

WELLINGTON NOGUEIRA - PME5009 2022 - LISTA 1

Exercício 1:

Sistema1 a) representado em espaço de estados

```
A = [[0,1,0];[0,0,1];[-0.005,-0.11,-0.7]];
B = [[0.1];[-0.2];[-0.1]];
C = [0,0,1];
D = [0];
ss_Model_1 = ss(A,B,C,D)
```

Sistema2 b) representado em espaço de estados

```
A2 = [[-0.2,-0.2,0.4];[0.5,0,1];[0,-0.4,-0.4]];
B2 = [[0.01];[0];[0.05]];
C2 = [1,0,0];
D2 = [0];
ss_Model_2 = ss(A2,B2,C2,D2)
```

i) Verificar e Justificar se os sistemas são controláveis e Observáveis:

```
% Matrizes de controlabilidade
Matriz_Controlabilidade_1 = ctrb(A,B)
Matriz_Controlabilidade_2 = ctrb(A2,B2)
% Verificando o posto (rank) da matrizes de controlabilidade
Posto_Matriz_Contrl_1 = rank(Matriz_Controlabilidade_1)
Posto_Matriz_Contrl_2 = rank(Matriz_Controlabilidade_2)

% Matrizes de observabilidade:
Matriz_Observabilidade_1 = obsv(A,C)
Matriz_Observabilidade_2 = obsv(A2,C2)
% Verificando o posto (rank) das matrizes de observabilidade
Posto_Matriz_Obsv_1 = rank(Matriz_Observabilidade_1)
Posto_Matriz_Obsv_2 = rank(Matriz_Observabilidade_2)
%
```

```
Matriz_Controlabilidade_1 =
    0.1000    -0.2000    -0.1000
   -0.2000    -0.1000     0.0915
   -0.1000     0.0915    -0.0520
```

```
Matriz_Controlabilidade_2 =
    0.0100     0.0180    -0.0226
         0     0.0550    -0.0110
    0.0500    -0.0200    -0.0140
```

```
Posto_Matriz_Contrl_1 = 3
```

```
Posto_Matriz_Contrl_2 = 3
```

```
Matriz_Observabilidade_1 =
```

```
      0      0  1.0000
-0.0050 -0.1100 -0.7000
 0.0035  0.0720  0.3800
```

```
Matriz_Observabilidade_2 =
```

```
 1.0000      0      0
-0.2000 -0.2000  0.4000
-0.0600 -0.1200 -0.4400
```

```
Posto_Matriz_Obsv_1 = 3
```

```
Posto_Matriz_Obsv_2 = 3
```

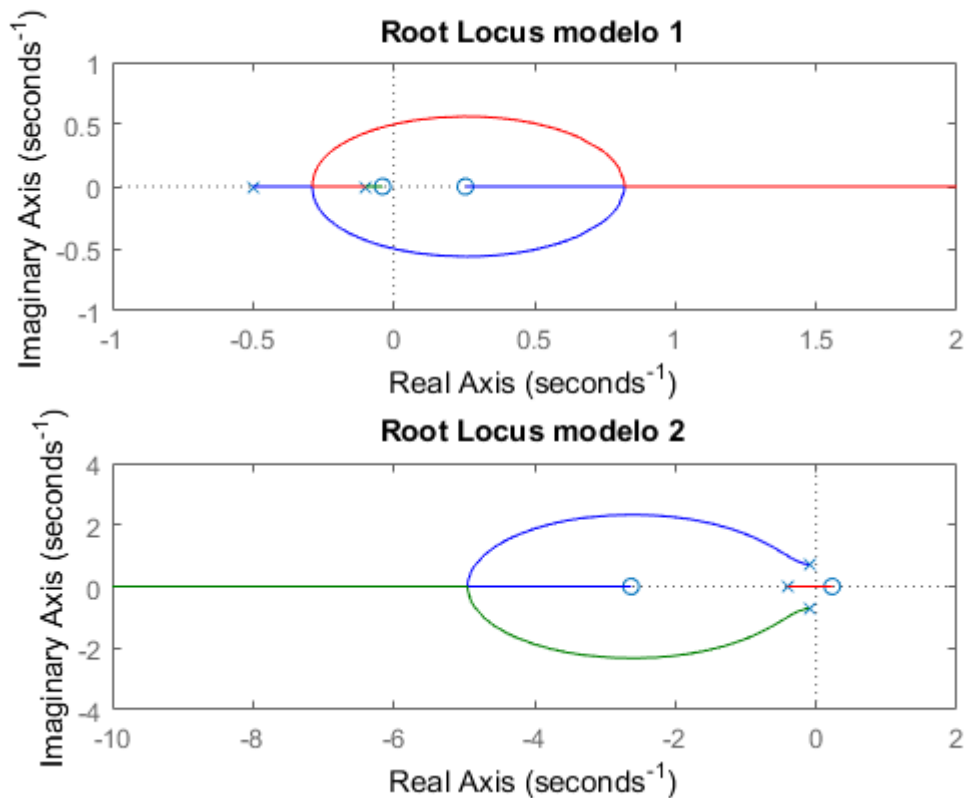
R) Ambos os postos (rank) das matrizes de controlabilidade e observabilidade são iguais a ordem do sistema, o que significa que temos uma solução linearmente independente que relaciona entradas com cada um dos estados (no caso da matriz de controlabilidade) e uma solução linearmente independente que relaciona as saídas com cada um dos estados (no caso da matriz de observabilidade). Concluimos então que o sistema é completamente controlável e completamente observável.

ii) Verificar e Justificar se são estáveis em malha aberta

```
Estabilidade_Sistema1 = eig(A)
Estabilidade_Sistema2 = eig(A2)
tf_model1 = ss(A,B,C,D);
tf_model2 = ss(A2,B2,C2,D2);
figure;
subplot(211);
rlocus(tf_model1);
title('Root Locus modelo 1');
subplot(212);
rlocus(tf_model2);
title('Root Locus modelo 2');
```

```
Estabilidade_Sistema1 =
-0.5000
-0.1000
-0.1000
```

```
Estabilidade_Sistema2 =
-0.4000 + 0.0000i
-0.1000 + 0.7000i
-0.1000 - 0.7000i
```



R) Eigen Value (auto valor) da matriz A, representa os polos do sistema e sua parte real negativa indica que o sistema é estável e irá convergir seguindo a direção apontada pelos auto valores da matriz. Nesse caso ambos os sistemas *Sistema1* e *Sistema2* é estável em malha aberta, pois todos os auto valores possuem parte real negativa indicando que seus polos estão no semiplano complexo esquerdo.

iii) Simular por 60 seg para entrada em degrau unitário e impulso unitário

```

ke1_B1 = 0;
ke2_B1 = 0;
ke3_B1 = 0;

Aobs_B1 = A2;
Bobs_B1 = B2;
Cobs_B1 = C2;
Dobs_B1 = D2;
keobs_B1 = [(0);(0);(0)];

ke1_A1 = 0;
ke2_A1 = 0;
ke3_A1 = 0;

Aobs_A1 = A;
Bobs_A1 = B;

```

```

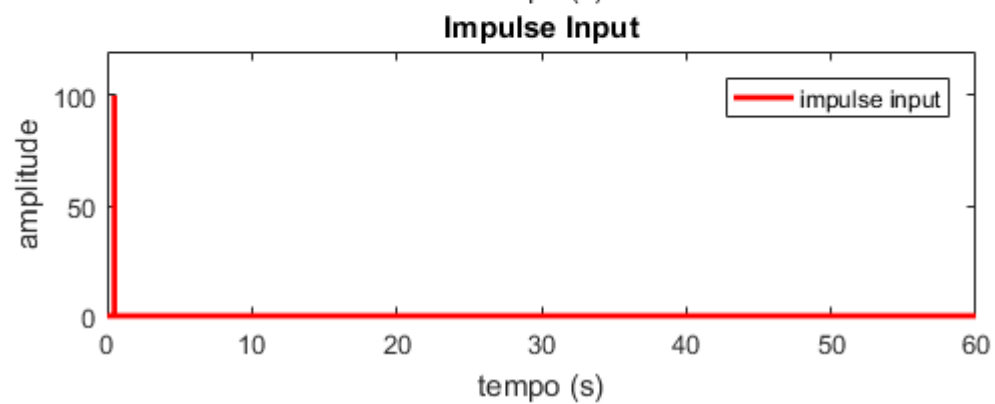
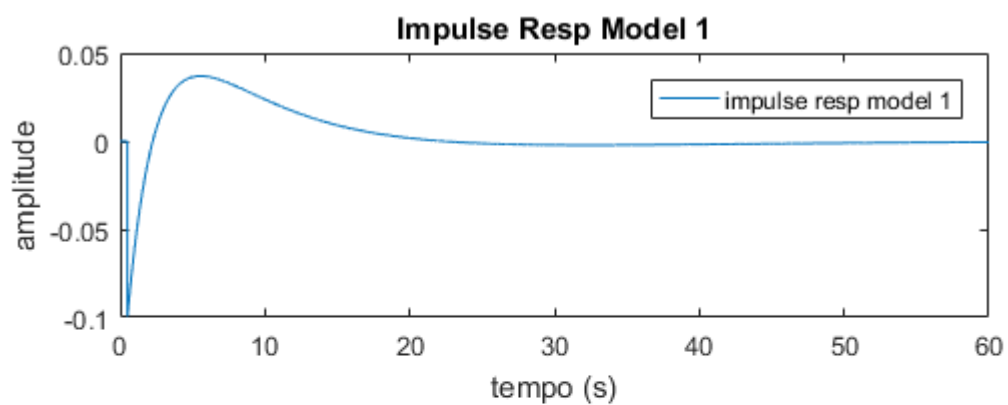
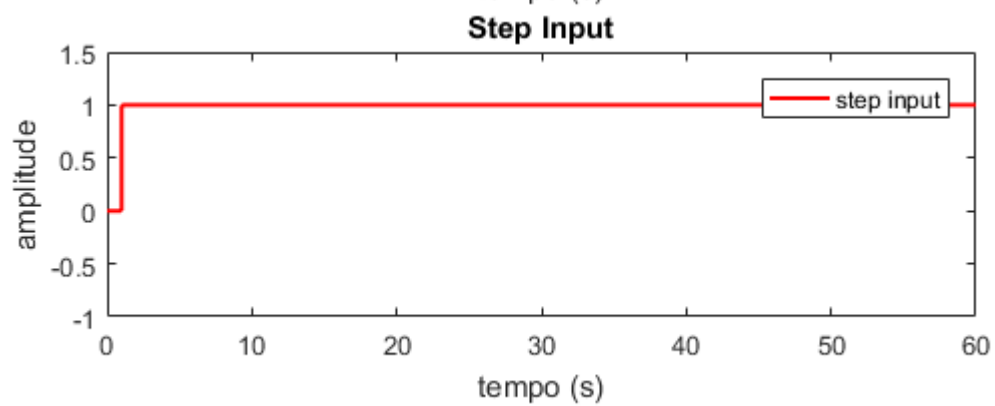
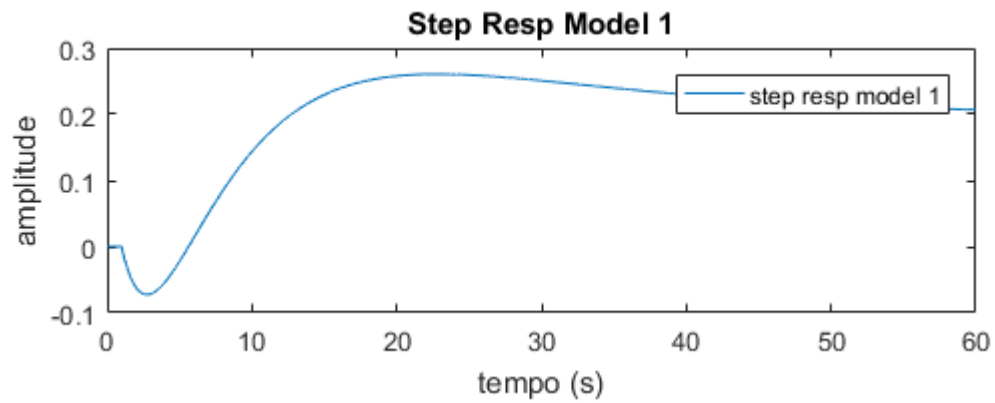
Cobs_A1 = C;
Dobs_A1 = D;
Keobs_A1 = [(0);(0);(0)];

t_sim = 60;
sim('Exercicio_1_sim',t_sim);

figure;
subplot(211);
plot(step_resp_model1);
title('Step Resp Model 1');
xlabel('tempo (s)');
ylabel('amplitude');
legend({'step resp model 1'},'Location','northeast');
hold;
subplot(212);
plot(step_input,'-r','Linewidth',1.5);
axis([0 t_sim -1 1.5]);
title('Step Input');
xlabel('tempo (s)');
ylabel('amplitude');
legend({'step input'},'Location','northeast');

figure
subplot(211);
plot(impulse_resp_model1);
title('Impulse Resp Model 1');
xlabel('tempo (s)');
ylabel('amplitude');
legend({'impulse resp model 1'},'Location','northeast');
subplot(212);
plot(impulse_input,'-r','Linewidth',2);
axis([0 t_sim 0 120]);
title('Impulse Input');
xlabel('tempo (s)');
ylabel('amplitude');
legend({'impulse input'},'Location','northeast');

