Aluno: Epitácio Pessoa de Brito Neto

Matrícula: 11506856

Professora: Ana Wyse

Prova 3 - Sistemas e Controle de Automação

1) Determine o esboço do lugar das raízes e explique o significado do traçado obtido:

a) G (s) = (s + 1) / [(s + 2) (s + 3) (s + 4)]
b) G (s) = [(s + 5) (s + 6)] / [(s + 2) (s + 3) (s + 4)]

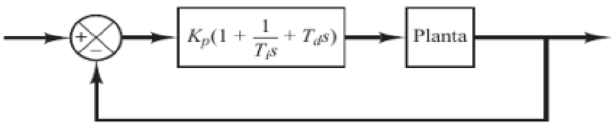
2) Construa o diagrama de Bode e explique o significado do traçado obtido:

a) G (s) = 200s / [(s+2)(s+10)]
b) H(jw) =10 / [(1+jw)(10+jw)]
c) G (s) = 5 / $(s+5)^2$

Obs: Para as questões 1 e 2, pelo menos um dos itens precisa ser esboçado manualmente, de acordo com as técnicas de traçado estudadas, o outro pode ser traçado utilizando um software (nesse caso faça um print onde apareçam os comandos utilizados).

3) A tensão de saída de um amplificador é 3,8V quando a tensão de entrada é 20mV. Qual é o ganho do amplificador adimensional (Vo/Vi) em decibéis?

4) Considere o sistema abaixo, onde o primeiro bloco é um controlador PID dado por Gc(s) e o segundo (planta) dado por G(s) = 0,5 / (1+2s)(1+s)(1+0,5s). Projete um controlador PID utilizando um dos métodos de Ziegler-Nichols.



5) Mostre que não é possível projetar um controlador PID utilizando os métodos de Ziegler-Nichols para uma planta dada por G(s) = (s+2)(s+3) / s(s+1)(s+5).

6) Seja um sistema de ordem n dado por: x'(t)= Ax + Bu. Esse sistema é dito controlável se a matriz de controlabilidade Mc=[B AB A^2B …A^{n-1}B] tem posto n. Considerando A=[-2 0; -1 1] e B=$[2\ 1]^T$ verifique se o sistema é controlável.

7) Seja um sistema de ordem n dado por: x'(t)= Ax + Bu, com saída y(t)= Cx+D. Esse sistema é dito observável se a matriz de observabilidade Mo=$[C\ CA\ CA^2\ \dots\ CA^{n-1}]^T$ tem posto n. Considerando A=[-2 0; -1 1], B=$[0\ 0]^T$, C=$[1\ 0]^T$ e D=[0], verifique se o sistema é observável.

Bibliotecas

In [1]:
```
import matplotlib.pyplot as plt
import scipy.signal as sps
import numpy as np
import control.matlab as ml
```

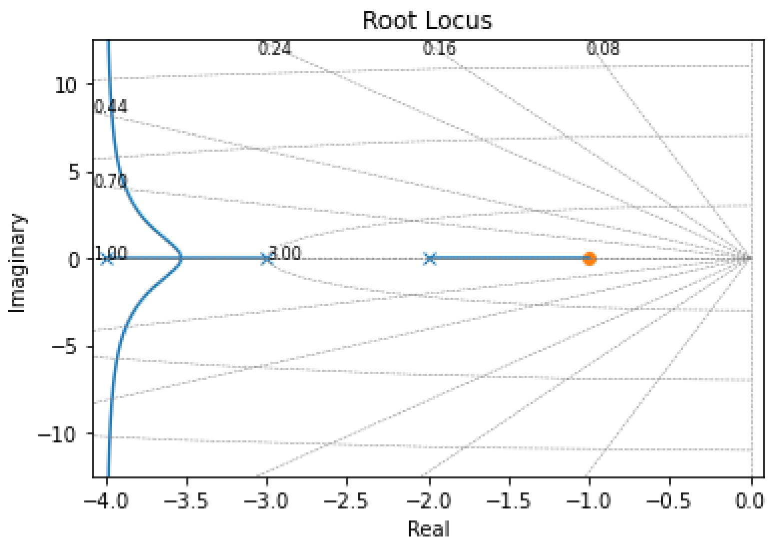
1. a)

In [2]:
```
num = np.array([1,1])
den = np.polymul(np.polymul(np.array([1,2]), np.array([1,3])), np.array([1,4]))

G = ml.tf(num,den)
G
```

