Universidade Federal de Juiz de Fora

3º Trabalho de Controle

Aluno: Dylan Soares de Vasconcelos Moreno.

07 de julho de 2024

Sumário

Questão 1:

Letra a)

A partir das equações de estado e os valores de L, C e R dados na questão, foi possível obter uma representação no espaço de estados do sistema:

$$\mathrm{L}\frac{\mathrm{di_L}}{\mathrm{dt}} = -(1-\mathrm{d})\mathrm{v_c} + \mathrm{E_s}$$

$$\mathrm{C}\frac{\mathrm{dv_c}}{\mathrm{dt}} = -(1-\mathrm{d})\mathrm{i_L} - \frac{\mathrm{v_c}}{\mathrm{R}}$$

Sendo: L = 6 mH, C = 1mF, R = 100 Ω, PWM de 50%, ou seja, duty cycle de 0,5.

Foi considerado $E_s = 1V$, e selecionou as seguintes variáveis de estados $x_1 = i_L$ e $x_2 = v_c$, substituindo as variáveis na equação chegou em:

$$\mathrm{L}\dot{\mathrm{x}}_1 = -(1-\mathrm{d})\mathrm{x}_2 + \mathrm{E_s}$$

$$\dot{\mathrm{x}}_1 = \frac{-(1-\mathrm{d})}{\mathrm{L}}\mathrm{x}_2 + \frac{\mathrm{E_s}}{\mathrm{L}}$$

$$\dot{\mathrm{x}}_2 = -(1-\mathrm{d})\mathrm{x}_1 - \frac{\mathrm{x}_2}{\mathrm{R}}$$

$$\dot{\mathrm{x}}_2 = \frac{-(1-\mathrm{d})}{\mathrm{C}}\mathrm{x}_1 - \frac{\mathrm{x}_2}{\mathrm{RC}}$$

Substituindo os valores:

$$\dot{\mathrm{x}}_1 = \frac{-250}{3}\mathrm{x}_2 + \frac{500}{3}$$

$$\dot{\mathrm{x}}_2 = 500\,\mathrm{x}_1 - 10\mathrm{x}_2$$

Assim chegamos na seguinte representação matricial do espaço de estados:

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} = \begin{bmatrix} 0 & -\frac{250}{3} \\ 500 & -10 \end{bmatrix} * \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} \frac{500}{3} \\ 0 \end{bmatrix} u$$

$$y = \begin{bmatrix} 0 & 1 \end{bmatrix} * \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$$

Letra b)

Em seguida, foi determinado o conjunto de ganhos de realimentação de estado para obter 20% de ultrapassagem e um tempo de acomodação de 0,5 segundo.

Para a obtenção dos ganhos de realimentação é possível ser realizado por três métodos, sendo eles:

Método 1: Matriz de transformação:

Esse método consiste em 5 etapas, as quais são subdivididas em:

Etapa 1: Verificar a condição de controlabilidade do sistema, a partir da matriz de controlabilidade calcula-se o posto dessa matriz se for igual ao número de variáveis de estados é controlável e pode seguir para a próxima etapa.

Etapa 2: A partir da equação característica de A (matriz de estados), ou seja, $|sI - A| = s^n + \propto_{n-1} s^{n-1} + \ldots + \propto_1 s + \propto_0$

Determina a matriz W

$$W = \begin{bmatrix} \propto_1 & \propto_2 & \cdots & \propto_{n-1} & 1 \\ \propto_2 & \propto_3 & \cdots & 1 & 0 \\ \vdots & \vdots & \cdots & \vdots & \vdots \\ 1 & 0 & \cdots & 0 & 0 \end{bmatrix}$$

Etapa 3: Determinar a matriz de transformação T, essa matriz transforma a matriz da equação do sistema na forma canônica controlável, caso esteja na forma canônica controlável, T = Identidade

T= U*W, sendo U a matriz de controlabilidade

$$U = [\mathbf{B} \quad \mathbf{AB} \quad \ldots \quad \mathbf{A^{n-1}B}]$$

Etapa 4: A partir da posição dos autovalores desejados, determina-se os valores de $a_{n-1}, \ldots, a_1, a_0$, a partir do polinômio característico desejado.

$$(s-\mu_1)(s-\mu_2)\cdots(s-\mu_n) = s^n + a_{n-1}s^{n-1} + \ldots + a_1 s + a_0$$

Etapa 5: Determinar matriz de ganho K de realimentação de estado de acordo com a seguinte equação:

$$K = [a_0 - \alpha_0 \;\; a_1 - \alpha_1 \;\; \ldots \;\; a_{n-2} - \alpha_{n-2} \;\; a_{n-1} - \alpha_{n-1}]T^{-1}$$

Método 2: Substituição direta:

Utilizado em sistemas de baixa ordem ($n \leq 3$), sendo K no máximo até k_3 ,ou seja, K = $[k_1, k_2, k_3]$.

Por seguinte é substituído a matriz K no polinômio característico $|sI - A + BK|$ e igualado ao polinômio característico desejado $(s-\mu_1)(s-\mu_2)\cdots(s-\mu_n)$.

$$|sI - A + BK| = (s-\mu_1)(s-\mu_2)\cdots(s-\mu_n)$$

Com isso é obtido os valores dos elementos da matriz de ganho K a partir da igualdade acima.

Método 3: Fórmula de Ackermann:

$$K = [0\;0\;\ldots\;0\;1]U^{-1}\emptyset(A)$$

onde $\emptyset(A) = A^n + +a_{n-1}A^{n-1} + \ldots + a_1 A + a_0 I$, vem a partir do polinômio característico desejado: $\lambda^n + a_{n-1}\lambda^{n-1} + \ldots + a_1\lambda + a_0$

Na questão em específico foi escolhido o segundo método:

Para o cálculo do polinômio característico desejado foi levado em consideração os dados apresentados e utilizou as seguintes fórmulas para encontrar os polos:

$$\zeta = \frac{-\ln\left(\frac{\%UP}{100}\right)}{\sqrt{\pi^2 + ln^2\left(\frac{\%UP}{100}\right)}} \quad , \quad T_{s(2\%)} = \frac{4}{\zeta\,\omega_n} \quad , \quad s^2 + 2\zeta\omega_n s + \omega_n{}^2$$

Em primeiro lugar, calcula-se o zeta:

$$\zeta = \frac{-\ln\left(\frac{20}{100}\right)}{\sqrt{\pi^2 + \ln^2\left(\frac{20}{100}\right)}}$$

$$\zeta = 0{,}4559.$$

A seguir o ω_n pelo tempo de acomodação:

$$T_{s(2\%)} = \frac{4}{\zeta\,\omega_n}$$

$$\omega_n = \frac{4}{\zeta T_{s(2\%)}}$$

$$\omega_n = \frac{4}{0{,}4559 * 0{,}5}$$

$$\omega_n = 17{,}5477 \text{ rad/s}$$

Por fim aplica-se na fórmula de obtenção polinômio quadrático.

$$s^2 + 2\zeta\omega_n s + \omega_n{}^2$$

$$s^2 + 2 * 0{,}4559 * 17{,}5477s + 17{,}5477^2$$

Chegamos em um polinômio de aproximadamente de

$$s^2 + 16s + 307{,}9218$$

Utilizando o método 2 da substituição direta:

$$|sI - A + BK| = (s - \mu_1)(s - \mu_2)\cdots(s - \mu_n)$$

$$\begin{bmatrix} s & 0 \\ 0 & s \end{bmatrix} - \begin{bmatrix} 0 & -\frac{250}{3} \\ 500 & -10 \end{bmatrix} + \begin{bmatrix} \frac{500}{3} \\ 0 \end{bmatrix} * \begin{bmatrix} k_1 & k_2 \end{bmatrix}$$

$$\begin{bmatrix} s & \frac{250}{3} \\ -500 & s + 10 \end{bmatrix} + \begin{bmatrix} \frac{500}{3}k_1 & \frac{500}{3}k_2 \\ 0 & 0 \end{bmatrix}$$

$$\det\left(\begin{bmatrix} s + \frac{500}{3}k_1 & \frac{250}{3} + \frac{500}{3}k_2 \\ -500 & s + 10 \end{bmatrix}\right)$$

Chegamos no seguinte polinômio característico:

$$(\mathrm{s} + \frac{500}{3}\mathrm{k}_1)*(\,\mathrm{s} + 10) + 500 * (\frac{250}{3} + \frac{500}{3}\mathrm{k}_2) = \mathrm{s}^2 + 16\mathrm{s} + 307{,}9218$$

$$\mathrm{s}^2 + s\left(\frac{500}{3}\mathrm{k}_1 + 10\right) + (\frac{5000}{3}\mathrm{k}_1 + \frac{125000}{3} + \frac{250000}{3}\mathrm{k}_2) = \mathrm{s}^2 + 16\mathrm{s} + 307{,}9218$$

$$\left(\frac{500}{3}\mathrm{k}_1 + 10\right) = 16$$

$$\frac{500}{3}\mathrm{k}_1 = 6$$

$$\mathrm{k}_1 = 0{,}036.$$

$$(\frac{5000}{3}\mathrm{k}_1 + \frac{125000}{3} + \frac{250000}{3}\mathrm{k}_2) = 307{,}9218$$

$$(\frac{5000}{3} * 0{,}036 + \frac{125000}{3} + \frac{250000}{3}\mathrm{k}_2) = 307{,}9218$$

$$\frac{250000}{3}\mathrm{k}_2 = -41418{,}7449$$

$$\mathrm{k}_2 = -0{,}4970.$$

Simulando a resposta ao degrau desse sistema temos:

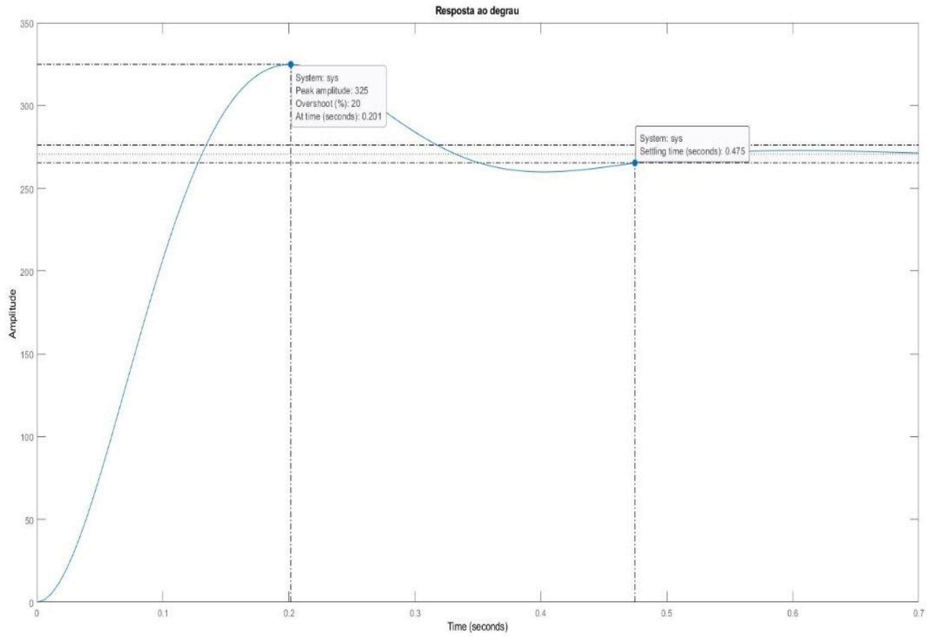

Figura 1: Resposta ao degrau com realimentação de estados.

A fim de comparação com a pergunta seguinte foi calculado o erro desse sistema, o qual foi obtido:

Erro = -269,6904.

Letra c)

Em sequência foi feito o controle integral, visando minimizar o erro da resposta ao degrau com ganhos de realimentação. Com isso é necessário recalcular os ganhos K e acrescentar um ganho Ke, como a ordem do sistema aumenta, precisamos de um polo que não atrapalhe os dominantes já escolhidos anteriormente. A partir da análise da função de transferência da planta notamos que não apresenta zeros, então o terceiro polo é escolhido com uma distância de 10 vezes os polos dominantes, e obtivemos um polo em -80.