```



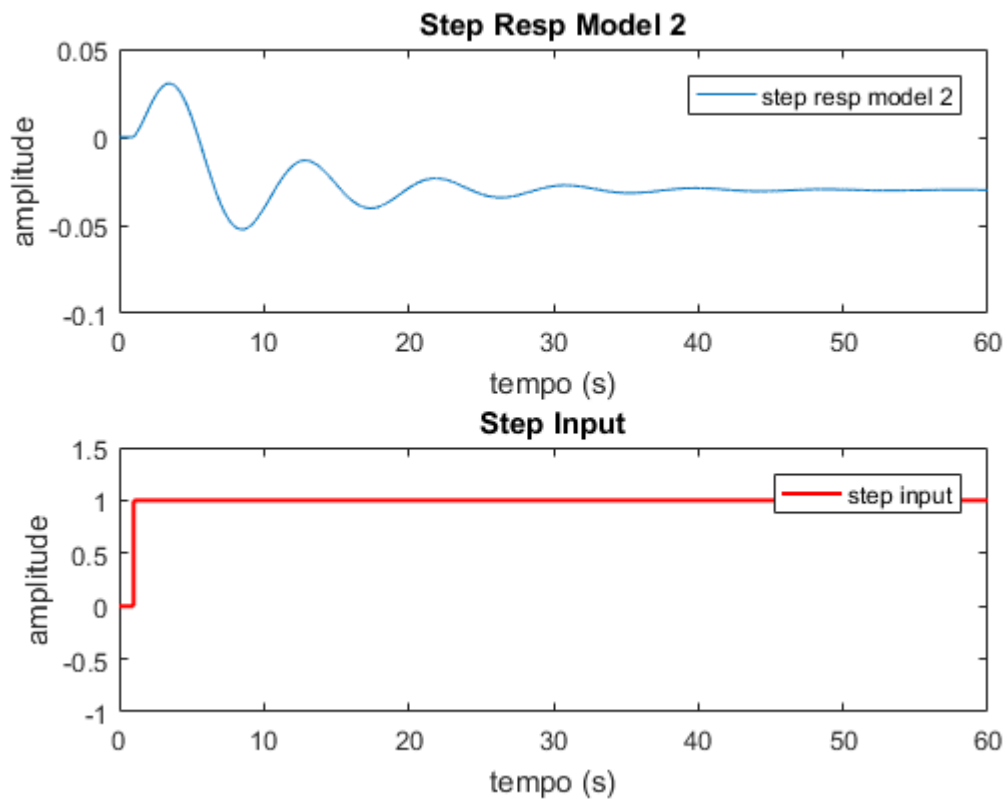
Obs.: Podemos notar que o sistema converge sendo estável em malha aberta como verificado no exercício anterior.

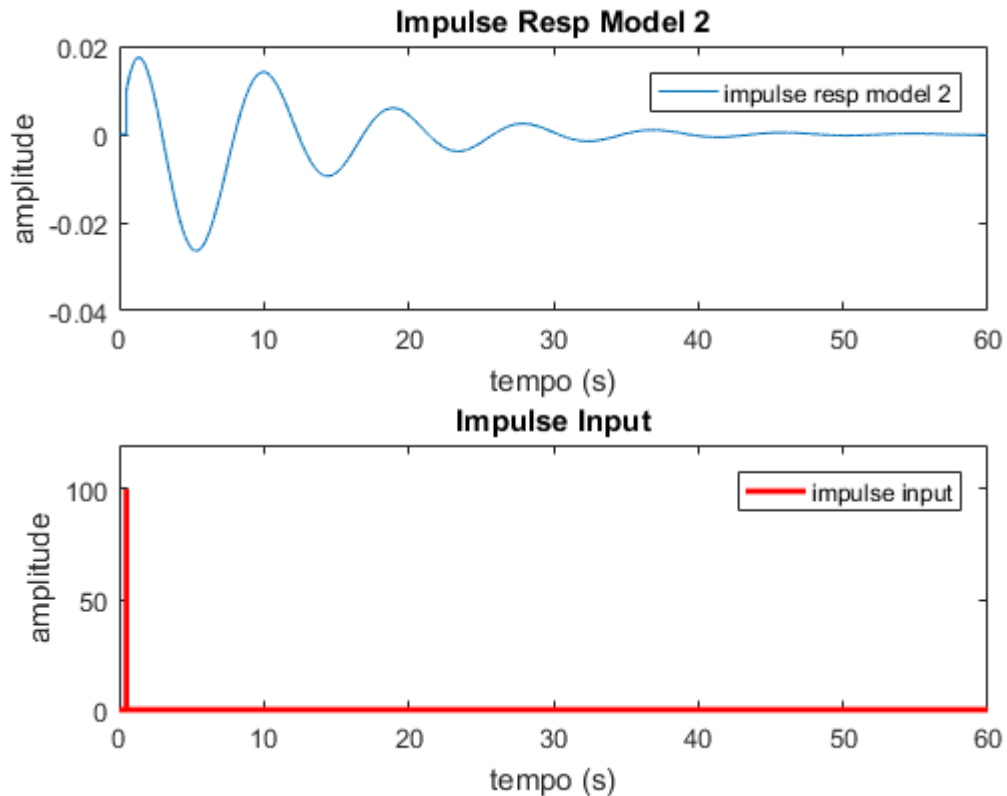
```

figure;
subplot(211);
plot(step_resp_model2);
title('Step Resp Model 2');
xlabel('tempo (s)');
ylabel('amplitude');
legend({'step resp model 2'}, 'Location', 'northeast');
subplot(212);
plot(step_input, '-r', 'Linewidth', 1.5);
axis([0 t_sim -1 1.5]);
title('Step Input');
xlabel('tempo (s)');
ylabel('amplitude');
legend({'step input'}, 'Location', 'northeast');

figure;
subplot(211);
plot(impulse_resp_model2);
title('Impulse Resp Model 2');
xlabel('tempo (s)');
ylabel('amplitude');
legend({'impulse resp model 2'}, 'Location', 'northeast');
subplot(212);
plot(impulse_input, '-r', 'Linewidth', 2);
axis([0 t_sim 0 120]);
title('Impulse Input');
xlabel('tempo (s)');
ylabel('amplitude');
legend({'impulse input'}, 'Location', 'northeast');

```





Obs.: Podemos notar que o sistema converge sendo estável em malha aberta como verificado no exercício anterior.

iv) Projetar observador de estados de ordem completa com polos à sua escolha

Segundo Ogata podemos utilizar o método de obtenção do ganho do observador por meio da equação da dinâmica do erro: $\dot{e} = (A - K_e C)e$ Tendo como resultado para o erro tendendo a 0 a equação: $sI - A + K_e C = 0$ Onde: $K_e = [[K_{e1}]; [K_{e2}]; [K_{e3}]]$ Calculando o determinante temos:

$$\det(sI - A + K_e C) = [(s * s * (s + 0.7)) + (-1 * (k_{e2} - 1) * 0.005) + (k_{e1} * 0 * 0.11)] - [(0.005 * s * k_{e1}) + (0.11 * (k_{e2} - 1) * (s)) + ((s + 0.7) * 0 * (-1))]$$

A escolha dos polos foi adotada considerando pouco mais que o dobro do valor do polo mais rápido do sistema, no caso do sistema A o polo era -0.5000, então o valor adotado foi -1.5 o que garante uma dinâmica mais rápida que o sistema, o terceiro polo foi adotado para aproximação para um sistema de segunda ordem sendo alocado 5x mais rápido que o polo escolhido com o valor de -7.5.