Out[2]:

$$\frac{s+1}{s^3+9s^2+26s+24}$$

In [3]:
```
rlist, klist = ml.rlocus(G, grid=True)
plt.show()
```



Como o número das assíntotas é definido pelo $N_{polos} - N_{zeros}$, neste exemplo, possuímos 2 assíntotas. Pelas regras de desenho do LGR, notamos que a função de transferência não possui polos ou zeros complexos, que pode ser observado pelo traçado destes em cima do eixo real e jamais cruzando o eixo imaginário. Ainda com relação à regra de desenho, ao demarcar os pontos com um x, é aplicada a regra em que, o traçado só pertence ao LGR se o número desses x for ímpar, desta forma, temos a situação acima, que entre -3 e -2 não pertence, devido ao número par à direita do ponto (-3,0).

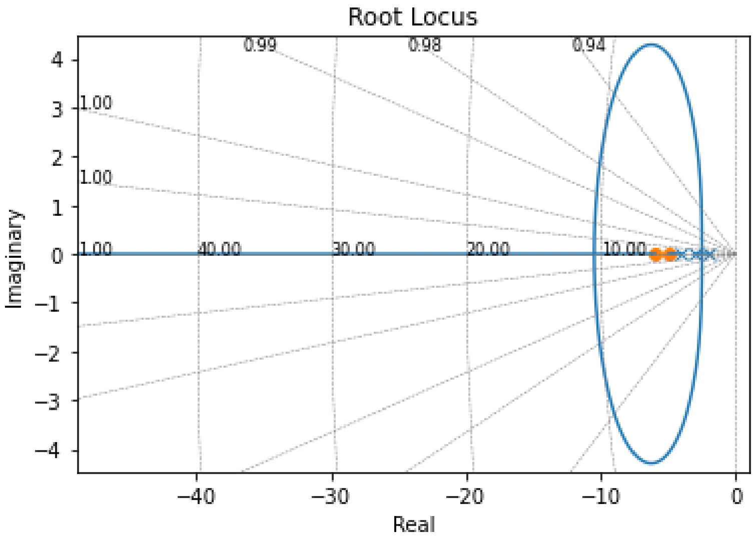
b)

In [4]:
```
num = np.polymul(np.array([1,5]), np.array([1,6]))
den = np.polymul(np.polymul(np.array([1,2]), np.array([1,3])), np.array([1,4]))

G = ml.tf(num,den)
G
```

Out[4]:

$$\frac{s^2+11s+30}{s^3+9s^2+26s+24}$$

In [5]:
```
rlist, klist = ml.rlocus(G, grid=True)
plt.show()
```



Podemos perceber que a trajetória que pertence totalmente ao eixo real se refere a uma das assíntotas e seu devido ângulo, bem como conseguimos enxergar que,o valor aproximado entre [-2,-1] possui trajetórias positivas e negativas pelo eixo imaginário, onde ambos se encontram no ponto (-10,0), ponto de chegada desta trajetória. Pode-se perceber o respeito das regras de traçado do LGR, por exemplo, demarcadas por x e o número de assíntotas, calculada por $N_{polos} - N_{zeros} = 1$ também é respeitada, como foi dito sobre a trajetória, anteriormente.

2. a)

$$G(s) = \frac{200s}{(s+2)(s+10)}$$

$G(j\omega) = 200j\omega * \frac{1}{j\omega+2} * \frac{1}{j\omega+10}$

$G(j\omega) = 200j\omega * \frac{1}{2(\frac{j\omega}{2}+1)} * \frac{1}{10(\frac{j\omega}{10}+1)}$

$G(j\omega) = 200 * \frac{1}{2} * \frac{1}{10} * j\omega * \frac{1}{\frac{j\omega}{2}+1} * \frac{1}{\frac{j\omega}{10}+1}$

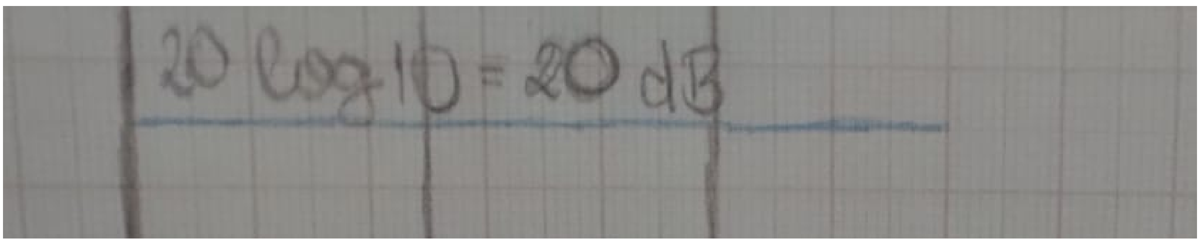
$G(j\omega) = 10 * j\omega * \frac{1}{\frac{j\omega}{2}+1} * \frac{1}{\frac{j\omega}{10}+1}$