Abaixo é apresentado a nova representação no espaço de estados, a equação característica do novo sistema e os valores obtidos nos ganhos, esses usados para plotagem do gráfico com controle integral.

$$\begin{bmatrix} \dot{\boldsymbol{x}} \\ \dot{x}_N \end{bmatrix} = \begin{bmatrix} A - BK & BK_e \\ -C & \bar{0} \end{bmatrix} \begin{bmatrix} \boldsymbol{x} \\ x_N \end{bmatrix} + \begin{bmatrix} \bar{0} \\ 1 \end{bmatrix} r$$

$$y = \begin{bmatrix} C & 0 \end{bmatrix} \begin{bmatrix} \boldsymbol{x} \\ x_N \end{bmatrix}$$

Aplicando o determinante da matriz A, chegamos em:

$$s^3 + s^2\left(10 + 500 * \frac{k_1}{3}\right) + s(5000 * \frac{k_1}{3} + \frac{125000}{3} s + 250000 * \frac{k_2}{3}) + 500 * K_e$$

$$k_1 = 0{,}516\ , k_2 = -0{,}49126 \text{ e } k_e = 0{,}29553744$$

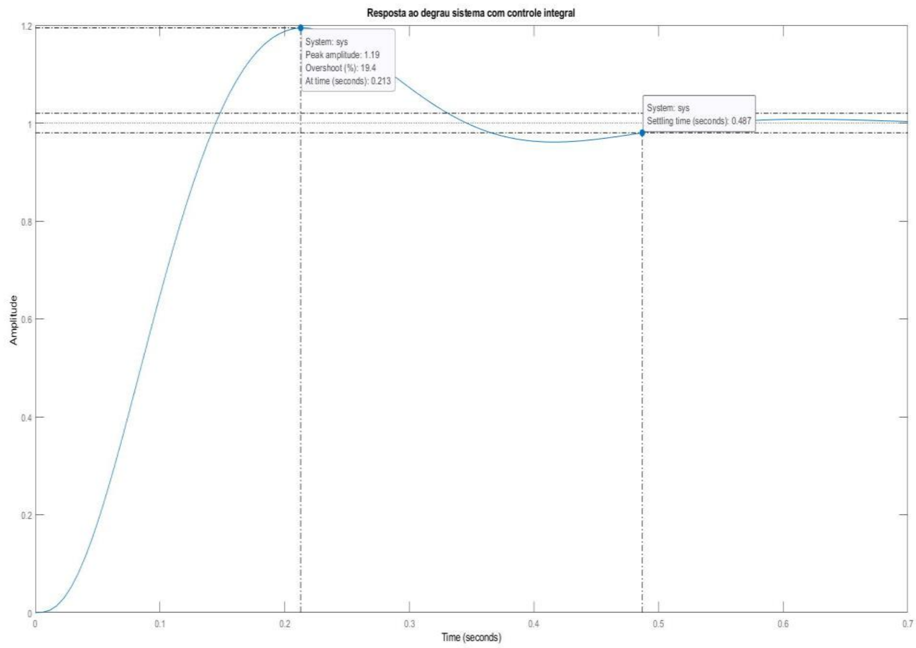

Figura 2: Resposta ao degrau do sistema com realimentação de estados e controle Integral

O Erro com o controle integral é de:

Erro = 0

Com o propósito de comparação foi feito o diagrama de simulação no Simulink:

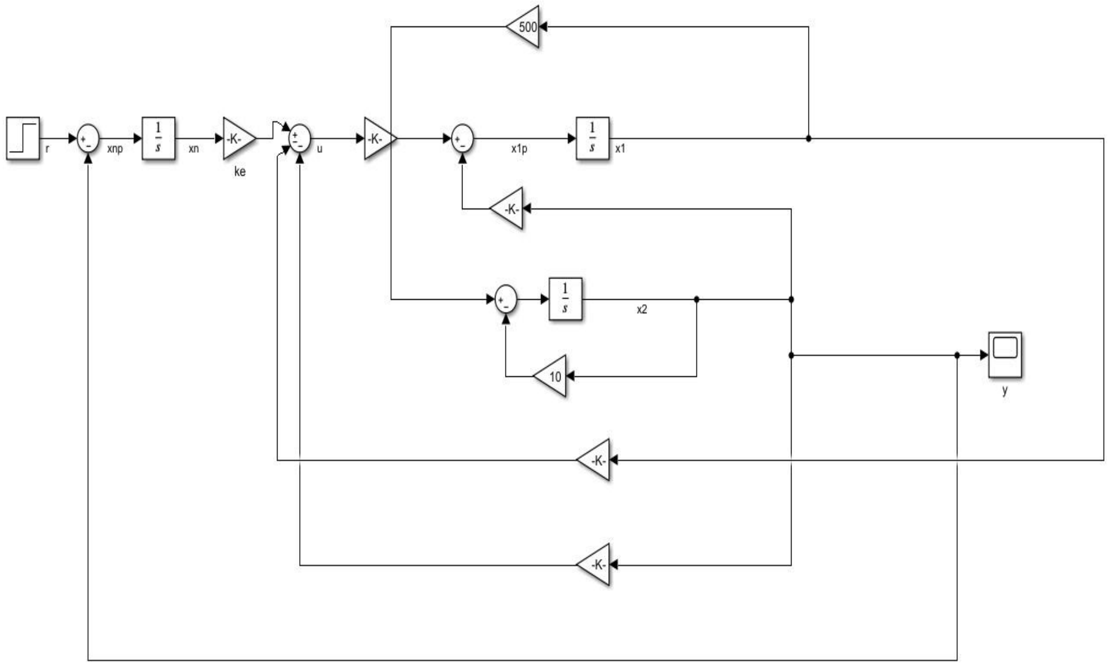

Figura 3: Diagrama de simulação da realimentação dos ganhos com controle integral

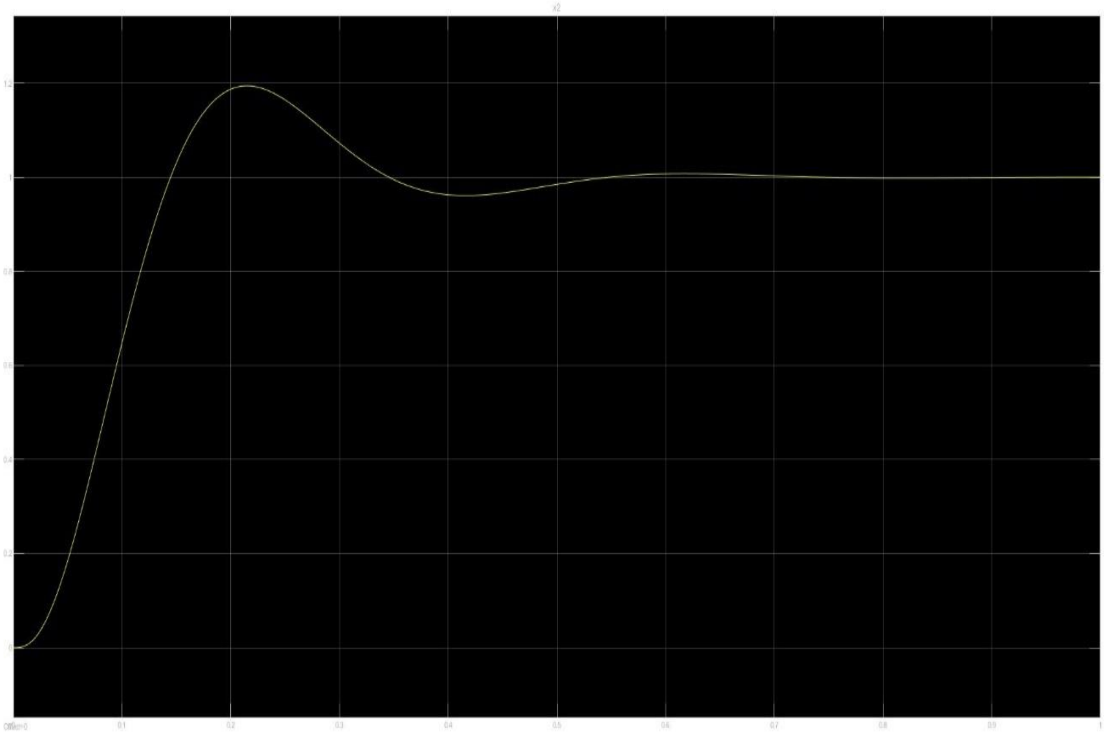

Figura 4: Resposta ao degrau com Sistema com Controle Integral no Simulink

Código do MATLAB questão 1:

```
clc;
clear;
close all;

syms k1 k2 Ke s


%% Planta
A= [0 -250/3;500 -10];
B = [500/3;0];
C = [0 1];
D =0;
[num,den] = ss2tf(A,B,C,D);
G = tf(num,den);

%% letra b)
%polos desejados
UP =20;
Ts =0.5;
I =eye(2);
zeta = -log(UP/100)/sqrt(pi^2+(log(UP/100)^2));
wn= 4/(zeta*Ts);
den_d =[1 2*zeta*wn wn^2];
polos_desejados = roots(den_d);
p1 =polos_desejados(1,1);
p2 =polos_desejados(2,1);

K = acker(A,B,polos_desejados);

figure;
step(A-B*K,B,C,D);
title('Resposta ao degrau');


% Cálculo do erro em regime permanente do sistema controlado por
% realimentação de estados (RE) para entrada degrau unitário
E_RE=(1/s)*(1-C*((s*I-A+B*K)^(-1))*B);
erro_RE=subs(s*E_RE,s,0);


%% letra c
I=eye(3);
% como na função de transferência da planta não possui zeros escolhemos o
% mais distante possivel dos polos dominantes, aqui foi escolhido um polo
% 10x mais distante
p3 =10*real(p1);
polos_desejados_integral =[p1,p2,p3];
polinomio_desejado = expand(((s-p1)*(s-p2)*(s-p3)));
clear k1 k2;
```

```
syms k1 k2;
K_int=[k1,k2];
Aint =[A-B*K_int B*Ke;-C 0];
det_int_antigo = det(s*eye(3)-Aint);
Kint=[0.516,-0.49126];
Ke = 0.29553744;
Aint =[A-B*Kint B*Ke;-C 0];
det_int = det(s*eye(3)-Aint);
Cint =[C 0];
Bint=[0;0;1];
Dint=0;
figure();
step(Aint,Bint,Cint,Dint);   % resposta ao degrau do sistema com controle integral
title('Resposta ao degrau sistema com controle integral');
E_REint=(1/s)*(1-(Cint*(inv(s*I-Aint))*Bint));
erro_REint=subs(s*E_REint,s,0);

[num_int,den_int]=ss2tf(Aint,Bint,Cint,Dint);
FT_controle_integral = tf(num_int,den_int);
```

Questão 2:

Letra a)

A partir da planta modelado no espaço de estados abaixo projetou uma matriz de ganho de realimentação de estado completo e uma matriz de ganho do observador de estados para alocar os polos do sistema e malha fechada em:

$$s_{1,2} = -1{,}4 \pm j1{,}4 \, , s_{3,4} = -2 \pm j1$$

e os polos do observador em:

$$s_{1,2} = -18 \pm j5 \, , s_{3,4} = -20$$

Sua matriz é dada como:

$$\dot{\boldsymbol{x}}(\boldsymbol{t}) = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ -2 & -5 & -1 & -13 \end{bmatrix} \boldsymbol{x}(\boldsymbol{t}) + \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \end{bmatrix} u(t)$$

$$\text{y(t)} = \begin{bmatrix} 1 & 0 & 0 & 0 \end{bmatrix} \boldsymbol{x}(\boldsymbol{t}) + [0] u(t)$$

Com a realimentação de estado completo e observador integrados pode ser obtido a seguinte matriz relacionada aos ganhos, variáveis e erros estimados:

$$\begin{bmatrix} \dot{x} \\ \dot{e} \end{bmatrix} = \begin{bmatrix} A - BK & BK \\ 0 & A - LC \end{bmatrix} \begin{bmatrix} x \\ e \end{bmatrix}$$

A equação característica associada é:

$$\Delta(\lambda) = \det(\lambda I - (A - BK) * \det(\lambda I - (A - LC))$$

Ou seja, a equação característica pode ser obtida a partida da multiplicação dos determinantes da matriz de ganhos de realimentação de estado com a determinante da matriz do observador.

