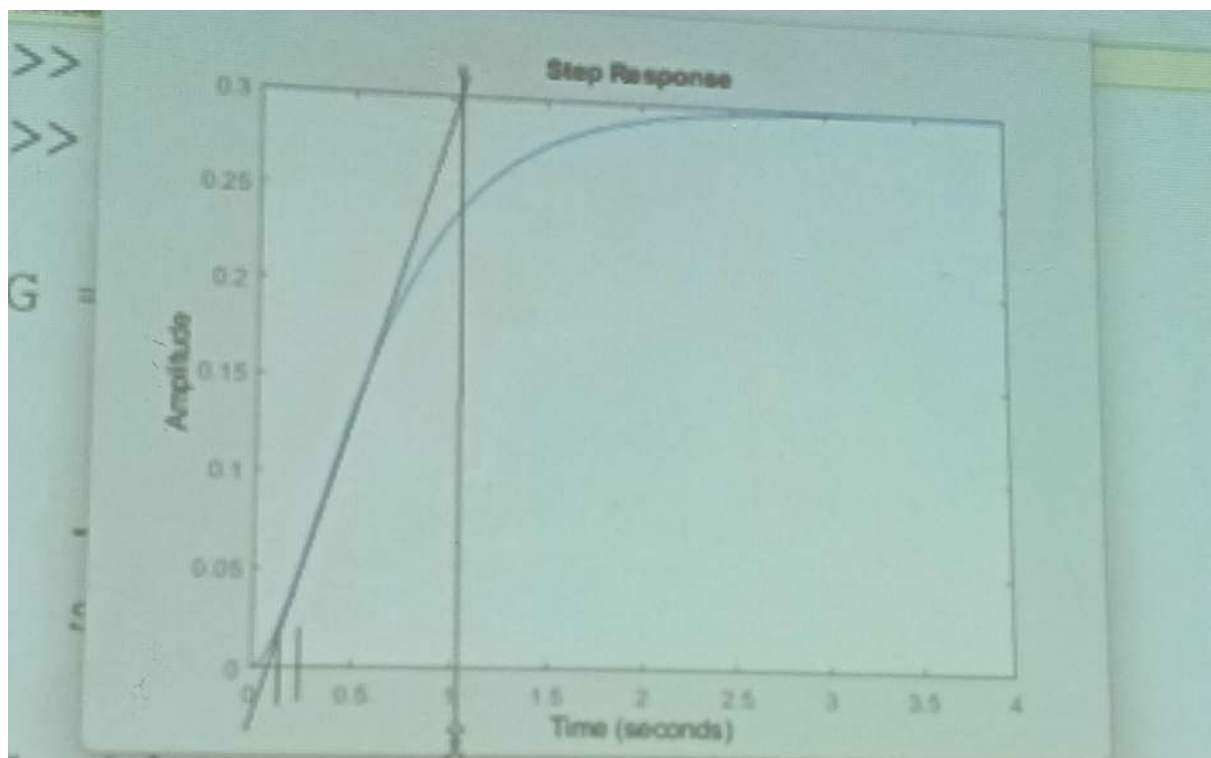
L(s) (chamando do G no Matlab)

```
>> numG = [0 0 3];  
>> denG = [1 7 10];  
>> G = tf(numG,denG)  
  
G =  
  
      3  
-----  
s^2 + 7 s + 10  
  
Continuous-time transfer function.  
fx >>
```

Aplicando Step(G)



Traçando o gráfico



$$K = 0,3$$

$$TD = 0,09 \text{ s}$$

$$T1 = 1,1 - 0,09 \approx 1,0 \text{ s}$$

Com isso, projetarmos o controlador PI:

$$\begin{array}{l} \text{P} \Rightarrow \frac{K_p}{K \cdot T_D} \\ \text{PI} \Rightarrow \frac{0,9 \cdot T_1}{K \cdot T_D} \quad 3,3 \cdot T_D \\ \text{PID} \Rightarrow \frac{1,2 \cdot T_1}{K \cdot T_D} \quad 2 \cdot T_D \quad 0,5 \cdot T_D \end{array}$$

$$K = 0,3 \quad ; \quad T_D = 0,09s \quad ; \quad T_1 = 1s$$

\Rightarrow CONTROLADOR PI:

$$K_p = \frac{0,9 \cdot T_1}{K \cdot T_D} \Rightarrow K_p = \frac{0,9 \cdot 1}{0,3 \cdot 0,09} \Rightarrow \underline{K_p = 33,3}$$
$$T_i = 3,3 \cdot T_D \Rightarrow \underline{T_i = 0,29s}$$

$$D(s) = K_p + \frac{K_p}{T_i \cdot s}$$
$$D(s) = 33,3 + \frac{33,3}{0,29 \cdot s}$$
$$D(s) = \frac{33,3 \times 0,29 \cdot s + 33,3}{0,29 \cdot s}$$
$$D(s) = \frac{9,65 \cdot s + 33,3}{0,29 \cdot s}$$