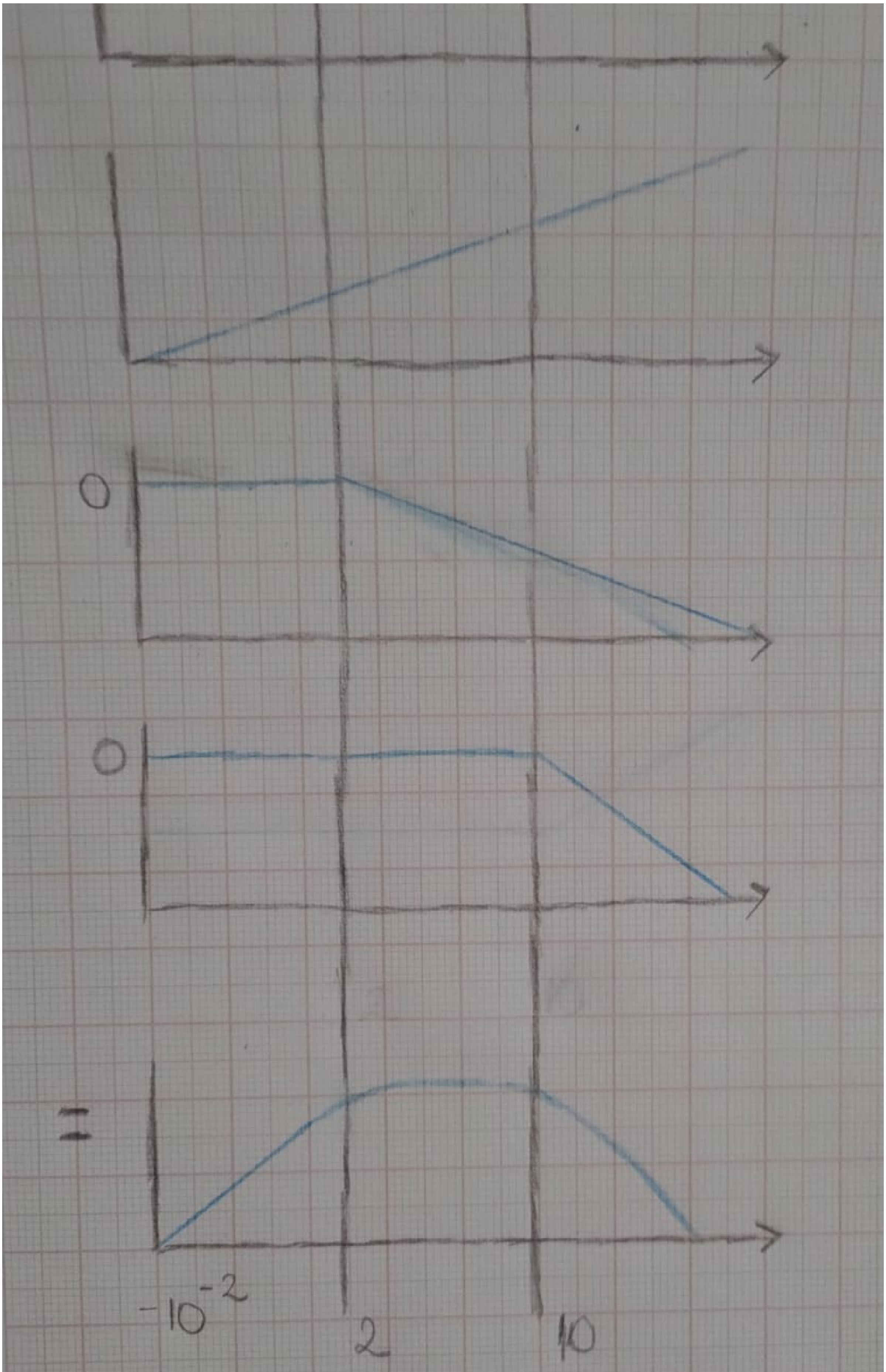
Para cada uma das frações na função $G(j\omega)$, respectivamente, teremos diagramas correspondentes no esboço que virá em seguida, bem como o diagrama de Bode resultante que representa o comportamento total. Antes disso, dada a equação seguinte:

$$W_c = \frac{1}{T}$$

Temos:

$W_{C1} = 2$ e $W_{C2} = 10$





$$20 \log 10^{-2} = -40 \text{ dB}$$

Como os dois últimos gráficos (com excessão do gráfico resultante) estão atuando em zero até o ponto 2, o único comportamento que influenciará o gráfico resultante será o segundo gráfico, subindo, até o 2, 20db/década. Após isto, na frequência de corte, entre [2,10], o segundo e terceiro gráfico atuam neste intervalo, um crescendo e outro descrescendo, portanto, existe esse comportamento estacionário, já que ambos estão se anulando com o valor de 20db/década. Por fim, quando saímos da segunda frequência de corte, temos as mesmas influências atuando na situação anterior, no entanto, devemos considerar a descida de 20db/década do quarto gráfico. Já que o segundo e terceiro gráfico se anulam, sobra apenas a influência do quarto gráfico com uma descida de 20db/década, gerando então, a representação do gráfico resultante.

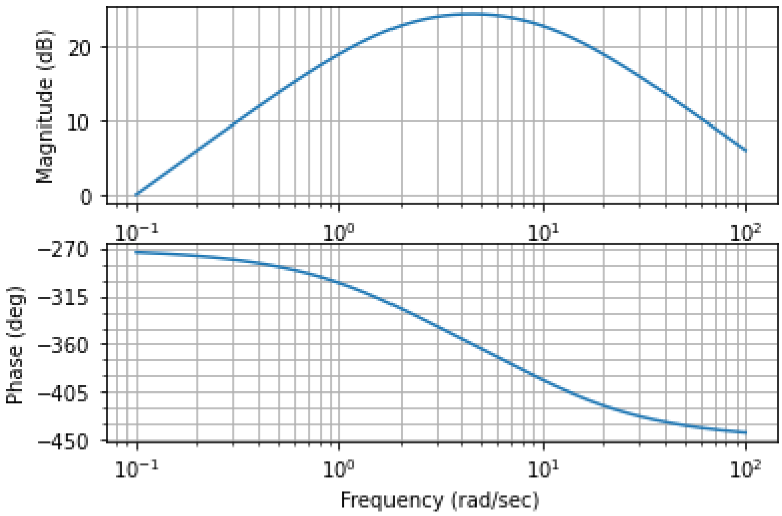
Solução por software

In [6]:
```
num = np.array([200,0])
den = np.polymul(np.array([1,2]), np.array([1,10]))

G = ml.tf(num,den)
G
```

Out[6]:

$$\frac{200s}{s^2 + 12s + 20}$$

In [7]:
```
mag, phase, w = ml.bode(G)
plt.show()
```



b)

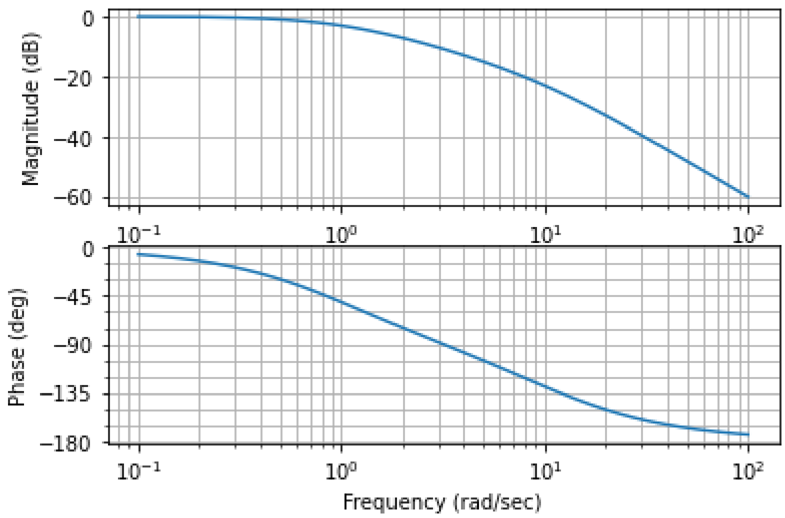
In [8]:
```
num1 = 10
den1 = np.polymul(np.array([1,1]), np.array([1,10]))

G1 = ml.tf(num1,den1)
G1
```

Out[8]:

$$\frac{10}{s^2 + 11s + 10}$$

In [9]:
```
mag1, phase1, w1 = ml.bode(G1)
```

```
plt.show()
```



Para este caso, temos $W_{C1} = 1$ e $W_{C2} = 10$, no entanto, a função de transferência não possui influência do $j\omega$ no numerador, portanto, temos apenas uma constante = 1. Por isto, o início do gráfico se mantém em zero até a primeira frequência de corte, onde sofre a primeira influência, decaindo. Ao chegar na segunda frequência de corte, portanto, temos influência de dois comportamentos que com mesmo sentido, portanto, o processo de queda da magnitude é intensificada, no fim.