É importante salientar que ambas devem ser estáveis, portanto os polos devem ser selecionados ao semiplano esquerdo no plano s.

Diante disso, foi calculado os valores de K e de L a partir da formula de ackermann no MATLAB, assim chegamos em:

$$\text{K} = \begin{bmatrix} 17{,}6 & 24{,}68 & 19{,}12 & -6{,}2 \end{bmatrix}$$

$$L = \begin{bmatrix} 63 \\ 1369 \\ 10495 \\ 1479 \end{bmatrix}$$

Para confirmar que os polos estão alocados no lugar certo foi plotado o mapa de polos e zeros do sistema apenas com realimentação de estados e com o sistema com apenas o observador:

Sistema com realimentação de estados:

$$\dot{x} = (A - BK)\mathbf{x} + Bu$$

$$y = C\mathbf{x} + Du$$

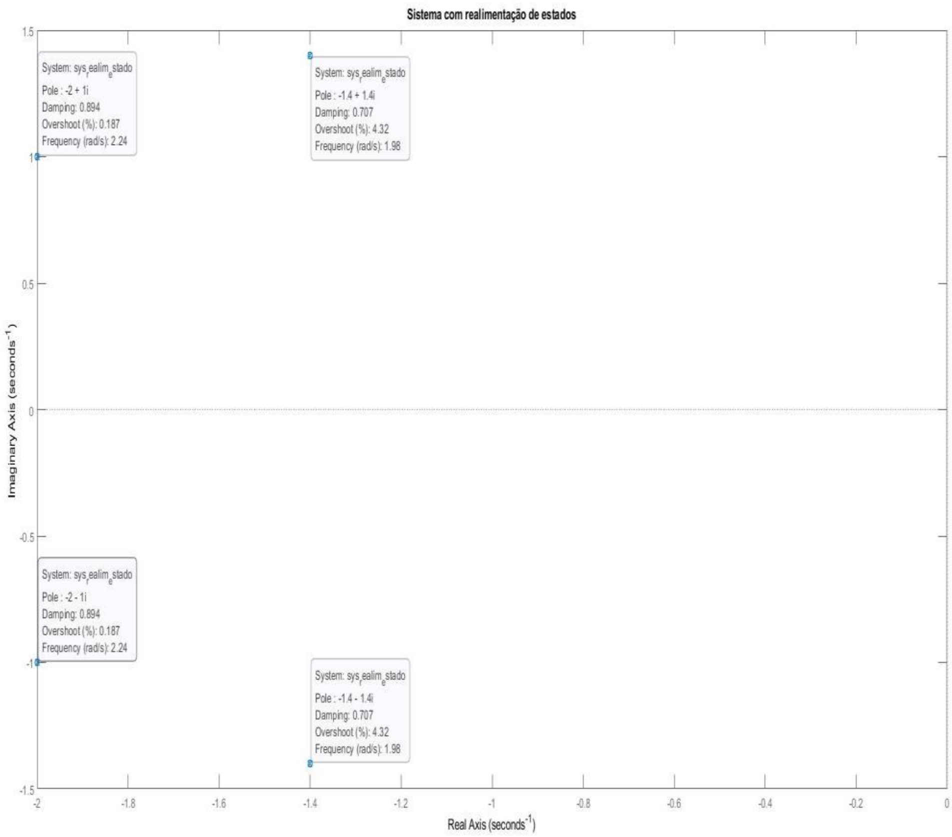

Figura 5: Mapa de polos e zeros para o sistema com realimentação de estados

Sistema com apenas o observador:

$$\dot{x} = (\mathrm{A} - \mathrm{LC})\mathbf{x} + \mathrm{Bu}$$

$$\mathrm{y} = \mathrm{C}\mathbf{x} + \mathrm{Du}$$

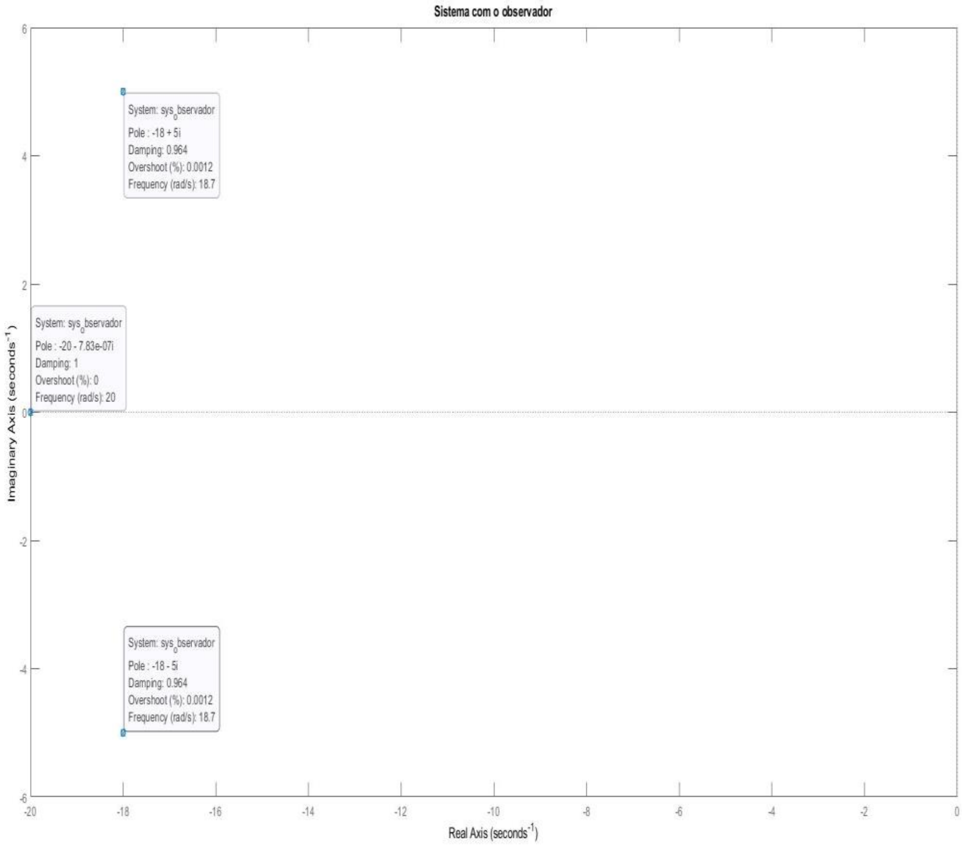

Figura 6: Mapas de polos e zeros do sistema com o observador

Perceba que nesse mapa de polos e zeros, por motivos de aproximações, o MATLAB acaba considerando um número ínfimo nos imaginários para os polos em -20, o que pode ser considerado como zero. Além disso, nota-se que não é possível verificar os dois polos em -20 isso ocorre devido se encontrarem no mesmo ponto.

Como recurso de reverificação foi calculado os autovalores do sistema especificado acima para ambos, com o auxílio da função ‘eig’ no MATLAB. Assim pode ser obtido os valores dos polos desejados.

```
-2.0000 + 1.0000i
-2.0000 - 1.0000i
-1.4000 + 1.4000i
-1.4000 - 1.4000i
```

Figura 7: autovalores obtidos a partir da função eig do ganho de realimentação de estados

```
-18.0000 + 5.0000i
-18.0000 - 5.0000i
-20.0000 + 0.0000i
-20.0000 + 0.0000i
```

Figura 8: autovalores obtidos a partir da função eig do observador

Letra b)

Por último é especificado a questão a simulação do sistema com realimentação de estados e observador integrado, escolhendo alguns valores inicias e estimados para verificar graficamente. O simulink feito é necessário rodar o código fornecido abaixo, pois foi utilizado em seus blocos de ganho as matrizes necessárias a fim de proporcionar uma melhor visualização esquemática.

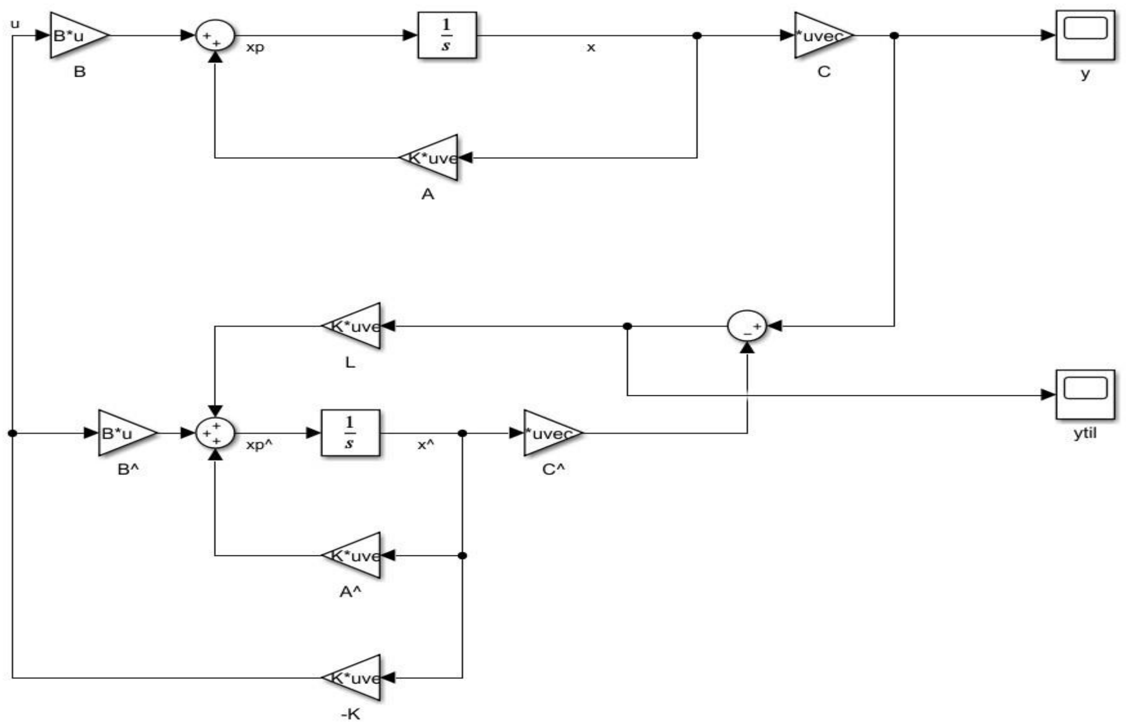

Figura 9: Simulink para o Sistema com realimentação de estados e Observador

	Valores de **x**	Valores de $\hat{\mathbf{x}}$
1º	[1,0,0,0]	[-1,0,0,0]
2º	[-1,0.1,0.4,0.5]	[2,2,2,2]
3º	[8,9,10,11]	[2,3,4,5]
4º	[2,3,2,2]	[1.2,5.4,4.5,3.3]

Tabela 1: Valores de x e $\hat{x}$ valores estimados

Note que a saída apresenta apenas valores com x_1, assim só é possível obter o erro que está representado como ytil $\widehat{x_1}$ e a saída y como x_1.