Dessa maneira a equação característica desejada é: $s^3 + 10.51 s^2 + 25.33 s + 20.93$ O que resulta nos seguintes ganhos K_e :

$$K_{e1_A1} = 87026;$$

$$K_{e2_A1} = -4185;$$

$$K_{e3_A1} = 9.81;$$

$A_{obs_A1} = A;$
 $B_{obs_A1} = B;$
 $C_{obs_A1} = C;$
 $D_{obs_A1} = D;$
 $Ke_{obs_A1} = [(Ke1_A1);(Ke2_A1);(Ke3_A1)];$

Adotando o mesmo método para o sistema B1:

$$\det(sI - A + Ke * C) = s^3 + s^2 * (0.6 + Ke1) + s * (0.4 * Ke3 + 0.2 * Ke2 + 0.4 * Ke1 + 0.58) - 0.2 * Ke3 - 0.24 * Ke2 + 0.4 * Ke1 + 0.2$$

A escolha dos polos é -1.2 (O polo mais rápido do sistema é -0.4) e o terceiro polo adotado para aproximação para segunda ordem é -6 (5x mais rápido que polo escolhido)

Então a equação característica desejada é:

$$s^3 + 8.78 s^2 + 18.45 s + 10.61$$

O que resulta nos seguintes ganhos Ke:

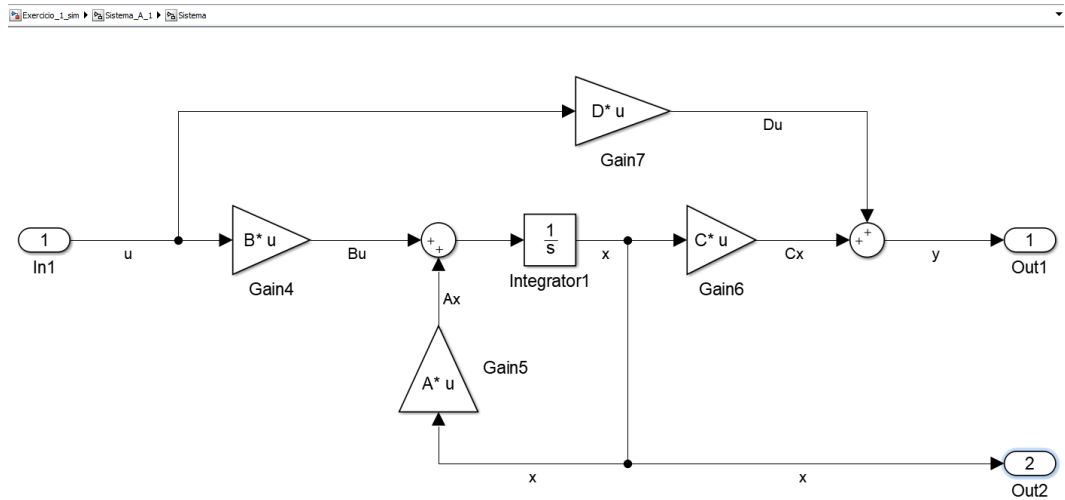
$Ke1_B1 = 8.18;$
 $Ke2_B1 = -42.462;$
 $Ke3_B1 = 15.264;$

$A_{obs_B1} = A2;$
 $B_{obs_B1} = B2;$
 $C_{obs_B1} = C2;$
 $D_{obs_B1} = D2;$
 $Ke_{obs_B1} = [(Ke1_B1);(Ke2_B1);(Ke3_B1)];$

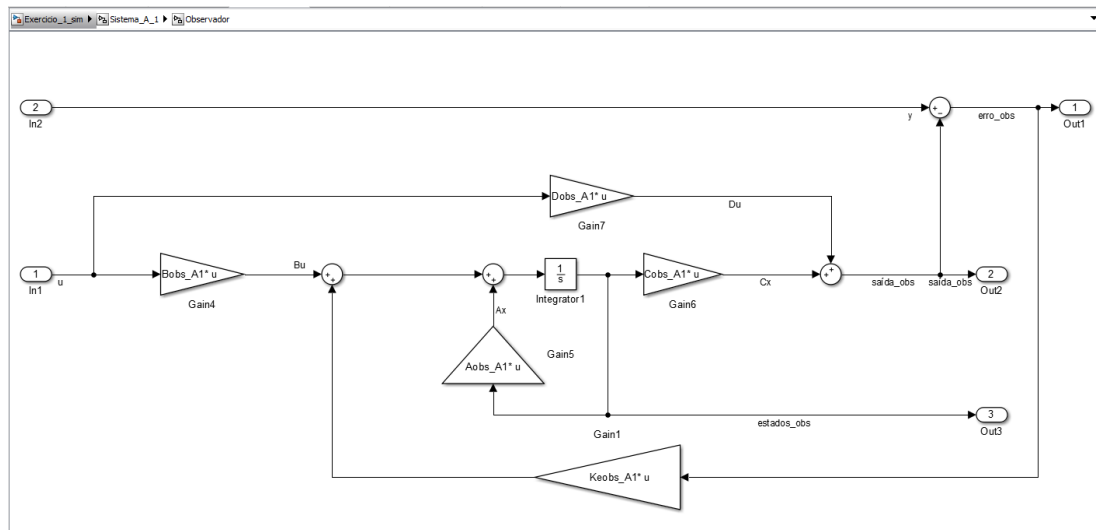
Para ser aproximado para segunda ordem, um dos polos precisa ser pelo menos 5x mais rápido que os polos de segunda ordem desejados, os polos desejados de segunda ordem precisam ser duas vezes mais rápidos que o polo mais rápido do sistema para o observador. Desse modo o polo mais rápido do sistema é -0.5 então os polos desejados do sistema escolhidos foram -1.5 (mais de 2x o polo mais rápido do sistema) O terceiro polo foi alocado 5x mais rápido que o polo escolhido possibilitando aproximar o sistema para segunda ordem. Com o polinômio desejado é possível calcular os ganhos de acordo com a resolução do determinante da matriz resultante do cálculo: $\det(s * A_Id - A + Ke * C)$.

Diagrama de blocos dos sistemas e dos observadores:

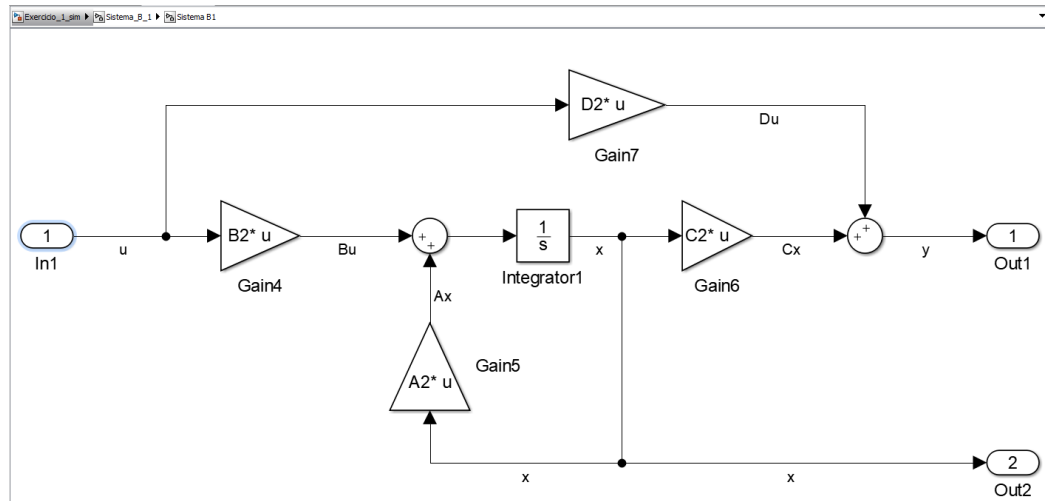
SISTEMA A:



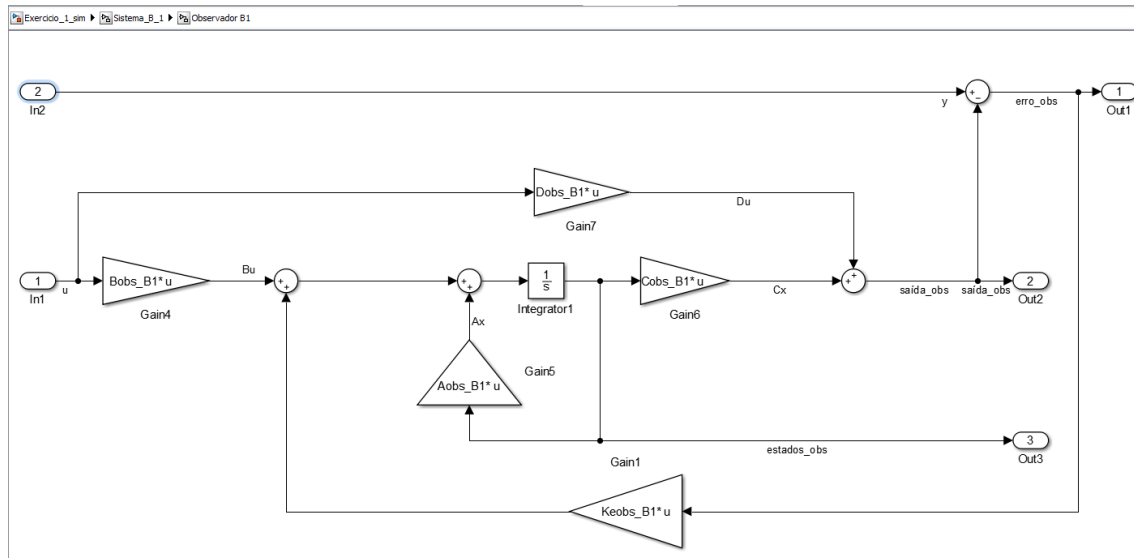
OBSERVADOR DE ORDEM COMPLETA SISTEMA A:



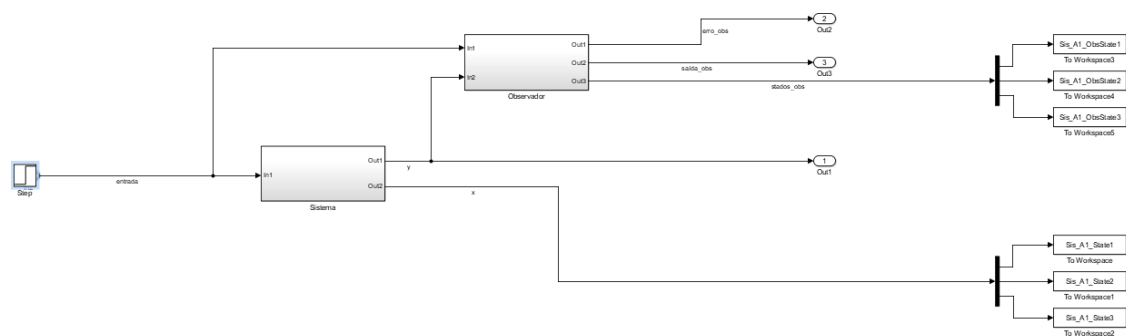
SISTEMA B:



OBSERVADOR DE ORDEM COMPLETA SISTEMA B:



CONFIGURAÇÃO UTILIZADA PARA SIMULAÇÃO DO SISTEMA + OBSERVADOR:



Comparando os resultados obtidos da simulação com condições iniciais diferentes para planta e para o observador:

```
t_sim = 100;
sim('Exercicio_1_sim',t_sim);
%
% Sistema A reposta ao degrau:
figure;
subplot(221);
plot(step_resp_model1,'-black','Linewidth',1);
xlabel('tempo (s)');
ylabel('amplitude');
axis([0 t_sim -0.11415 0.30241]);
hold on
plot(step_resp_saidaobs_model1,'-r','Linewidth',0.7);
title('Modelo A x Observador A');
legend({'step resp model A', 'step obsrv model A'},'Location','southeast');
```

```

hold off
subplot(223);
plot(step_input, '-b', 'Linewidth', 1.5);
axis([0 t_sim -1 1.5]);
title('Step Input');
xlabel('tempo (s)');
ylabel('amplitude');
title('Step Input');
legend({'step input'}, 'Location', 'southeast');

subplot(222); %erro obsv
plot(step_resp_erroobs_model1, '-black', 'Linewidth', 1);
title('Erro Observador A');
xlabel('tempo (s)');
ylabel('amplitude');
legend({'Erro Obsrv A'}, 'Location', 'southeast');

subplot(224); %estados obsv x estados modelo
hold on
plot(Sis_A1_obsState1, '-magenta', 'Linewidth', 1);
plot(Sis_A1_obsState2, '-cyan', 'Linewidth', 1);
plot(Sis_A1_obsState3, '-yellow', 'Linewidth', 1);
plot(Sis_A1_State1, '--black', 'Linewidth', 1);
plot(Sis_A1_State2, '--r', 'Linewidth', 1);
plot(Sis_A1_State3, '--b', 'Linewidth', 1);
title('Estados Model x Obsrv');
xlabel('tempo (s)');
ylabel('amplitude');
legend({'ObsvState1', 'ObsvState2', 'ObsvState3', 'SistState1', 'SistState2',
'SistState3'}, 'Location', 'southeast');
hold off

% Sistema B reposta ao degrau:
figure;
subplot(221);
plot(step_resp_model2, '-black', 'Linewidth', 1);
xlabel('tempo (s)');
ylabel('amplitude');
axis([0 t_sim -0.06272 0.04083]);
hold on
plot(step_resp_saidaobs_model2, '-r', 'Linewidth', 0.7);
title('Modelo B x Observador B');
legend({'step resp model B', 'step obsrv model B'}, 'Location', 'southeast');
hold off
subplot(223);
plot(step_input, '-b', 'Linewidth', 1.5);
axis([0 t_sim -1 1.5]);
title('Step Input');
xlabel('tempo (s)');
ylabel('amplitude');
title('Step Input');
legend({'step input'}, 'Location', 'southeast');

subplot(222); %erro obsv
plot(step_resp_erroobs_model2, '-black', 'Linewidth', 1);
title('Erro Observador B');
xlabel('tempo (s)');

```

```

ylabel('amplitude');
legend({'Erro Obsrv B'}, 'Location', 'southeast');

subplot(224); %Estados obsv x estados modelo
hold on
plot(Sis_B1_ObsState1, '-magenta', 'Linewidth', 1);
plot(Sis_B1_ObsState2, '-cyan', 'Linewidth', 1);
plot(Sis_B1_ObsState3, '-yellow', 'Linewidth', 1);
plot(Sis_B1_State1, '--black', 'Linewidth', 1);
plot(Sis_B1_State2, '--r', 'Linewidth', 1);
plot(Sis_B1_State3, '--b', 'Linewidth', 1);
title('Estados Model x Obsrv');
xlabel('tempo (s)');
ylabel('amplitude');
legend({'ObsvState1', 'ObsvState2', 'ObsvState3', 'SistState1', 'SistState2',
'SistState3'}, 'Location', 'northeast');
hold off

% Sistema A reposta ao impulso:
figure;
subplot(221);
plot(impulse_resp_model1, '-black', 'Linewidth', 1);
xlabel('tempo (s)');
ylabel('amplitude');
axis([0 t_sim -0.11661 0.05402]);
hold on
plot(imp_resp_saidaobs_model1, '-r', 'Linewidth', 0.7);
title('Modelo A x Observador A');
legend({'impulse resp model A', 'impulse obsrv model A'}, 'Location', 'southeast');
hold off
subplot(223);
plot(impulse_input, '-b', 'Linewidth', 1.5);
axis([0 t_sim -10 100]);
title('Impulse Input');
xlabel('tempo (s)');
ylabel('amplitude');
title('Impulse Input');
legend({'impulse input'}, 'Location', 'northeast');

subplot(222); %erro obsv
plot(imp_resp_erroobs_model1, '-black', 'Linewidth', 1);
title('Erro Observador A');
xlabel('tempo (s)');
ylabel('amplitude');
legend({'Erro Obsrv A'}, 'Location', 'southeast');

subplot(224); %Estados obsv x estados modelo
hold on
plot(Sis_A1_imp_ObsState1, '-magenta', 'Linewidth', 1);
plot(Sis_A1_imp_ObsState2, '-cyan', 'Linewidth', 1);
plot(Sis_A1_imp_ObsState3, '-yellow', 'Linewidth', 1);
plot(Sis_A1_imp_State1, '--black', 'Linewidth', 1);
plot(Sis_A1_imp_State2, '--r', 'Linewidth', 1);
plot(Sis_A1_imp_State3, '--b', 'Linewidth', 1);
title('Estados Model x Obsrv');

```

```

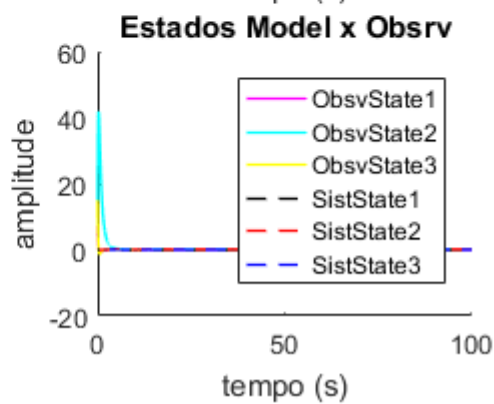
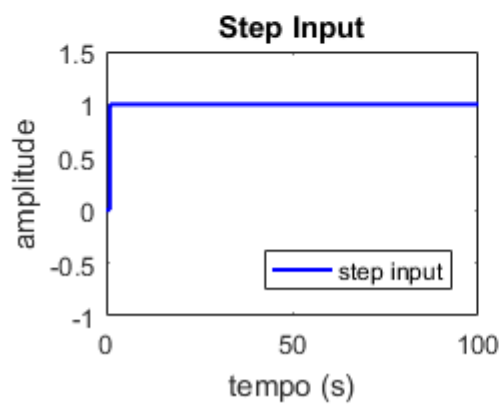
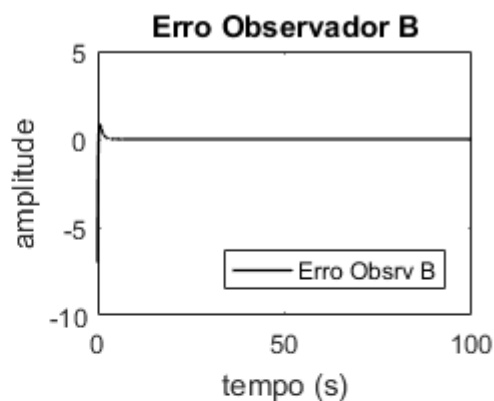
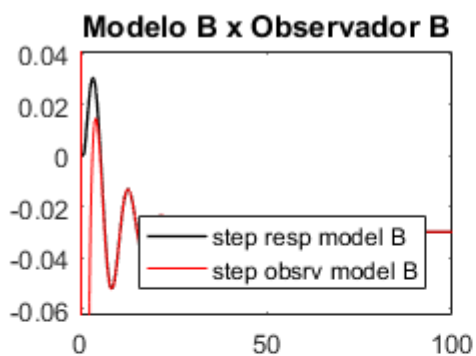
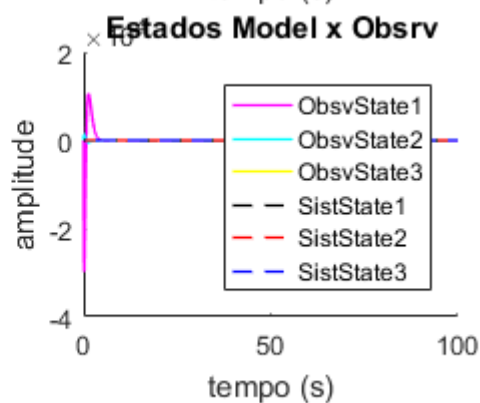
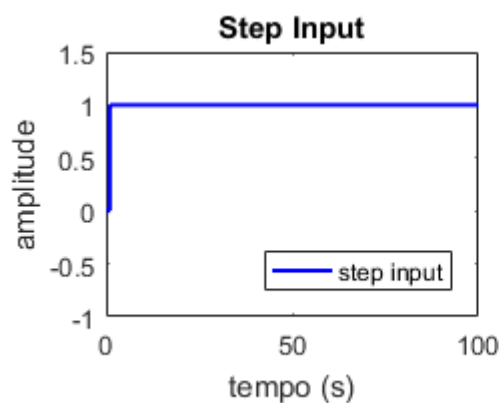
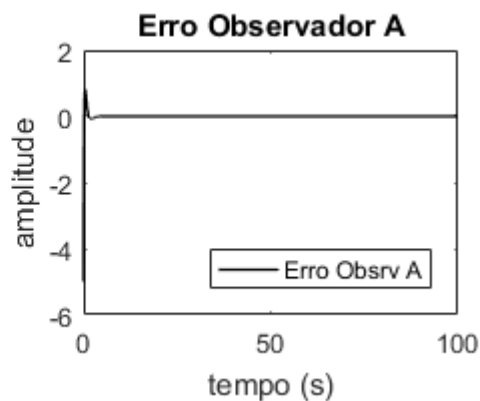
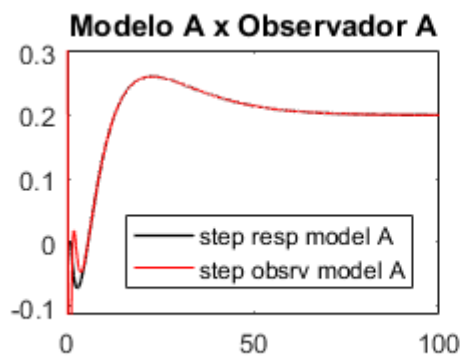
xlabel('tempo (s)');
ylabel('amplitude');
legend({'ObsvState1', 'ObsvState2', 'ObsvState3', 'SistState1', 'SistState2',
'SistState3'}, 'Location', 'southeast');
hold off

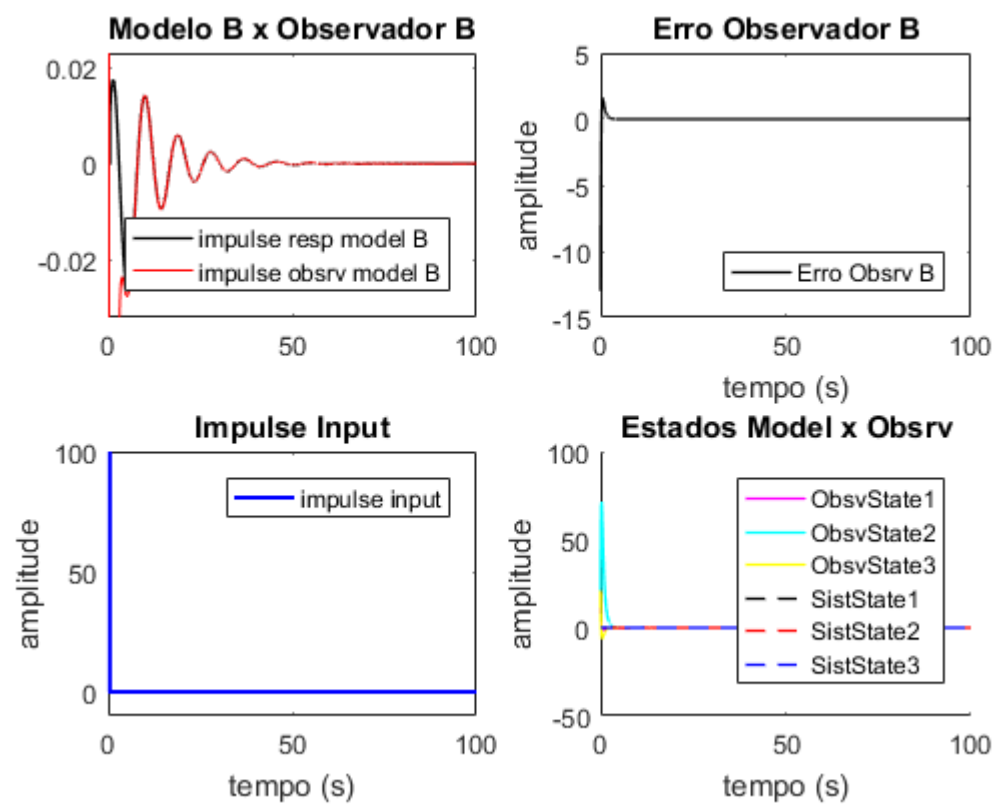
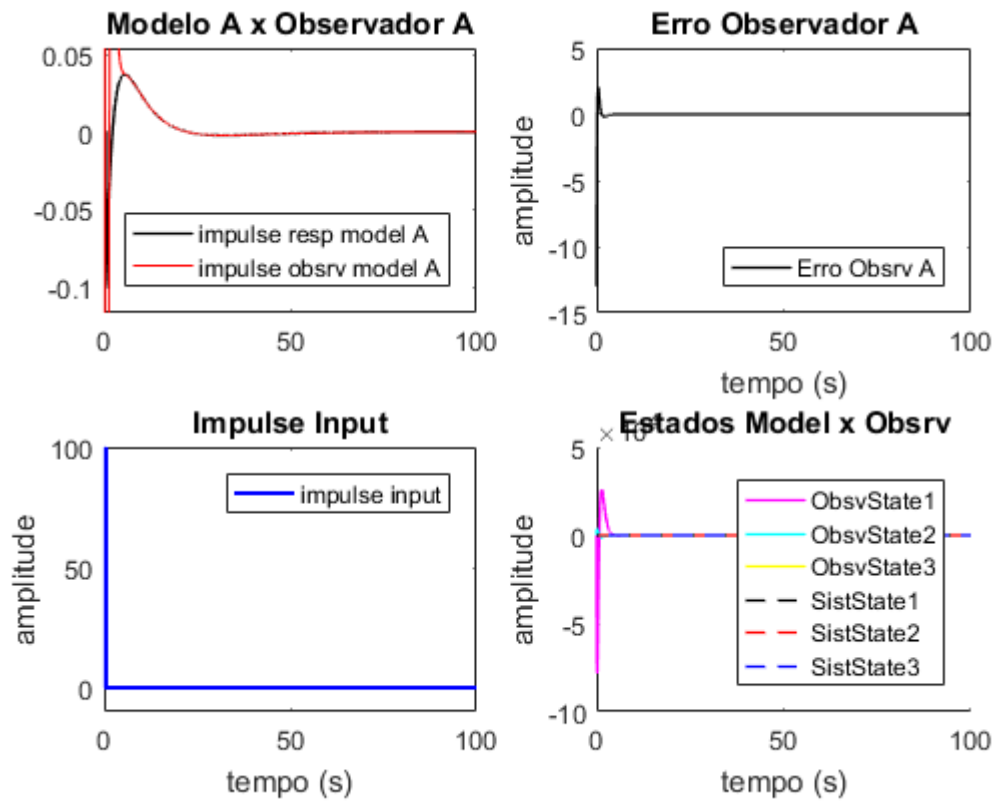
% Sistema B reposta ao impulso:
figure;
subplot(221);
plot(impulse_resp_model2, '-black', 'Linewidth', 1);
xlabel('tempo (s)');
ylabel('amplitude');
axis([0 t_sim -0.0319 0.02292]);
hold on
plot(imp_resp_saidaobs_model2, '-r', 'Linewidth', 0.7);
title('Modelo B x Observador B');
legend({'impulse resp model B', 'impulse obsrv model B'}, 'Location', 'southeast');
hold off
subplot(223);
plot(impulse_input, '-b', 'Linewidth', 1.5);
axis([0 t_sim -10 100]);
title('Impulse Input');
xlabel('tempo (s)');
ylabel('amplitude');
title('Impulse Input');
legend({'impulse input'}, 'Location', 'northeast');

subplot(222); %erro obsv
plot(imp_resp_erroobs_model2, '-black', 'Linewidth', 1);
title('Erro Observador B');
xlabel('tempo (s)');
ylabel('amplitude');
legend({'Erro Obsrv B'}, 'Location', 'southeast');

subplot(224); %Estados obsv x estados modelo
hold on
plot(Sis_B1_imp_ObsState1, '-magenta', 'Linewidth', 1);
plot(Sis_B1_imp_ObsState2, '-cyan', 'Linewidth', 1);
plot(Sis_B1_imp_ObsState3, '-yellow', 'Linewidth', 1);
plot(Sis_B1_imp_State1, '--black', 'Linewidth', 1);
plot(Sis_B1_imp_State2, '--r', 'Linewidth', 1);
plot(Sis_B1_imp_State3, '--b', 'Linewidth', 1);
title('Estados Model x Obsrv');
xlabel('tempo (s)');
ylabel('amplitude');
legend({'ObsvState1', 'ObsvState2', 'ObsvState3', 'SistState1', 'SistState2',
'SistState3'}, 'Location', 'northeast');
hold off