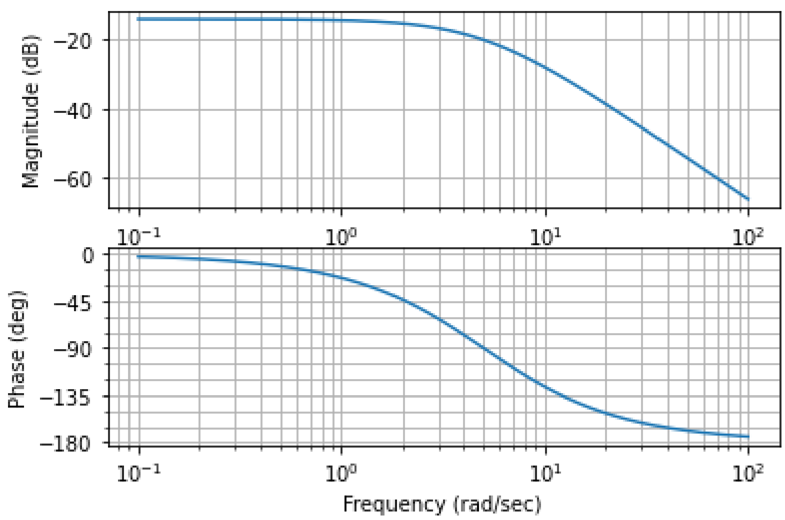
c)

In [10]:
```
num2 = 5
den2 = np.polymul(np.array([1,5]), np.array([1,5]))

G2 = ml.tf(num2,den2)
G2
```

Out[10]:

$$\frac{5}{s^2 + 10s + 25}$$

In [11]:
```
mag2, phase2, w2 = ml.bode(G2)
plt.show()
```



Para este, temos $W_{C1} = W_{C2} = 5$, portanto, temos uma única frequência de corte, bem como não possui um termo dependente de $j\omega$ no denominador. Portanto, até chegar na primeira (e única) frequência de corte, o diagrama possui o comportamento estável e sem alterações de magnitude até que, ao chegar na frequência de corte, possui a influência de dois comportamentos de mesma intensidade e sentido, portanto, decai de forma mais intensa logo em seguida.

3.

$K = \frac{3.8}{20*10^{-3}} = 190 \longrightarrow K = 20log190 =$ **45.6dB**

4.

In [12]:

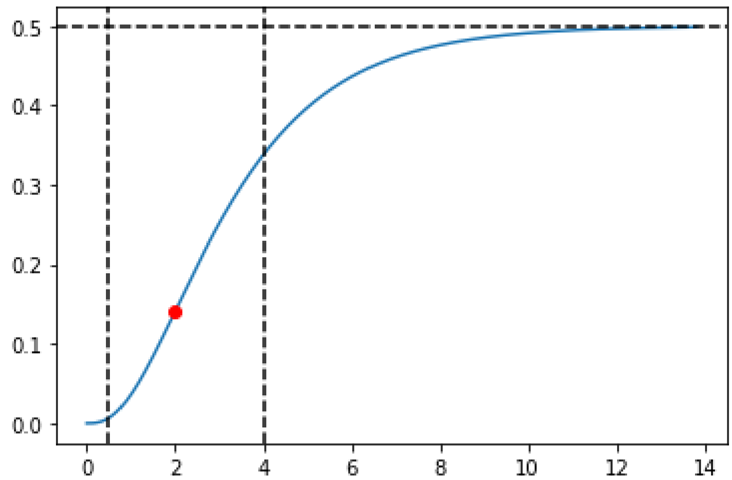
```
num3 = 0.5
den3 = np.polymul(np.polymul(np.array([1,2]), np.array([1,1])), np.array([1,0.5]))
G3 = ml.tf(num3,den3)
G3
```