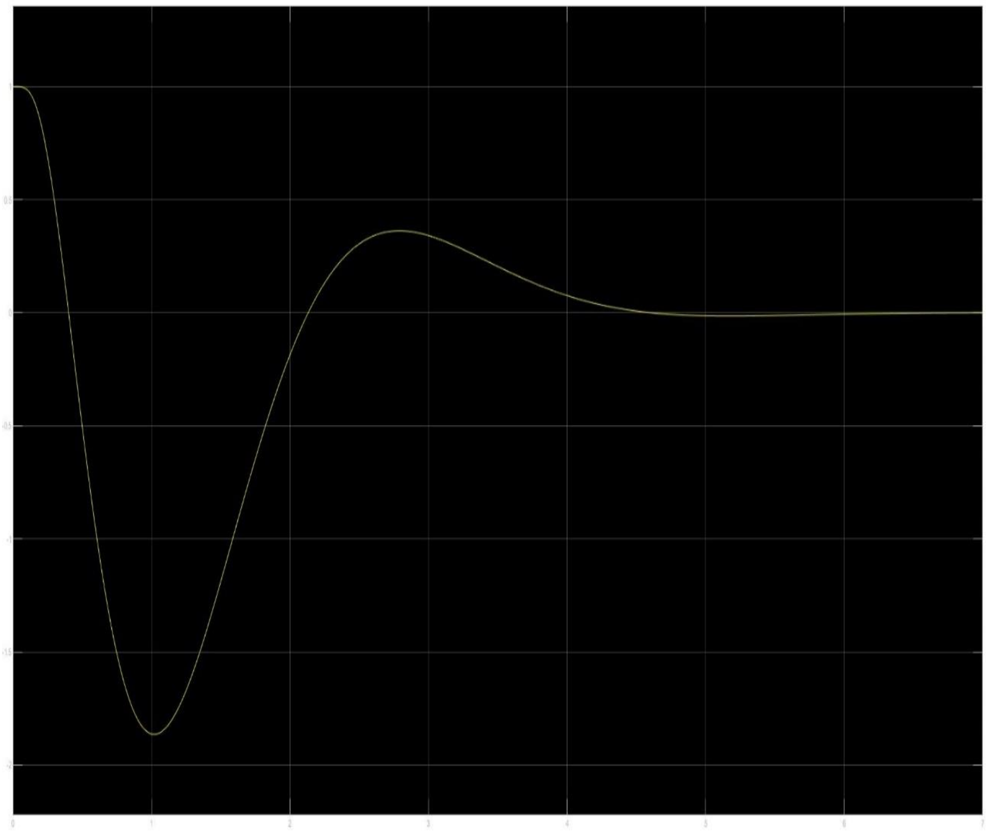

Figura 10: Saída y com os valores de 1º da tabela 1-Simulink

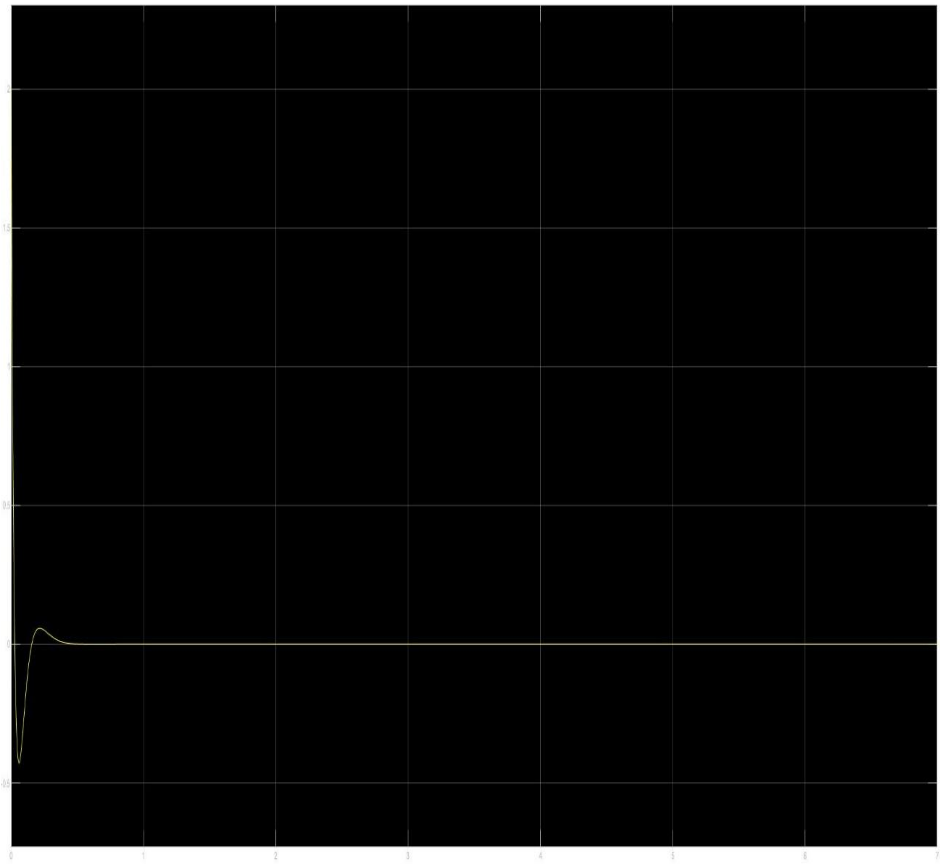

Figura 11: Saída ytil com os valores de 1º da tabela 1-Simulink

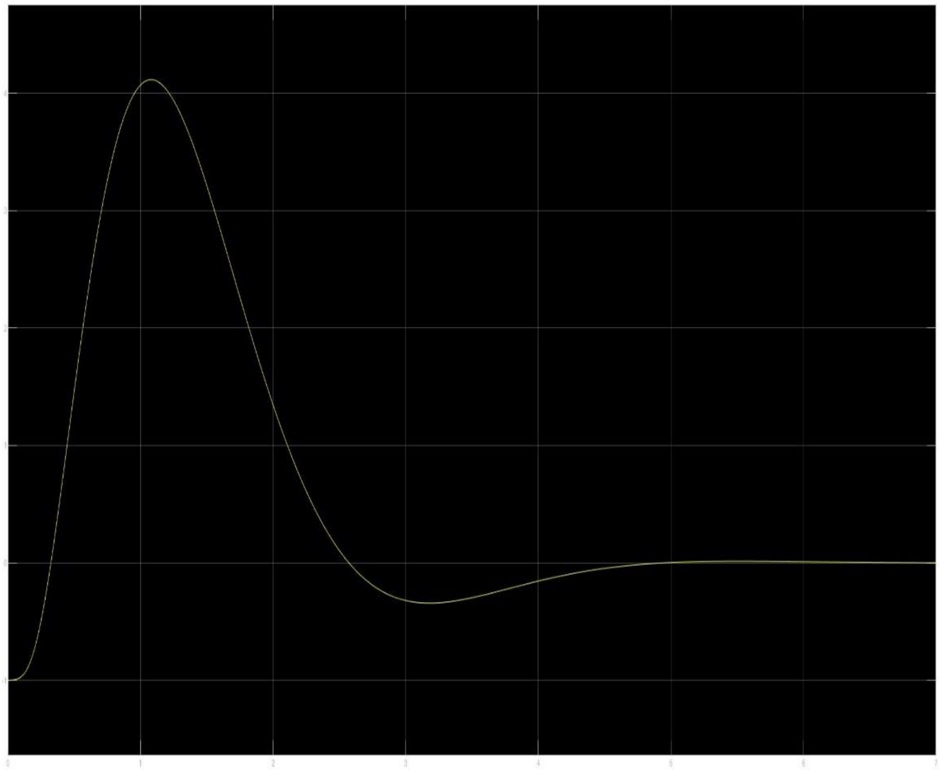

Figura 12: Saída y com os valores de 2º da tabela 1-Simulink

Figura 13: Saída ytil com os valores de 2º da tabela 1-Simulink

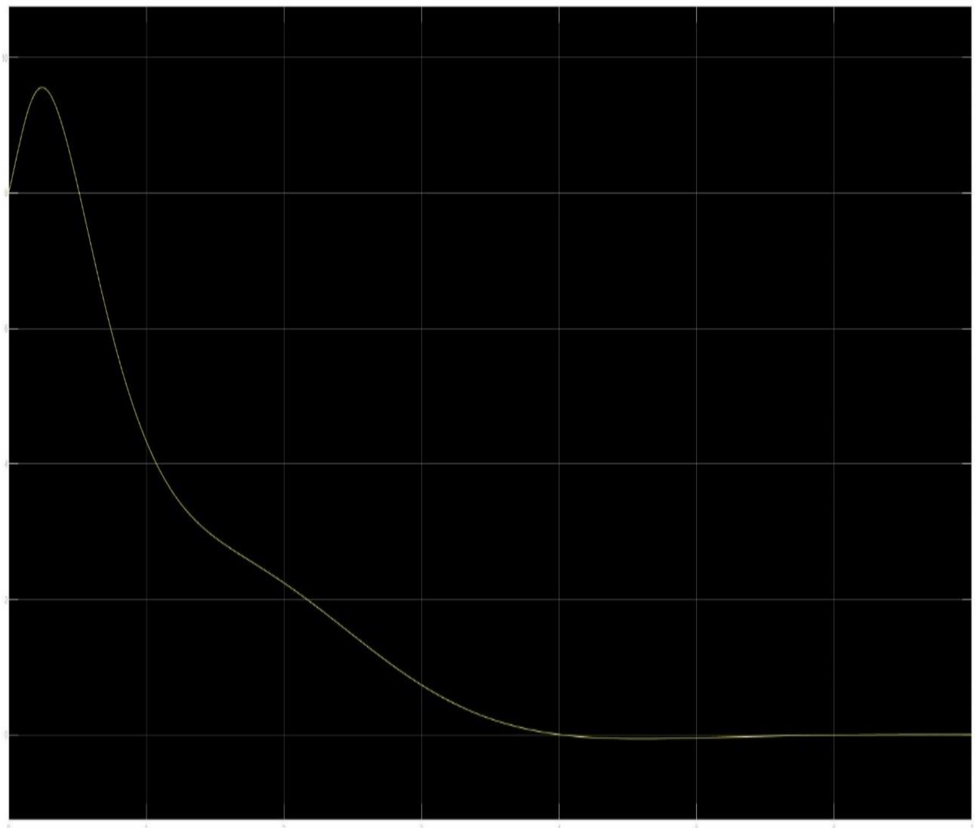

Figura 14: Saída y com os valores de 3º da tabela 1-Simulink

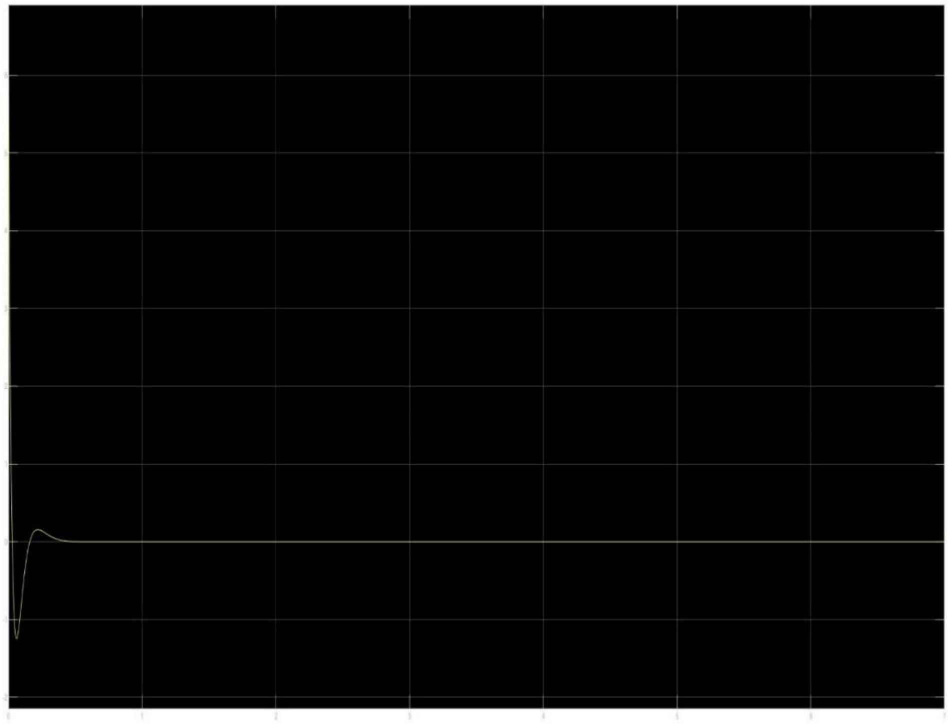

Figura 15: Saída ytil com os valores de 3º da tabela 1-Simulink

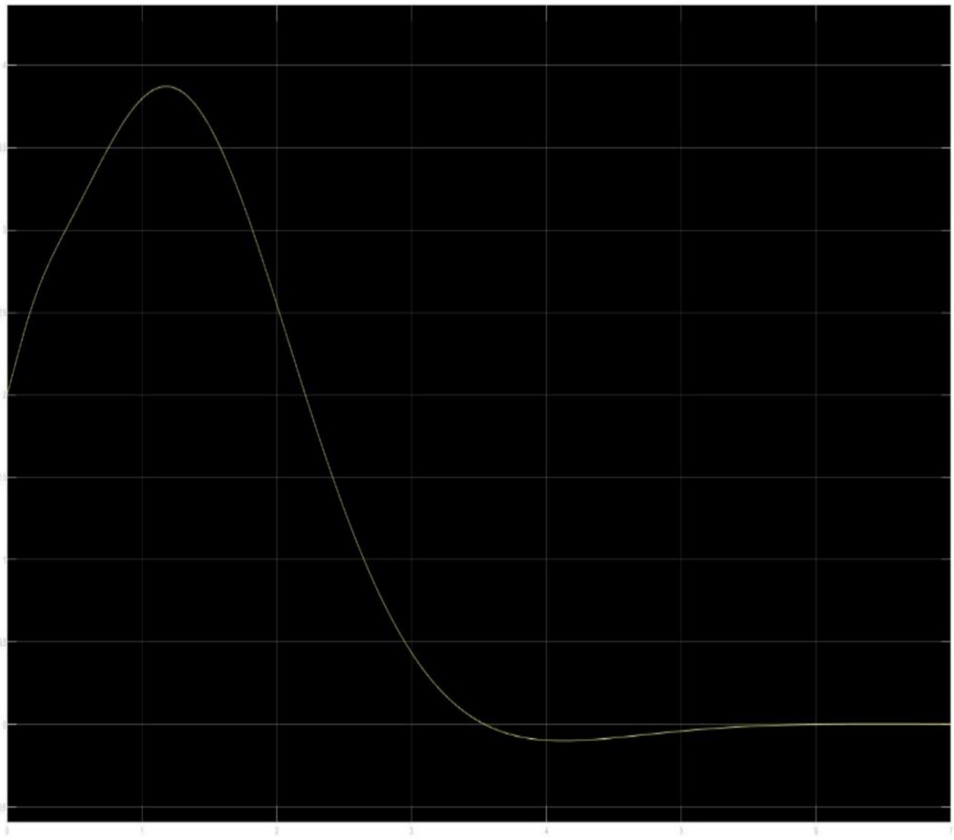

Figura 16: Saída y com os valores de 4º da tabela 1-Simulink

Figura 17: Saída ytil com os valores de 4º da tabela 1-Simulink

Por fim, para comparar com os resultados obtidos no Simulink, foi programado no MATLAB a saída de y e o erro estimado com o código fornecido em aula e com algumas alterações para o modelo da questão.