```





v) Refazer o mesmo com observadores de Luenberger (Ordem reduzida)

Para o projeto do observador de ordem reduzida temos para ambos os sistemas um estado relacionado a saída pela matriz C (x3 no caso do sistema A e x1 no caso do sistema B). Dessa maneira podemos reduzir a ordem do observador para estimar apenas os outros dois estados.

Iniciamos reordenando as matrizes para separar a parte medida e a parte estimada. Para o sistema A temos o estado x3 medido e os demais estimados, reordenando as matrizes chegamos nos seguintes valores:

```
A_Aaa = [-0.7];
A_Aab = [-0.005, -0.11];
A_Aba = [(0);(1)];
A_Abb = [0,1;0,0];

A_Ba = [-0.1];
A_Bb = [0.1;-0.2];

A_Ca = [1];
A_Cb = [0,0];

% A equação característica do observador para o sistema A será então:
pr=[-1.5 -1.5];
eqco = poly(pr);
% Com isso os polos desejados são -1.5, resultando na seguinte equação
% desejada:  $s^2+3.00s+2.25$ 
% Calculando os valores de Ke:
A_Ke = acker(A_Abb',A_Aab',pr);
A_Ke1 = A_Ke(1);
A_Ke2 = A_Ke(2);
A_Ke = [A_Ke1;A_Ke2]

% Seguindo a equação apresentada por Ogata para isolar a saída dos estados temos:
A_AObs = (A_Abb-(A_Ke*A_Aab));
A_BObs = ((A_AObs*A_Ke) + A_Aba - (A_Ke*A_Aaa));
A_FObs = (A_Bb - (A_Ke*A_Ba));
A_CObs = [[0 0];[eye(2)]];
A_DObs = [1;A_Ke];

% Para o sistema B temos o estado x1 medido e os demais estimados,
% reordenando as matrizes chegamos nos seguintes valores:
B_Aaa = [-0.2];
B_Aab = [-0.2, 0.4];
B_Aba = [(0.5);(0)];
B_Abb = [0,1;-0.4,-0.4];

B_Ba = [0.01];
B_Bb = [0;0.05];

B_Ca = [1];
B_Cb = [0,0];

BreordA = [B_Aaa, B_Aab; B_Aba, B_Abb]

% A equação característica do observador para o sistema B será então:
```



```

B_pr=[-1.2 -1.2];
eqco = poly(B_pr);
% Nesse caso o observador será de fato de ordem
% 2. Com isso os polos desejados são -1.2, resultando na seguinte equação
% desejada:  $s^2+2.4s+1.44$ 

% Calculando os valores de Ke:
B_Ke = acker(B_Abb',B_Aab',B_pr)
B_Ke1 = B_Ke(1);
B_Ke2 = B_Ke(2);
B_Ke = [B_Ke1;B_Ke2]
% Seguindo a equação apresentada por Ogata para isolar a saída dos estados temos:
B_AObs = (B_Abb-(B_Ke*B_Aab))
B_BObs = ((B_AObs*B_Ke) + B_Aba - (B_Ke*B_Aaa))
B_FObs = (B_Bb - (B_Ke*B_Ba))
B_CObs = [[0 0];[eye(2)]]
B_DObs = [1;B_Ke]

t_sim = 100;
sim('Exercicio_1e_sim',t_sim);

% Sistema A reposta ao degrau Observador de ordem reduzida:
% Considerado condições iniciais aleatórias para o observador
figure;
subplot(212);
plot(OrdMin_step_input,'-b','Linewidth',1.5);
axis([0 t_sim -1 1.5]);
title('Step Input');
xlabel('tempo (s)');
ylabel('amplitude');
title('Step Input');
legend({'A step input'},'Location','southeast');

subplot(211); %estados obsv x estados modelo
hold on
plot(Sis_A1_OrdMin_ObsState1,'-magenta','Linewidth',1);
plot(Sis_A1_OrdMin_ObsState2,'-cyan','Linewidth',1);
plot(Sis_A1_OrdMin_ObsState3,'-yellow','Linewidth',1);
plot(Sis_A1_OrdMin_State1,'--black','Linewidth',1);
plot(Sis_A1_OrdMin_State2,'--r','Linewidth',1);
plot(Sis_A1_OrdMin_State3,'--b','Linewidth',1);
axis([0 t_sim -60 10]);
title('A Estados Model x Obsrv');
xlabel('tempo (s)');
ylabel('amplitude');
legend({'ObsvState1', 'ObsvState2', 'ObsvState3', 'SistState1', 'SistState2',
'SistState3'},'Location','southeast');
hold off

% Sistema A reposta ao impulso Observador de ordem reduzida:
% Considerado condições iniciais aleatórias para o observador
figure;
subplot(212);
plot(OrdMin_impulse_input,'-b','Linewidth',1.5);
axis([0 t_sim -10 100]);
title('A Impulse Input');
xlabel('tempo (s)');