Out[12]:

$$\frac{0.5}{s^3 + 3.5s^2 + 3.5s + 1}$$

In [13]:

```
yout, T = ml.step(G3)

# Encontrando o ponto de inflexão
dy = np.diff(yout) # Primera derivada
idx_max_dy = np.argmax(dy)

plt.plot(T, yout)
plt.plot(T[idx_max_dy], yout[idx_max_dy], 'or', label='estimated inflection point')
plt.axvline(x=0.5, color='k', linestyle='--')
plt.axvline(x=4, color='k', linestyle='--')
plt.axhline(y=0.5, color='k', linestyle='--')
plt.show()
```



$$Tabela$$

	K_p	T_i	T_d
PID	$1.2\frac{T}{L}$	$\frac{0.5}{L}$	$0.5L$

$$Resultados$$

Sejam $T = 3.5$, $K = 0.5$ e $L = 0.5$, temos:

	K_p	T_i	T_d
PID	8.4	1	0.25

$$PID = K_p(1 + \frac{1}{T_i s} + T_d s) = K_p + \frac{K_i}{s} + K_d s$$

Substituindo os valores da tabela na equação, temos $K_p = 8.4$, $K_i = 8.4$ e $K_d = 2.1$, portanto:

$$PID = 8.4 + \frac{8.4}{s} + 2.1s$$

$$PID = \frac{8.4s^2 + 8.4s + 2.1}{s}$$

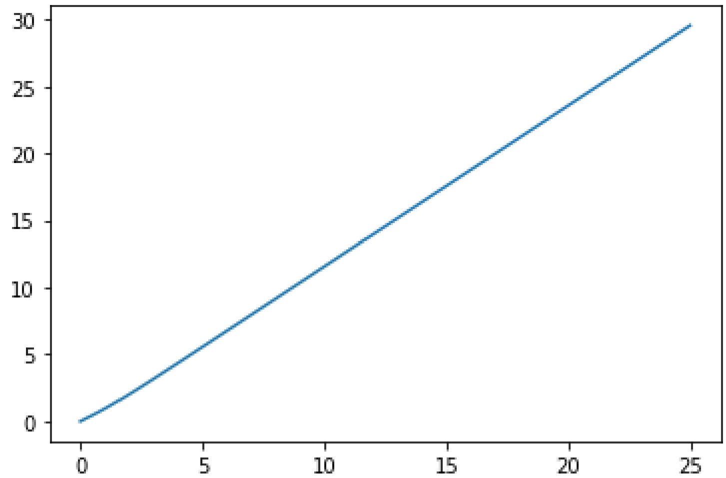
5.

In [14]:
```
num4 = np.polymul(np.array([1,2]), np.array([1,3]))
den4 = np.polymul(np.polymul(np.array([1,1]), np.array([1,0])), np.array([1,5]))
G4 = ml.tf(num4,den4)
G4
```

Out[14]:

$$\frac{s^2 + 5s + 6}{s^3 + 6s^2 + 5s}$$

In [15]:
```
yout1, T1 = ml.step(G4)
plt.plot(T1, yout1)
plt.show()
```



Para aplicar o método de Ziegler-Nichols, após o degrau, a curva resultante deve ter o comportamento de uma sigmoide, ou seja, uma curva que se assemelha à letra S. Isso se deve a uma característica de equações de comportamento sigmoidal que é a representação da função de transferência em um modelo de primeira ordem com atraso, necessária para aplicação do método. Como podemos ver, o comportamento da curva resultante da nossa função não corresponde com as características necessárias, portanto, não podemos utilizar o método Ziegler-Nichols, neste caso.

6.

$$AB = \begin{bmatrix} -2 & 0 \\ -1 & 1 \end{bmatrix} * \begin{bmatrix} 2 \\ 1 \end{bmatrix} = \begin{bmatrix} -4 \\ -1 \end{bmatrix}$$

$$M_c = [\,B \quad AB\,] = \begin{bmatrix} 2 & -4 \\ 1 & -1 \end{bmatrix}$$

$Posto(M_c) = 2 \longrightarrow$ **É controlável.**

7.

Supus que $C = [\,1 \quad 0\,]^T$ fosse, no caso, a não transposta de forma que fosse possível avaliar a observabilidade já que, com a transposta, o cálculo de CA não seria possível dada as dimensões das matrizes durante para a operação de multiplicação.

$$CA = [\,1 \quad 0\,] * \begin{bmatrix} -2 & 0 \\ -1 & 1 \end{bmatrix} = [\,-2 \quad 0\,]$$

$$M_o = \begin{bmatrix} C \\ CA \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ -2 & 0 \end{bmatrix}$$

$Posto(M_o) = 1 \longrightarrow$ **Não é observável.**