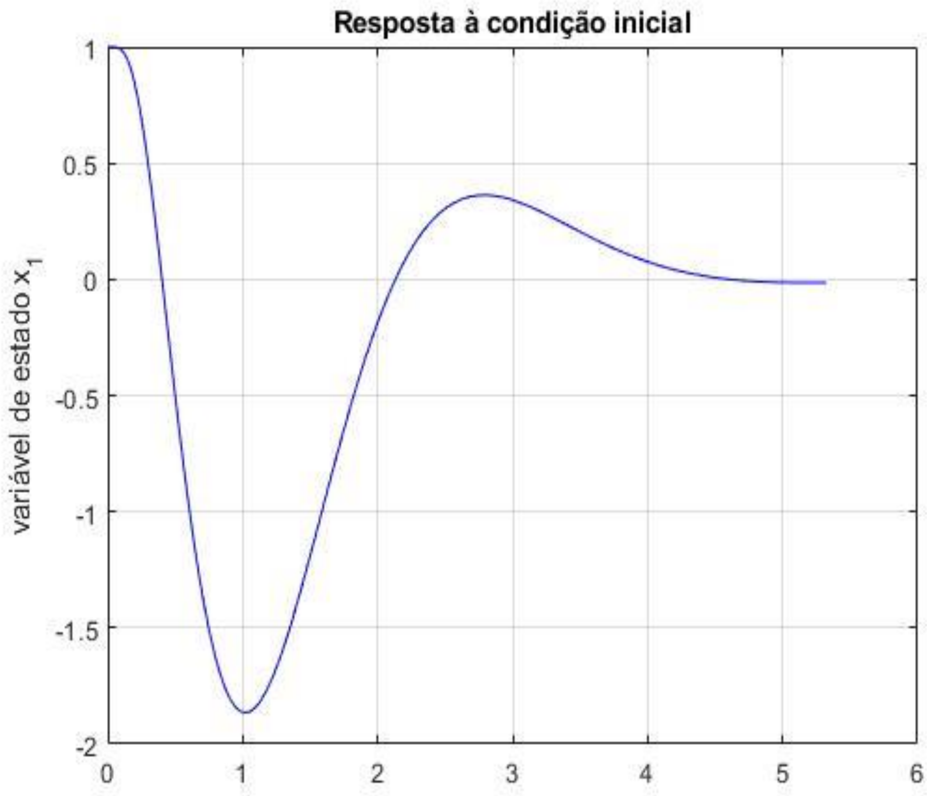

Figura 18: Saída y com os valores de 1º da tabela 1-Simulado

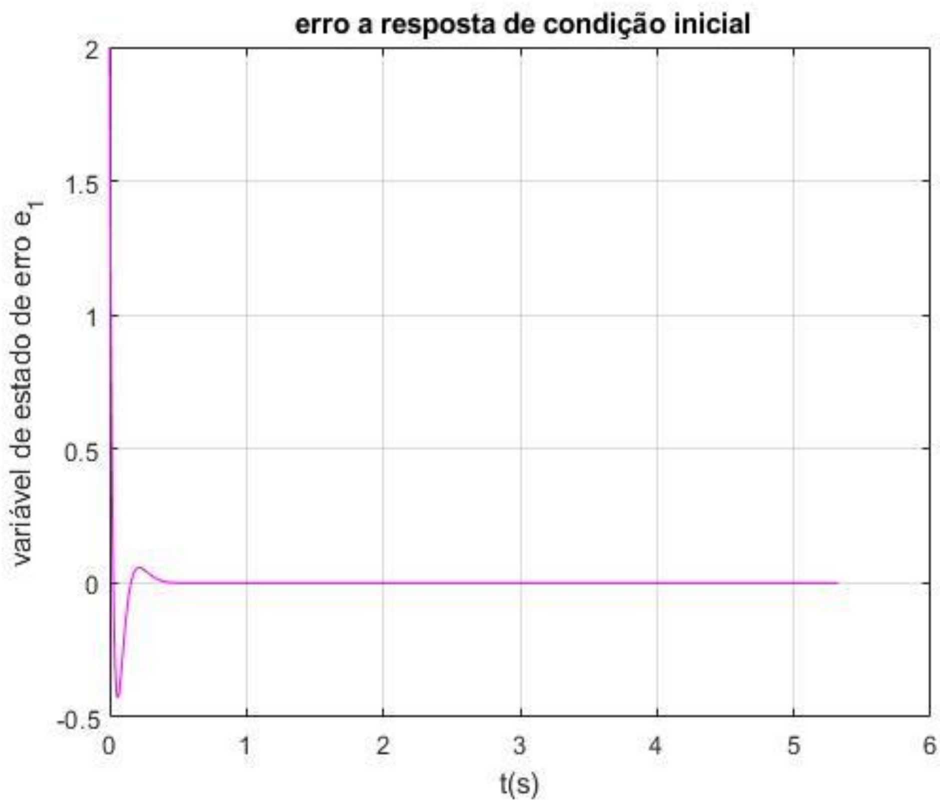

Figura 19: Saída ytil com os valores de 1º da tabela 1-Simulado

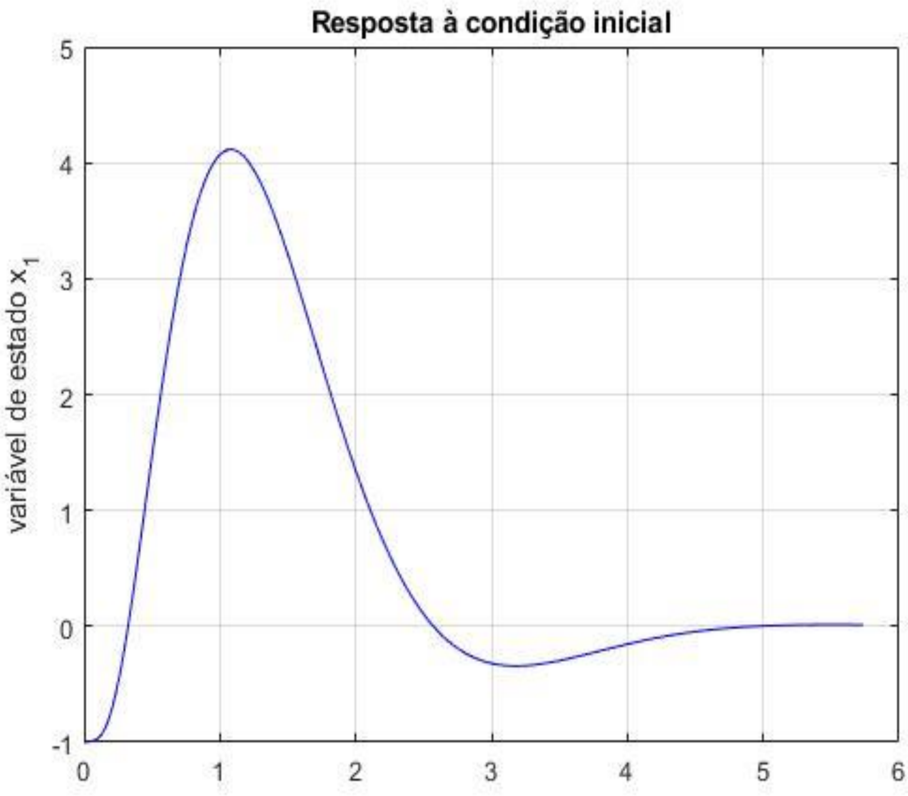

Figura 20: Saída y com os valores de 2º da tabela 1-Simulado

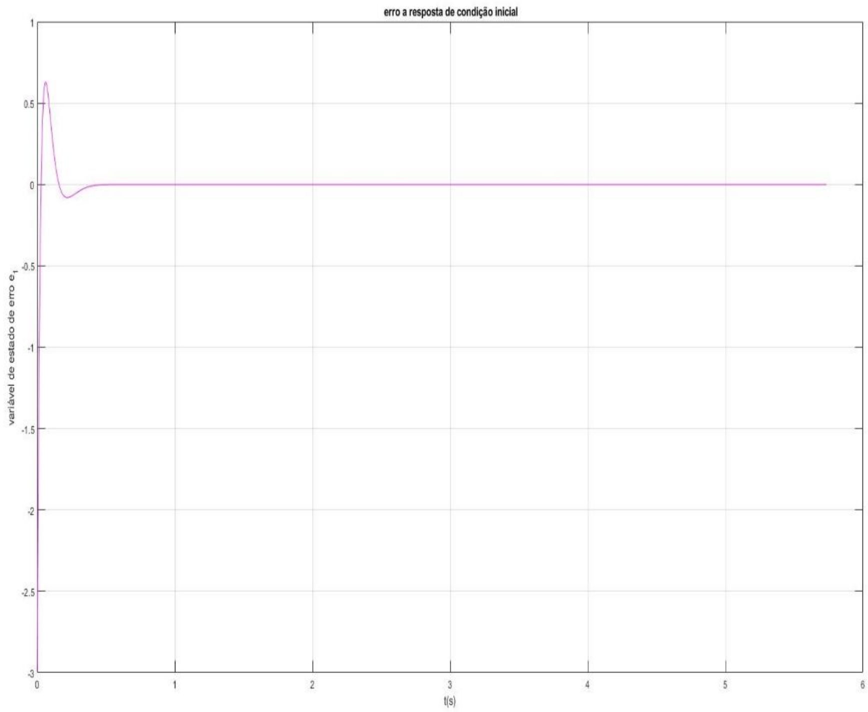

Figura 21: Saída ytil com os valores de 2º da tabela 1-Simulado

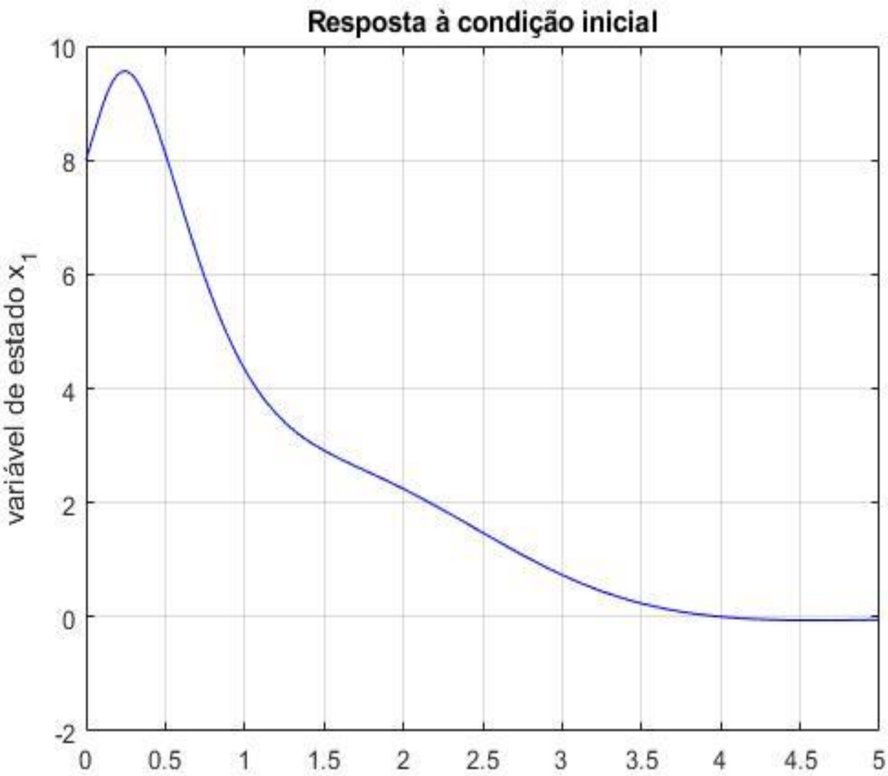

Figura 22: Saída y com os valores de 3º da tabela 1-Simulado

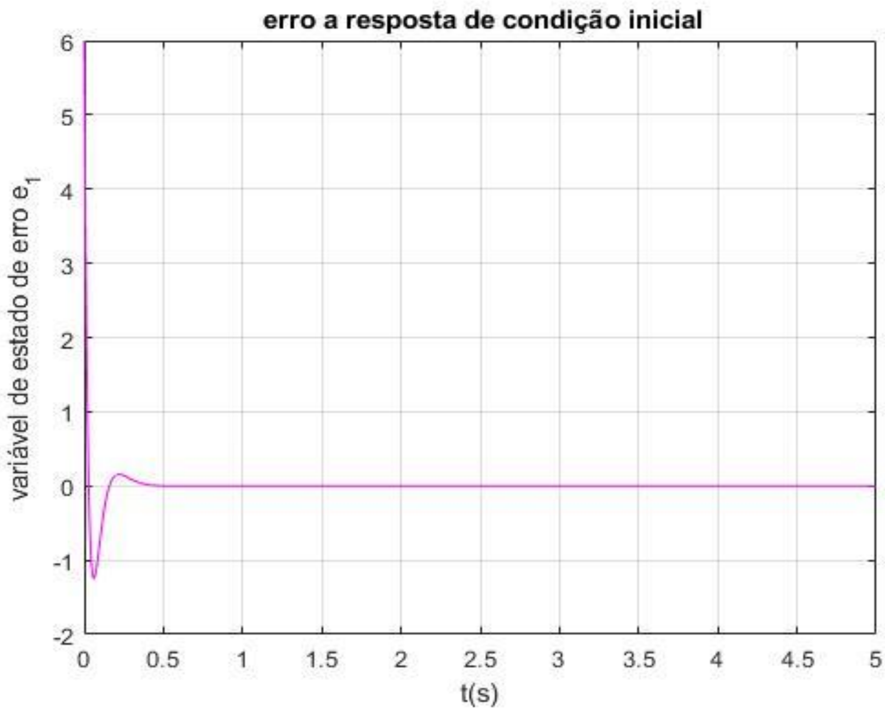

Figura 23: Saída ytil com os valores de 3º da tabela 1-Simulado

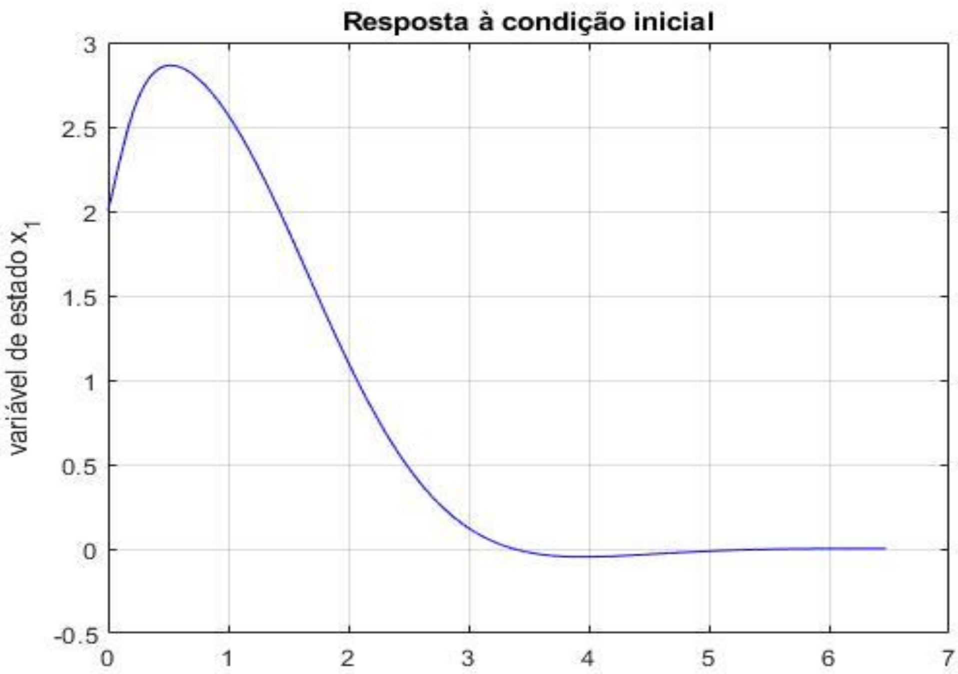

Figura 24: Saída y com os valores de 4º da tabela 1-Simulado

Figura 25:Saída ytil com os valores de 4º da tabela 1-Simulado

Código do MATLAB questão 2:

```
clc
clear
close all
```

```
syms s

%% Planta Modelada
A = [0 1 0 0; 0 0 1 0; 0 0 0 1; -2 -5 -1 -13];
B = [0; 0; 0; 1];
C = [1 0 0 0];
D = 0;
x0 =[2,3,2,2];


[nump,denp] = ss2tf(A,B,C,D);
F_planta = tf(nump,denp);



%% Polos desejados
Polos_ganho_realimentacao = [-1.4+1.4i, -1.4-1.4i, -2+1i, -2-1i];
Polos_ganho_observador = [-18+5i, -18-5i, -20, -20];

% Ganho realimentacao
K_realimentacao = acker(A, B, Polos_ganho_realimentacao);

% Ganho observador
L_observador = acker(A', C', Polos_ganho_observador)';

% Sistema apenas com controlador
Ak = A - B * K_realimentacao;
sys_realim_estado = ss(Ak, B, C, D);

%Sistema apenas com observador
Aob=A-L_observador*C;
sys_observador = ss(Aob,B,C,D);

Bob = B-K_realimentacao;
% mapa de polos e zeros sistema com controlador
figure();
pzmap(sys_realim_estado);
title("Sistema com realimentação de estados");
figure();
pzmap(sys_observador);
title("Sistema com o observador");

%verificando se os polos estão alocados corretamente:
polos_sys_realim_estado =eig(sys_realim_estado);
polos_sys_observador=eig(sys_observador);
disp(polos_sys_realim_estado);
disp(polos_sys_observador);
```

```
% função de transferencia do compensador