Inserindo D no MatLab

```
Command Window
New to MATLAB? See resources for Getting Started.
>> numD = [0 9.65 33.3];
>> denD = [0 0.29 0];
>> D = tf(numD,denD)

D =

    9.65 s + 33.3
    -----
    0.29 s

Continuous-time transfer function.
fx>> |
```

Name	Value
D	1st of
denD	[0.2900, 0]
denG	[1, 10]
G	1st of
numD	[0 9.6500 33.3000]
numG	[0.5, 10]
s	1st of
sysCL	1st of

Colocando em cascata G e D

```
Command Window
New to MATLAB? See resources for Getting Started.
Continuous-time transfer function.
>> SysCL = feedback(D*G,1)

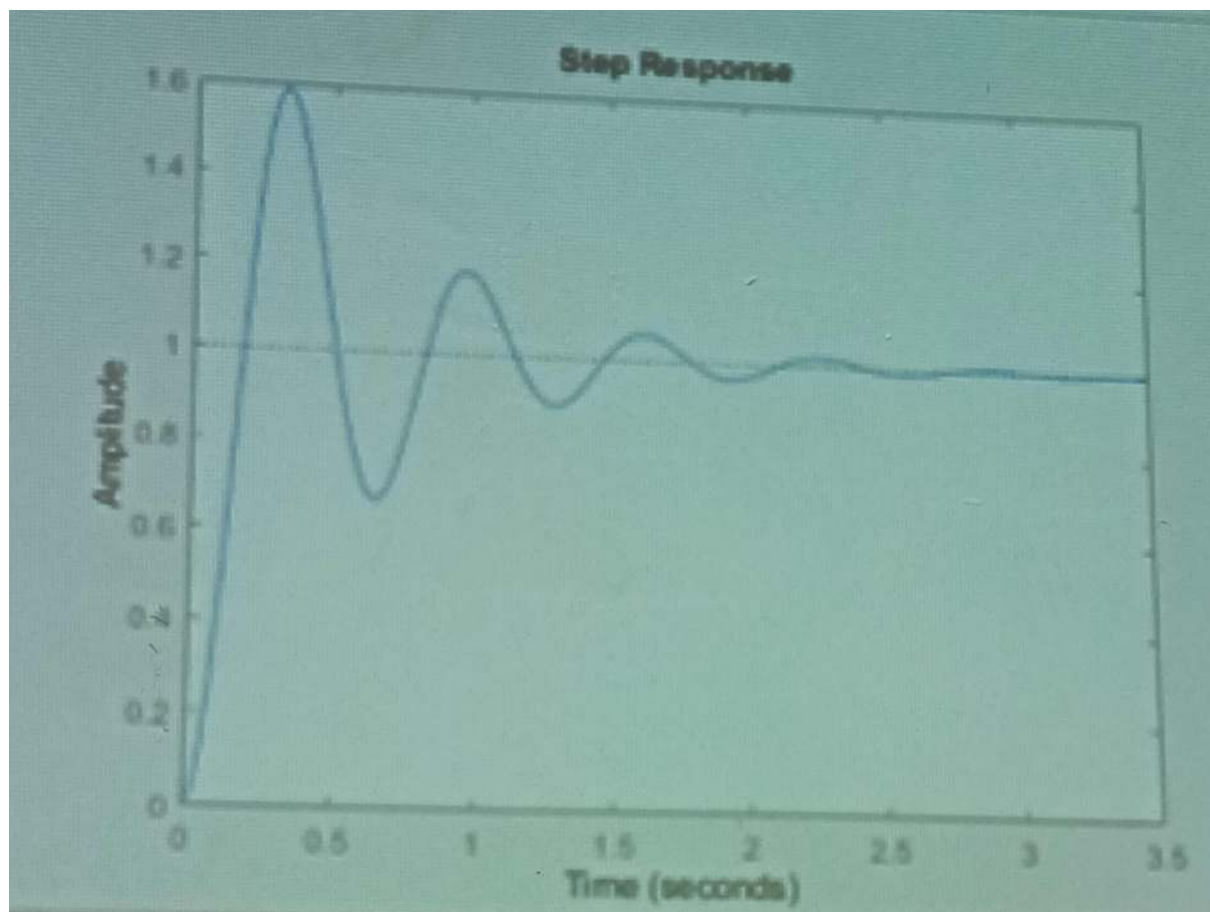
SysCL =

    28.95 s + 99.9
    -----
    0.29 s^3 + 2.03 s^2 + 31.85 s + 99.

Continuous-time transfer function.
fx>>
```

Name	Value
D	1st of
denD	[0.2900, 0]
denG	[1, 10]
G	1st of
numD	[0 9.6500 33.3000]
numG	[0.5, 10]
s	1st of
sysCL	1st of

Aplicando Step(SysCL)



Quinta-feira:

Aplicando os conhecimentos da aula do dia 18/10 e Projetar os circuitos PI como na aula de hoje