A_comp =A-B*K_realimentacao-L_observador*C;
[num,den] = ss2tf(A_comp,L_observador,-K_realimentacao,D);
F_comp = tf(num,den);

% Sistema completo
AA =[A-B*K_realimentacao,B*K_realimentacao;zeros(size(A)),A-
L_observador*C];
xc0 = [x0,0.8,0.5,-2.4,-1.3];
A_CO = (A-B*K_realimentacao-L_observador*C);
sys_completo1 = ss(AA, eye(8), eye(8), eye(8));
[y, t, x] = initial(sys_completo1, xc0);

x1 = [1 0 0 0 0 0 0 0]*x';
x2 = [0 1 0 0 0 0 0 0]*x';
x3 = [0 0 1 0 0 0 0 0]*x';
x4 = [0 0 0 1 0 0 0 0]*x';
e1 = [0 0 0 0 1 0 0 0]*x';
e2 = [0 0 0 0 0 1 0 0]*x';
e3 = [0 0 0 0 0 0 1 0]*x';
e4 = [0 0 0 0 0 0 0 1]*x';

% Define cores para os estados e erros
cores_estados = {'b', 'g', 'r', 'c'};
cores_erros = {'m', 'y', 'k', [0.5 0 0.5]};

% variaveis de estados e erros estimados
figure
plot(t, x1, 'Color', cores_estados{1}); grid
title('Resposta à condição inicial')
ylabel('variável de estado x_1')

figure();
plot(t, e1, 'Color', cores_erros{1}); grid
title(" erro a resposta de condição inicial");
ylabel('variável de estado de erro e_1')
xlabel('t(s)')

% Comparação das respostas homogêneas do sistema apenas com
controlador e do sistema com controlador-observador
figure
h1 = plot(t,x1, 'r'); % Sistema com controlador-observador em vermelho
hold on
initialplot(sys_realim_estado,x0);

title('Respostas à condição inicial')
legend('Sistema com controlador-observador','Sistema apenas
com controlador')
```

Questão 3:

Letra a)

Na última questão apresenta uma planta analógica e pede para que seja projetado um compensador digital com a margem de fase de 50º e uma margem de ganho maior ou igual a 10 dB.

Para isso foi utilizado o código do primeiro trabalho, o qual calcula um compensador analógico a partir do diagrama de bode, visto que é preciso de um compensador avanço de fase.

Obteve a seguinte função de transferência do compensador:

$$Gc = \frac{49{,}23s + 145.1}{s + 14.51}$$

Por conseguinte, conferiu se o compensador atendia os requisitos:

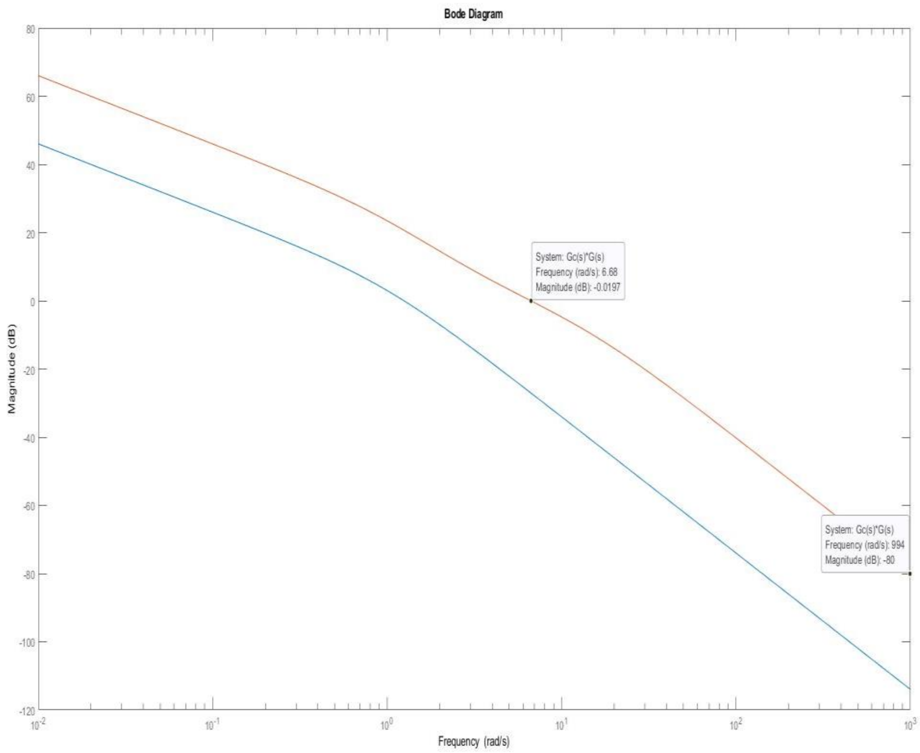

Figura 26: Diagrama de Bode de Magnitude de G(s) e Gc(s)*G(s)

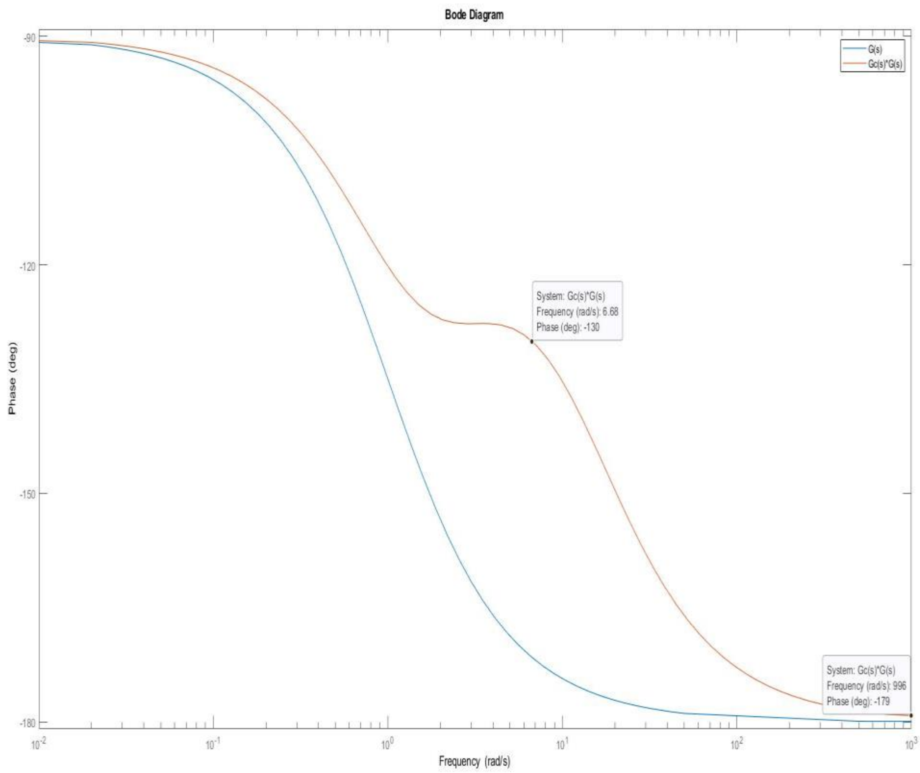

Figura 27: Diagrama de Bode de Fase de G(s) e Gc(s)*G(s)

```
Ganho compensado =
   Inf

Fase compensada =
   50.0044
```

Figura 28: Resultados da Margem de Ganho e Margem de Fase do Compensado

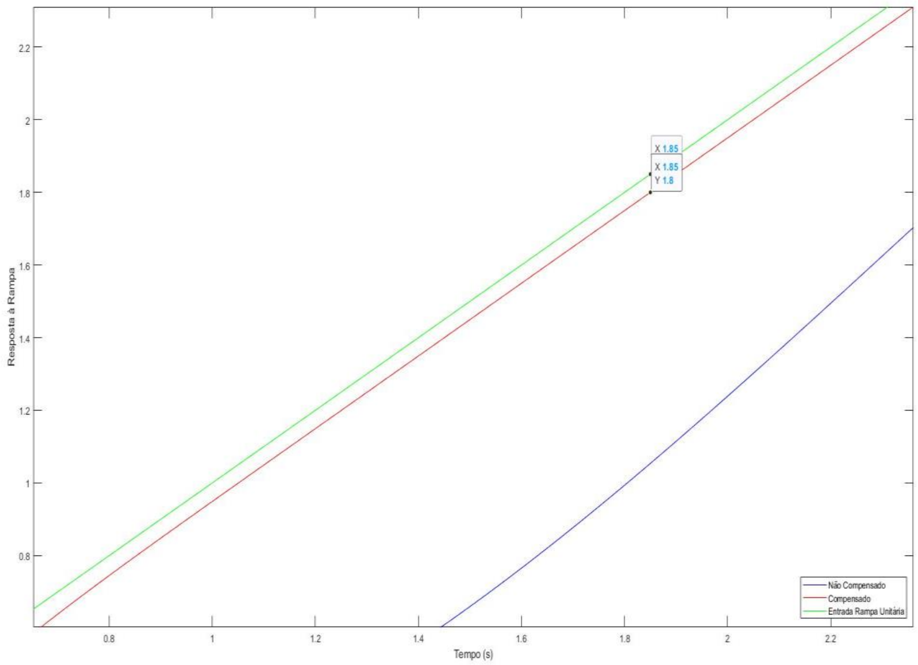

Figura 29 : Resposta a Rampa – Conferindo o erro do regime permanente

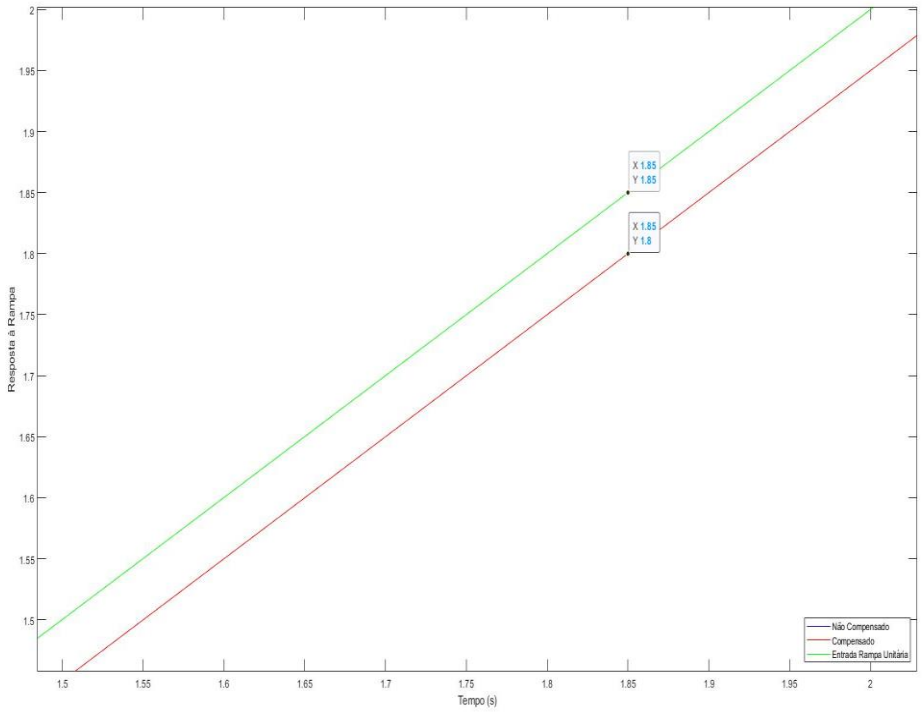

Figura 30: Resposta a Rampa- Erro em Regime Permanente atingido

Calculando o erro a partir de 1,85 -1,8 chegamos ao erro requerido de 0,05. Assim é notório que todos os requisitos da questão foram atendidos.

Letra b)

Em seguida o compensador modelado e a planta foram discretizados a partir de três amostragens de tempo com a relação de tustin, ou seja:

$\frac{0,15}{\omega_{\varphi M}}$; $\frac{0,5}{\omega_{\varphi M}}$ e foi estipulado um valor abaixo de $\frac{0,15}{\omega_{\varphi M}}$, sendo ele $\frac{0,01}{\omega_{\varphi M}}$

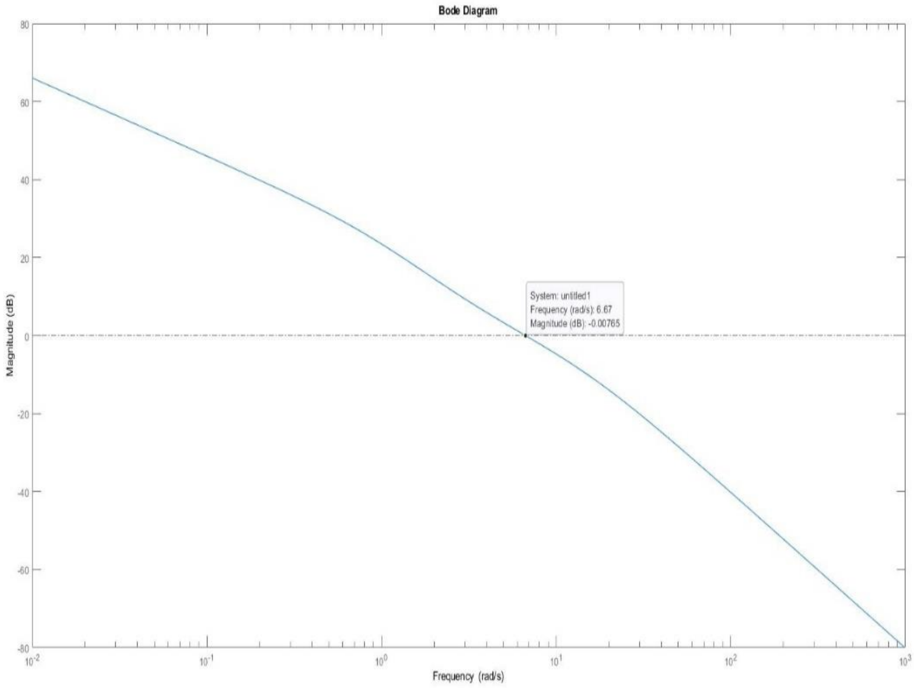

Figura 31:Gráfico da Magnitude de Bode de Gc(s)*G(s)

$\omega_{\varphi M}$ é a frequência (rad/s) de 0 dB da curva de magnitude da resposta em frequência em malha aberta do sistema contínuo compensado;

$$\omega_{\varphi M} = 6,67\ rad/s.$$

Os valores obtidos a partir dessa frequência:

$$T_{s1} = \frac{0,15}{\omega_{\varphi M}} = 0,0225.$$

$$T_{s2} = \frac{0,5}{\omega_{\varphi M}} = 0,750.$$

$$T_{s3} = \frac{0,01}{\omega_{\varphi M}} = 0,0015.$$

Letra c)

Após análise do compensador simulou o sistema em malha fechada contínuo compensado e com os dados amostrados compensado tanto para a entrada em degrau unitário quanto para entrada em rampa unitária e analisou o desempenho do sistema de controle

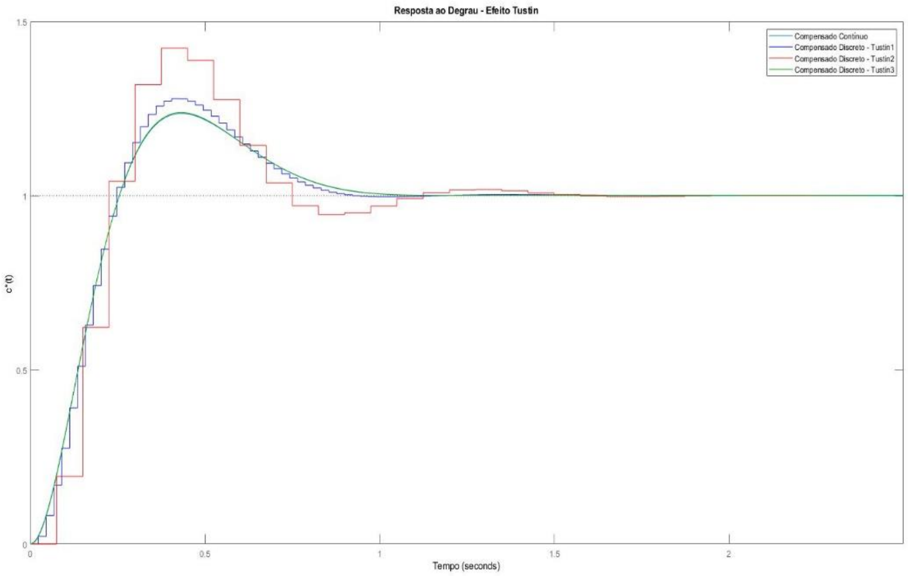

Figura 32: Gráfico Resposta ao Degrau - Tustin

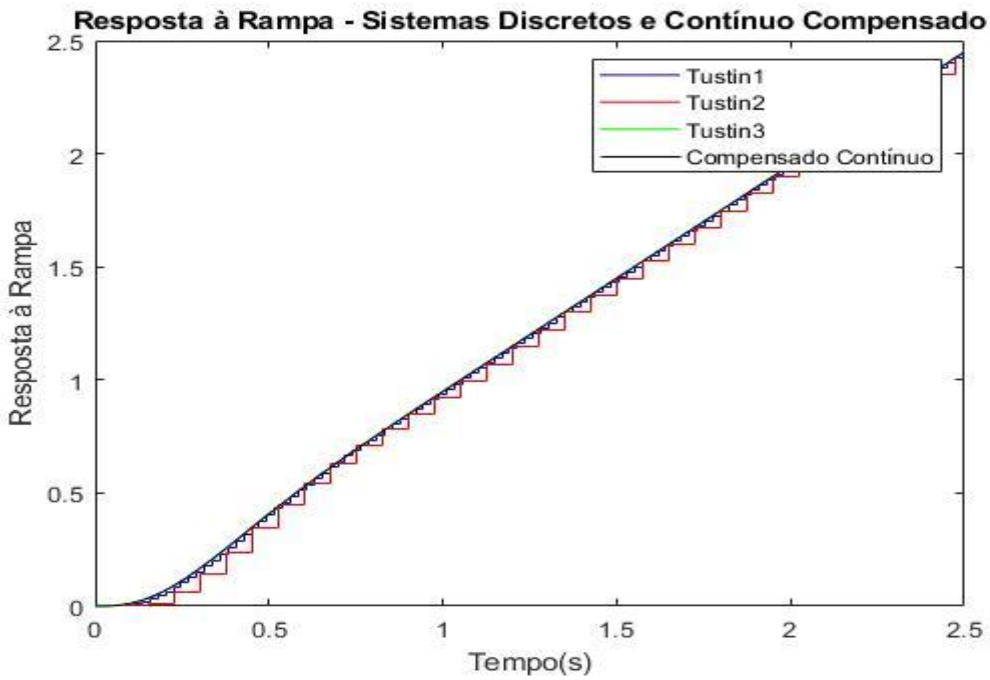

Figura 33: Gráfico Resposta à rampa para o sistema compensado.

Letra d)

Por fim, após a análise dos gráficos da resposta ao degrau e da resposta à rampa, foi escolhido o compensador que que utiliza o método do Tustin com o terceiro valor.

Esse apresenta a seguinte Função de transferência no tempo discreto:

$$\frac{X(z)}{E(z)} = \frac{48{,}81z - 48{,}59}{z - 0{,}9784}$$

Assim para obter o fluxograma do controle digital realizamos as seguintes manipulações:

$$\frac{X(z)}{E(z)} = \frac{48{,}81z - 48{,}59}{z - 0{,}9784} * \frac{z^{-1}}{z^{-1}}$$

$$\frac{X(z)}{E(z)} = \frac{48{,}81 - 48{,}59z^{-1}}{1 - 0{,}9784z^{-1}}$$

$$X(z)(\,1 - 0{,}9784z^{-1}) = E(z)(48{,}81 - 48{,}59z^{-1})$$

Convertendo para o tempo discreto temos:

$$x^*(t) - 0{,}9784x^*(t-T) = 48{,}81e^*(t) - 48{,}59e^*(t-T)$$

$$x^*(t) = \;0{,}9784x^*(t-T) + \;\;48{,}81e^*(t) - 48{,}59e^*(t-T)$$

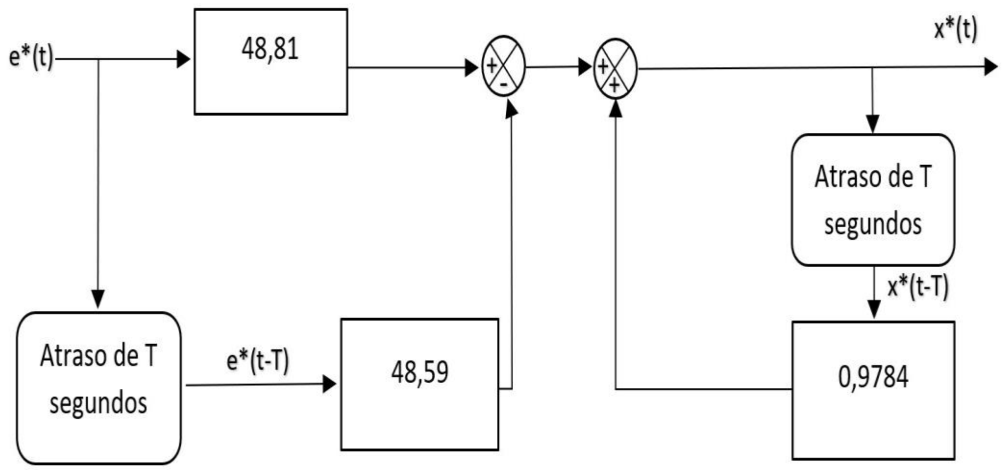

Figura 34: Fluxograma do controle digital

Código do MATLAB questão 3:

```
clc;
clear;
close all;
%Tempo Final
Tf = 2.5;

% Correção de fase e erro de regime permanente
aux = 0;
esst = 0.05;
MF =50;
MG = 10;

% Definir a Função de Transferência
num = 2;
den = [1 1 0];
G = tf(num,den);

% Cálculo do ganho baseado em 25% de erro
Kv = 1 / esst; derivador = tf([1 0], 1); GKv = G * derivador; K = Kv / dcgain(GKv);

% Diagrama de Bode
[ganho, fase] = margin(K * G); ganho = squeeze(ganho);
% Magnitude do ganho
valores_db = 20 * log10(ganho); % Converter para dB

while fase < MF || valores_db < MG
   % Margem de Ganho e Margem de Fase
   MFutil = MF - fase + aux; alfa =(1 - sind(MFutil)) / (1 + sind(MFutil));

   ganho_desejado_dB = -20 * log10(1 / sqrt(alfa)); [GANHO_KG, FASE_KG,w]
= bode(K * G);

   % Encontrar o índice da frequência mais próxima
   [~,indice_frequencia] = min(abs(20 * log10(GANHO_KG) -
ganho_desejado_dB));

   % Obter o ganho correspondente à frequência desejada
   Wn = w(indice_frequencia); T = 1 / (sqrt(alfa) * Wn); Wz = 1 / T; Wp = 1 /(alfa
* T);

   % Compensador de Avanço de Fase
   Kc = K / alfa; num_Gc = [Kc, Kc *Wz]; den_Gc = [1, Wp]; Gc = tf(num_Gc,
den_Gc);
   [ganho, fase] = margin(Gc * G); aux = aux + 0.01;
end

%Exibir os resultados
disp('Ganho compensado = '); disp(20 *log10(ganho)); disp('Fase compensada
= '); disp(fase);
```

```
%Diagrama de Bode
figure;
bode(G);
hold on
bode(Gc * G); legend('G(s)', 'Gc(s)*G(s)');

% Simulação e Resposta
t = 0:0.01:Tf; % Tempo de simulação
u_step = ones(size(t)); % Degrau unitário
u_ramp = t; % Rampa unitária

G_malha_fechada = feedback(G, 1);

% Sistema não compensado
[y_step_nao_compensado, ~] = step(G_malha_fechada, t);
[y_rampa_nao_compensado, ~] = lsim(G_malha_fechada, u_ramp, t);

Sist_comp_malha_fechada = feedback(Gc * G, 1);

% Sistema compensado
[y_step_compensado, ~] = step(Sist_comp_malha_fechada, t);
[y_rampa_compensado, ~] = lsim(Sist_comp_malha_fechada, u_ramp, t);
%
Q = 1 + 0.*t;

% Plot da resposta ao degrau
figure;
plot(t,y_step_nao_compensado, 'b', t, y_step_compensado, 'r', t, Q, 'g');
xlabel('Tempo (s)');
ylabel('Resposta ao Degrau');
legend('NãoCompensado', 'Compensado', 'Entrada Degrau Unitário');

figure();
plot(t, y_rampa_nao_compensado, 'b', t,y_rampa_compensado, 'r', t, t, 'g');
xlabel('Tempo (s)');
ylabel('Resposta à Rampa');
legend('Não Compensado', 'Compensado','Entrada Rampa Unitária', 'Location',
'southeast');

% Compensador Discreto usando método de Tustin


freq_zero_dB = 6.67;

% Exibir a frequência encontrada disp('Frequência em 0 dB = ');
% disp(freq_zero_dB);

% tempo de amostragem com Tustin
Ts1 = 0.15 /freq_zero_dB;
```

```
Ts2 = 0.5 /freq_zero_dB;
Ts3 = 0.01 /freq_zero_dB;

% Intervalo de Amostragem
T1 = 0:Ts1:Tf;
T2 = 0:Ts2:Tf;
T3 = 0:Ts3:Tf;

%Planta do sistema no Tempo Discreto
G_z1= c2d(G,Ts1, 'zoh');
G_z2 = c2d(G,Ts2, 'zoh');
G_z3 = c2d(G,Ts3, 'zoh');

% Compensador no tempo discreto pelo método de Tustin
Gc_z_tustin1 = c2d(Gc, Ts1, 'tustin');
Gc_z_tustin2 = c2d(Gc, Ts2, 'tustin');
Gc_z_tustin3 = c2d(Gc, Ts3, 'tustin');

% Malha fechada com compensador e planta com realimentação unitária
T_z_tustin1 = feedback(Gc_z_tustin1 * G_z1,1);
T_z_tustin2 = feedback(Gc_z_tustin2 * G_z2,1);
T_z_tustin3 = feedback(Gc_z_tustin3 * G_z3,1);

% Resposta ao Degrau para Sistema Discreto
figure;
step(Sist_comp_malha_fechada); % Resposta ao Degrau do Sistema
Compensado Analógico
hold on;
step(T_z_tustin1,T1,'b'); % Resposta ao Degrau do Sistema Compensado
Discreto - Tustin
title('Resposta ao Degrau - Efeito Tustin');
hold on;
step(T_z_tustin2,T2,'r'); % Resposta ao Degrau do Sistema Compensado
Discreto - Tustin
hold on;
step(T_z_tustin3,T3,'g'); % Resposta ao Degrau do Sistema Compensado
Discreto - Tustin
legend('Compensado Contínuo', 'Compensado Discreto - Tustin1',
'Compensado Discreto - Tustin2', 'Compensado Discreto - Tustin3');
xlabel('Tempo');
ylabel('c*(t)');



% Resposta à Rampa para Sistema Discreto
u_ramp_tustin1 = T1; % Rampa unitária para sistema discreto
u_ramp_tustin2 = T2; % Rampa unitária para sistema discreto
u_ramp_tustin3 = T3; % Rampa unitária para sistema discreto

[y_rampa_tustin1, Tz1] = lsim(T_z_tustin1, u_ramp_tustin1);
```

```
[y_rampa_tustin2, Tz2] = lsim(T_z_tustin2, u_ramp_tustin2);
[y_rampa_tustin3, Tz3] = lsim(T_z_tustin3, u_ramp_tustin3);

% % Resposta à Rampa para Sistema Contínuo Compensado e o tempo
discreto
% com tustin
[y_rampa_compensado_continuo, ~] = lsim(Sist_comp_malha_fechada,
u_ramp,t);
figure();
bode(Gc * G);
figure;
stairs(Tz1,y_rampa_tustin1,'b');
hold on;
stairs(Tz2,y_rampa_tustin2,'r');
hold on;
stairs(Tz3,y_rampa_tustin3,'g');
hold on;
plot(t, y_rampa_compensado_continuo, 'k');
xlabel('Tempo(s)');
ylabel('Resposta à Rampa');
legend('Tustin1', 'Tustin2', 'Tustin3','Compensado Contínuo');
title('Resposta à Rampa - Sistemas Discretos e Contínuo Compensado');
```