```

```

ylabel('amplitude');
title('Impulse Input');
legend({'impulse input'}, 'Location', 'northeast');

subplot(211); %Estados obsv x estados modelo
hold on
plot(Sis_A1_imp_OrdMin_ObsState1, '-magenta', 'Linewidth', 1);
plot(Sis_A1_imp_OrdMin_ObsState2, '-cyan', 'Linewidth', 1);
plot(Sis_A1_imp_OrdMin_ObsState3, '-yellow', 'Linewidth', 1);
plot(Sis_A1_imp_OrdMin_State1, '--black', 'Linewidth', 1);
plot(Sis_A1_imp_OrdMin_State2, '--r', 'Linewidth', 1);
plot(Sis_A1_imp_OrdMin_State3, '--b', 'Linewidth', 1);
axis([0 t_sim -2 0.5]);
title('A Estados Model x Obsrv');
xlabel('tempo (s)');
ylabel('amplitude');
legend({'ObsvState1', 'ObsvState2', 'ObsvState3', 'SistState1', 'SistState2',
'SistState3'}, 'Location', 'southeast');
hold off

% Sistema B reposta ao degrau Observador de ordem reduzida:
% Considerado condições iniciais aleatórias para o observador
figure;
subplot(212);
plot(OrdMin_step_input, '-b', 'Linewidth', 1.5);
axis([0 t_sim -1 1.5]);
title('Step Input');
xlabel('tempo (s)');
ylabel('amplitude');
title('B Step Input');
legend({'step input'}, 'Location', 'southeast');

subplot(211); %estados obsv x estados modelo
hold on
plot(Sis_B1_OrdMin_ObsState1, '-magenta', 'Linewidth', 1);
plot(Sis_B1_OrdMin_ObsState2, '-cyan', 'Linewidth', 1);
plot(Sis_B1_OrdMin_ObsState3, '-yellow', 'Linewidth', 1);
plot(Sis_B1_OrdMin_State1, '--black', 'Linewidth', 1);
plot(Sis_B1_OrdMin_State2, '--r', 'Linewidth', 1);
plot(Sis_B1_OrdMin_State3, '--b', 'Linewidth', 1);
axis([0 t_sim -0.08144 0.25]);
title('B Estados Model x Obsrv');
xlabel('tempo (s)');
ylabel('amplitude');
legend({'ObsvState1', 'ObsvState2', 'ObsvState3', 'SistState1', 'SistState2',
'SistState3'}, 'Location', 'southeast');
hold off

% Sistema B reposta ao impulso Observador de ordem reduzida:
% Considerado condições iniciais aleatórias para o observador
figure;
subplot(212);
plot(OrdMin_impulse_input, '-b', 'Linewidth', 1.5);
axis([0 t_sim -10 100]);
xlabel('tempo (s)');
ylabel('amplitude');

```

```

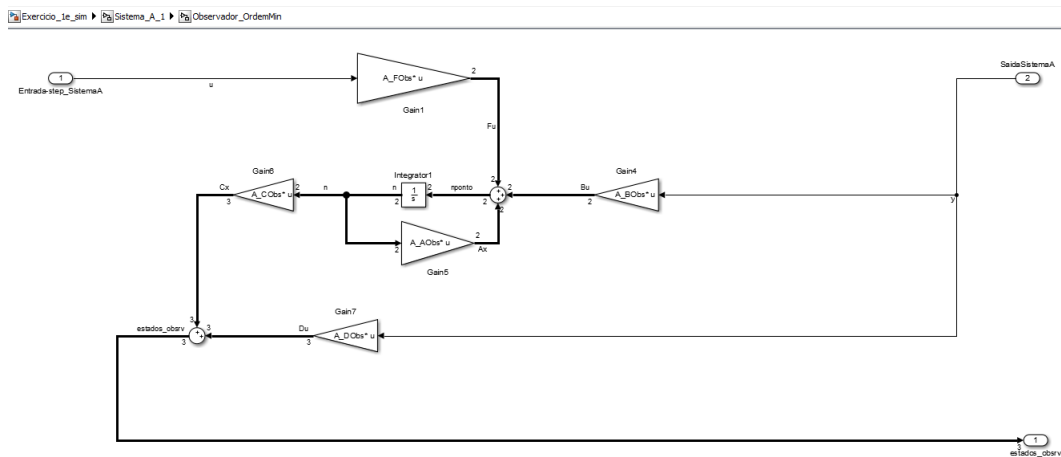
title('B Impulse Input');
legend({'impulse input'}, 'Location', 'northeast');

subplot(211); %Estados obsv x estados modelo
hold on
plot(Sis_B1_imp_OrdMin_ObsState1, '-magenta', 'Linewidth', 1);
plot(Sis_B1_imp_OrdMin_ObsState2, '-cyan', 'Linewidth', 1);
plot(Sis_B1_imp_OrdMin_ObsState3, '-yellow', 'Linewidth', 1);
plot(Sis_B1_imp_OrdMin_State1, '--black', 'Linewidth', 1);
plot(Sis_B1_imp_OrdMin_State2, '--r', 'Linewidth', 1);
plot(Sis_B1_imp_OrdMin_State3, '--b', 'Linewidth', 1);
axis([0 t_sim -0.05347 0.085]);
title('B Estados Model x Obsrv');
xlabel('tempo (s)');
ylabel('amplitude');
legend({'ObsvState1', 'ObsvState2', 'ObsvState3', 'SistState1', 'SistState2',
'SistState3'}, 'Location', 'southeast');
hold off

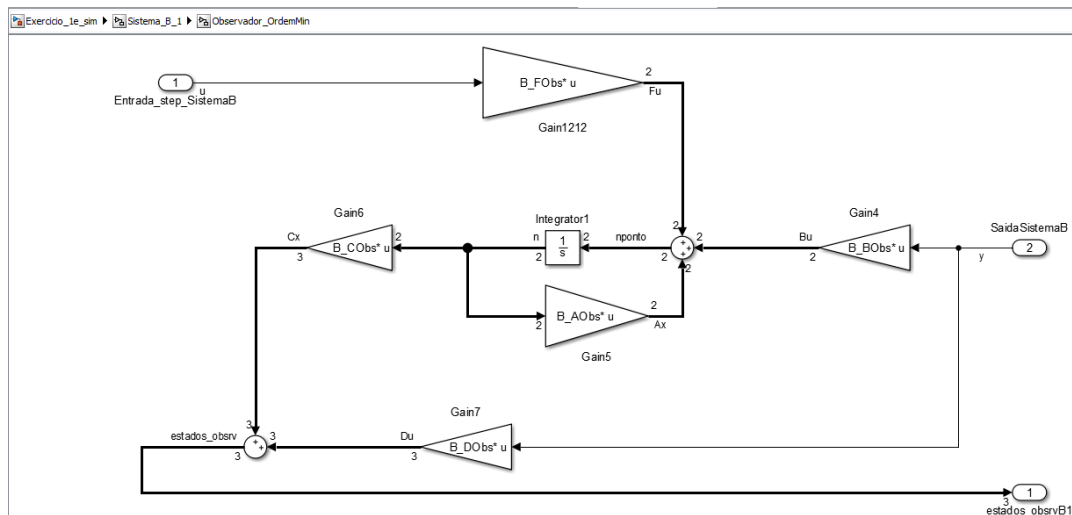
```

Diagrama de blocos dos sistemas e dos observadores:

OBSERVADOR DE ORDEM REDUZIDA SISTEMA A:

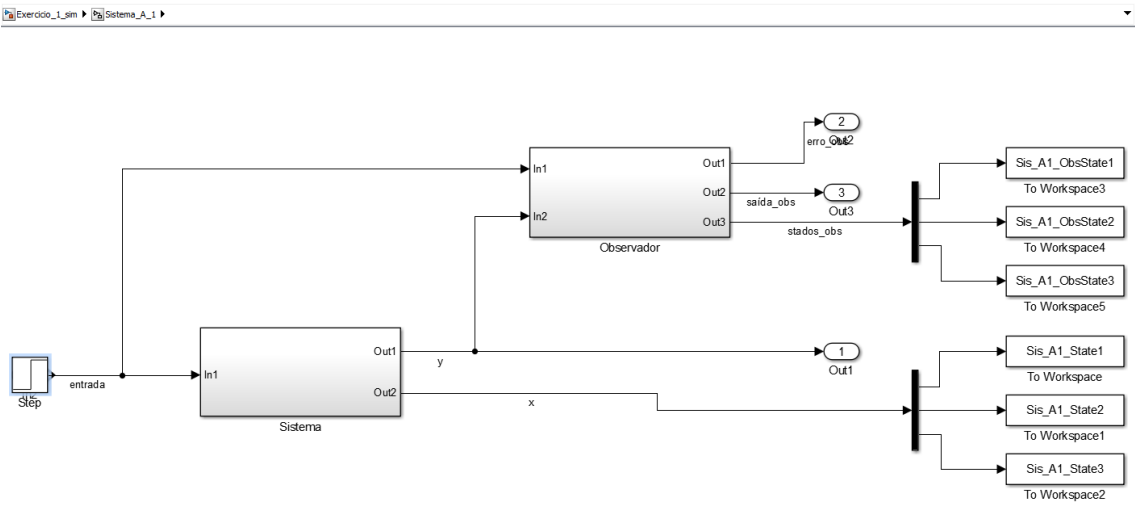


OBSERVADOR DE ORDEM REDUZIDA SISTEMA B:

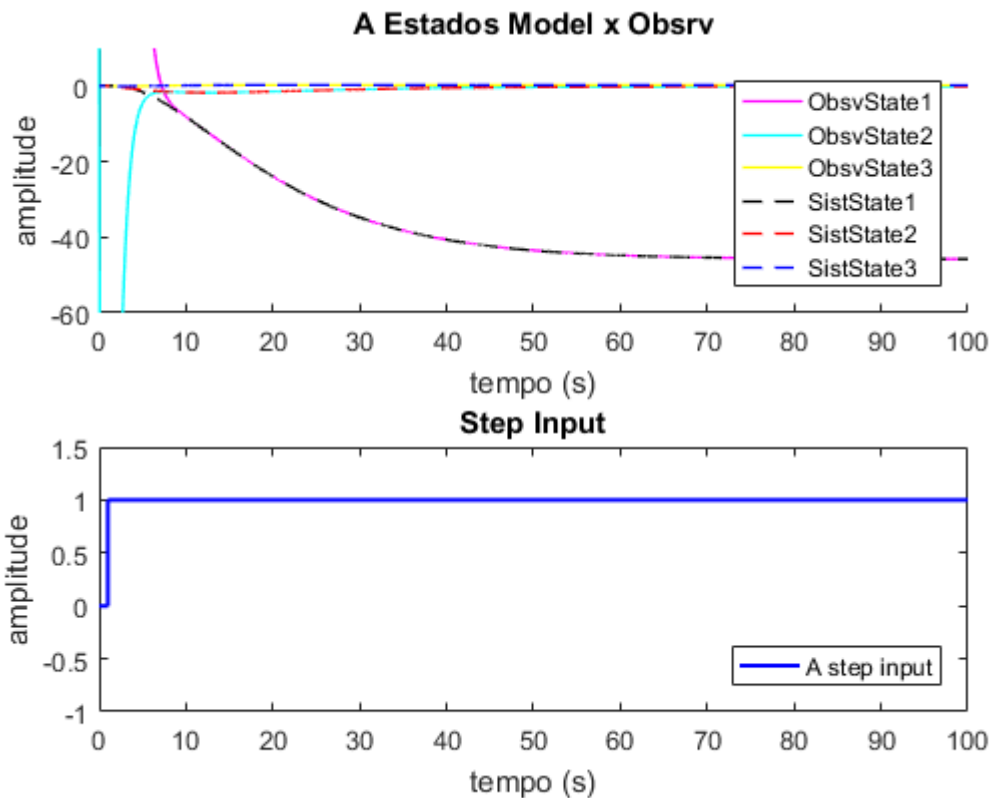


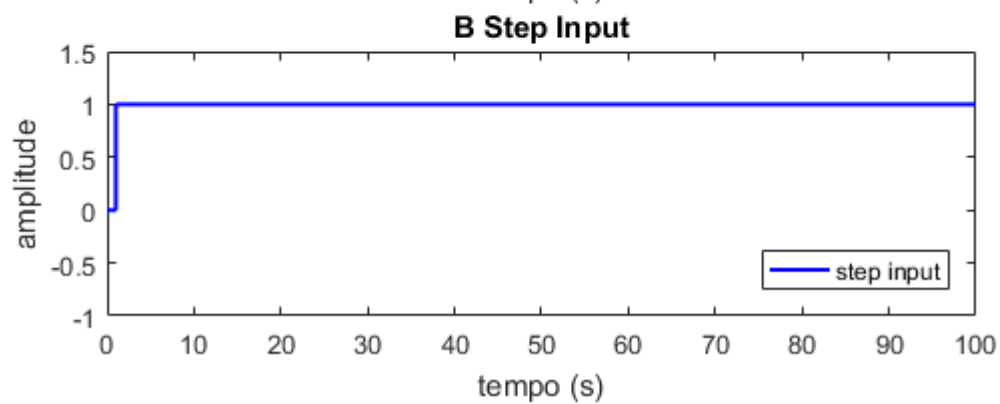
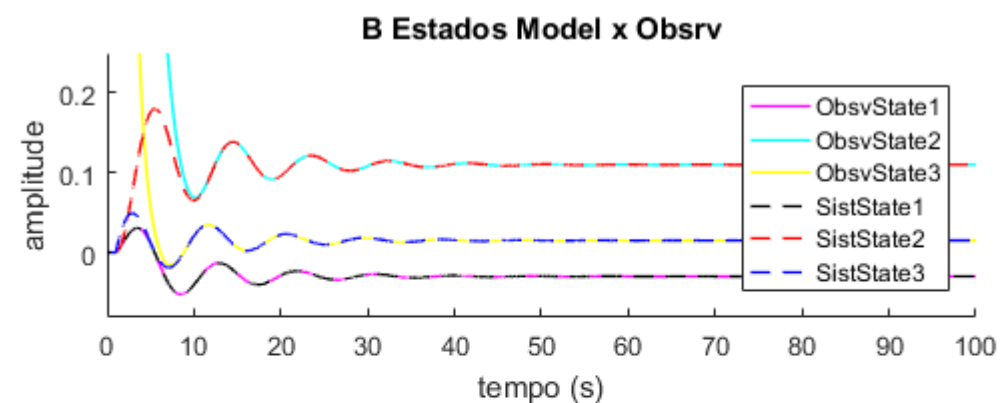
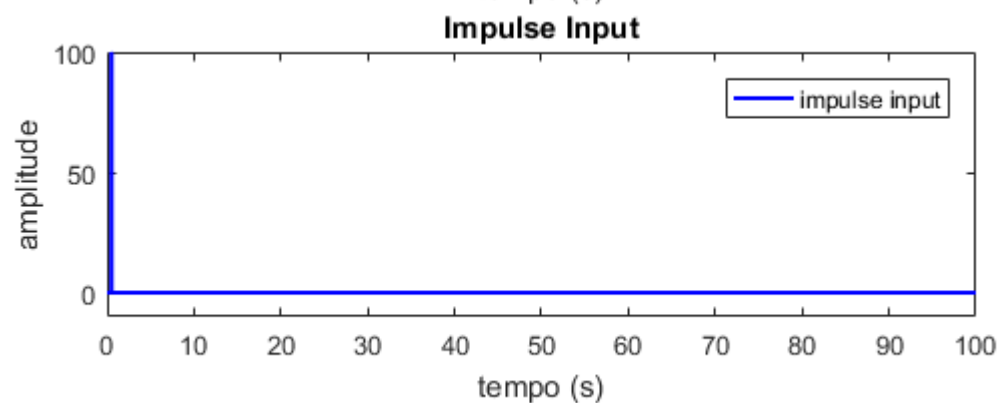
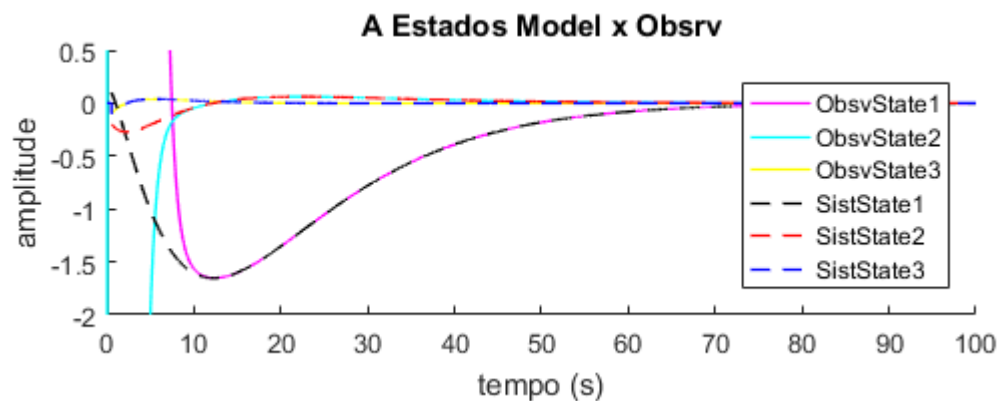
CONFIGURAÇÃO UTILIZADA PARA SIMULAÇÃO DO SISTEMA + OBSERVADOR:

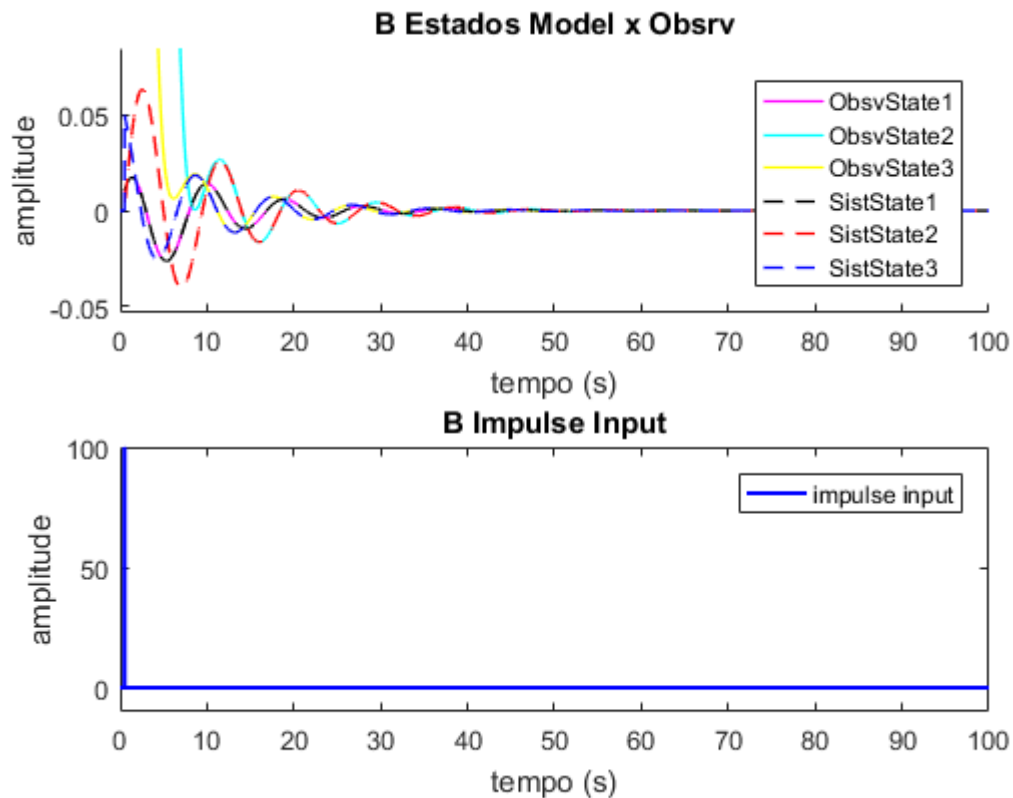
RESULTADO



RESULTADOS DA SIMULAÇÃO:







vi) Projetar controlador para tornar a resposta ao degrau mais rápida em malha fechada utilizando o método de alocação de polos e observador a sua escolha, avaliar sensibilidade da planta a escolha dos polos do controlador e observador

```
Para o sistema A: Escolha dos polos de MF do sistema:
mp = 0.15; %sobressinal
zeta = sqrt((log(mp)/pi)^2/((log(mp)/pi)^2+1)); %coeficiente de amortecimento
t_assent = 15; %tempo de assentamento
wn = 4/(zeta*t_assent);
%zeta = 0.5169 wn = 0.5159
s1 = -zeta*wn + j*wn*sqrt(1-zeta^2);
s2 = -zeta*wn - j*wn*sqrt(1-zeta^2);
% Polo 5x mais rápido para aproximação para segunda ordem:
s3 = - 1.35;
p_mf = [s1, s2, s3];
A_K_cntrl = acker(A,B, p_mf); %Ganhos do controlador para os polos desados para o
sistema A
% A_K_cntrl = -1.5685 -3.3935 -6.6148

% Para o sistema B:
% Escolha dos polos de MF do sistema:
mpB = 0.20; %sobressinal
zetaB = sqrt((log(mpB)/pi)^2/((log(mpB)/pi)^2+1)); %coeficiente de amortecimento
t_assentB = 20; %tempo de assentamento
```

```

wnB = 4/(zetaB*t_assentB);
    %zeta = 0.4559   wn = 0.4386
s1B = -zetaB*wnB + j*wnB*sqrt(1-zetaB^2);
s2B = -zetaB*wnB - j*wnB*sqrt(1-zetaB^2);

% Polo 5x mais rápido para aproximação para segunda ordem:
s3B = - 1.2;
p_mfB = [s1, s2, s3];
B_K_cntrl = acker(A2,B2, p_mfB); %Ganhos do controlador para os polos desejados para o
sistema A
% B_K_cntrl = -1.4596    3.3011    25.9586

% Adição do termo N para corrigir o erro em regime:
N_termo_A1f = rscale(AreordA,AreordB,AreordC,D,A_K_cntrl)
N_termo_B1f = rscale(Breord,B2,C2,D2,B_K_cntrl)

t_sim = 100;
sim('Exercicio_1f_sim',t_sim);

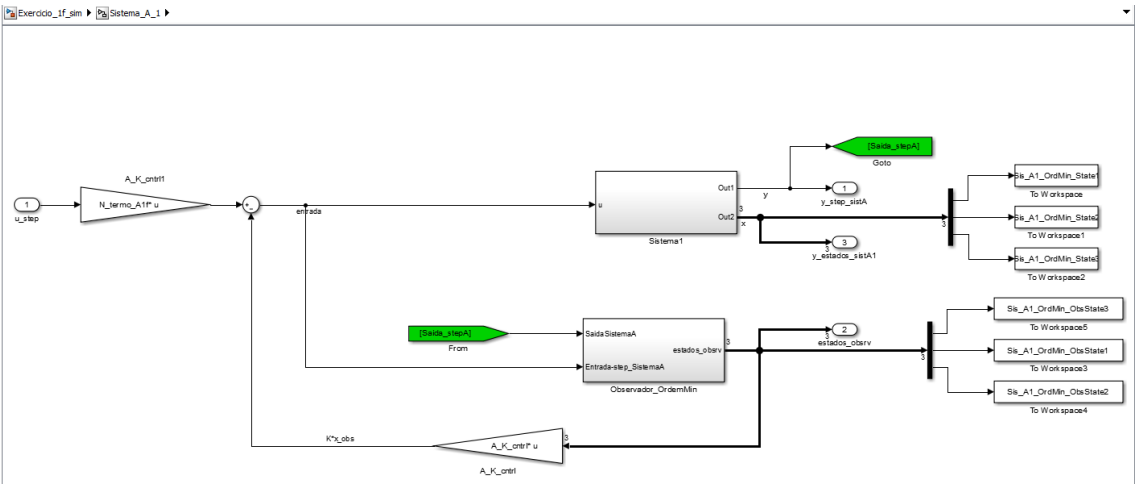
% Resposta ao degrau melhorada do sistema A
figure;
subplot(211);
plot(OrdMinCntrl_step_resp_model1,'-black','Linewidth',1);
xlabel('tempo (s)');
ylabel('amplitude');
axis([0 t_sim -28.28636 24]);
title('SISTEMA A');
legend({'step resp model A'},'Location','southeast');
subplot(212);
plot(OrdMinCntrl_step_input,'-b','Linewidth',1.5);
axis([0 t_sim -1 1.5]);
title('Resposta ao degrau com controle');
xlabel('tempo (s)');
ylabel('amplitude');
title('Step Input');
legend({'step input'},'Location','southeast');

% Resposta ao degrau melhorada do sistema B
figure;
subplot(211);
plot(OrdMinCntrl_step_resp_model2,'-black','Linewidth',1);
xlabel('tempo (s)');
ylabel('amplitude');
axis([0 t_sim -1.78414 2]);
title('SISTEMA B');
legend({'step resp model B'},'Location','southeast');
subplot(212);
plot(OrdMinCntrl_step_input,'-b','Linewidth',1.5);
axis([0 t_sim -1 1.5]);
title('Resposta ao degrau com controle');
xlabel('tempo (s)');
ylabel('amplitude');
title('Step Input');
legend({'step input'},'Location','southeast');

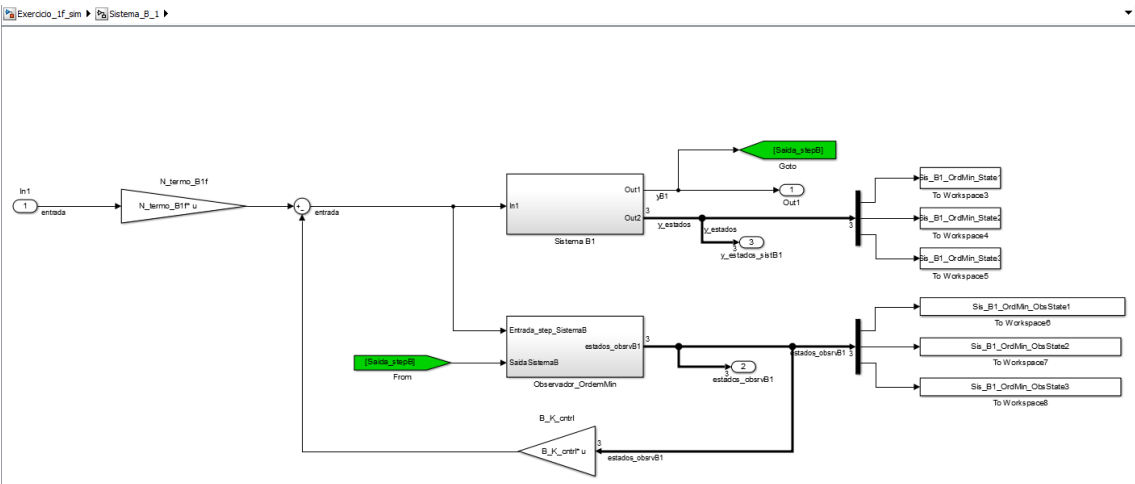
```

Configuração utilizada para simulação do sistema com controle por realimentação dos estados observados com observador de ordem reduzida:

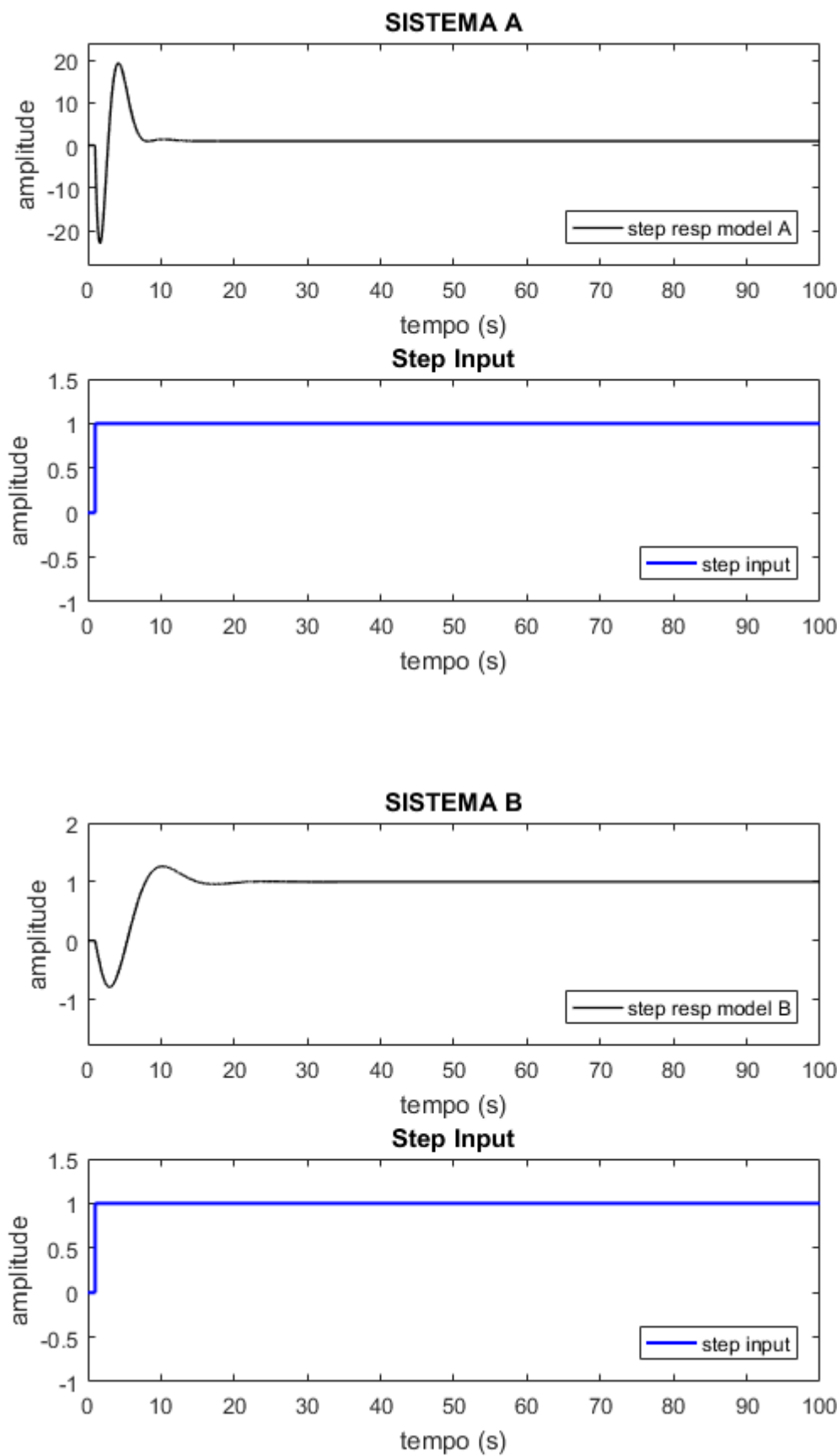
SISTEMA A



SISTEMA B



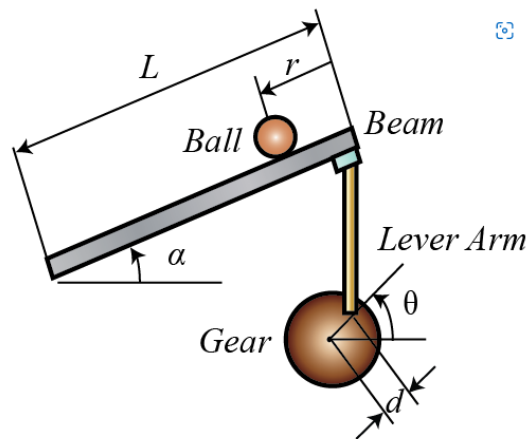
Resultados obtidos após o controle aplicado:



Exercício 2:

Sistema escolhido representado em espaço de estados:

O sistema escolhido é um sistema de suspensão e tem como base um exemplo genérico de modelo para ball & beam:



Fonte: [Control Tutorials for MATLAB and Simulink - Ball & Beam: System Modeling \(umich.edu\)](http://control.tutorialsforMATLABandSimulink.com/Ball%20Beam/System%20Modeling/umich.edu)

Com os seguintes parâmetros adotados:

(m)	mass of the ball	0.11 kg
(R)	radius of the ball	0.015 m
(d)	lever arm offset	0.03 m
(g)	gravitational acceleration	9.8 m/s ²
(L)	length of the beam	1.0 m
(J)	ball's moment of inertia	9.99e-6 kg.m ²
(r)	ball position coordinate	
(alpha)	beam angle coordinate	
(theta)	servo gear angle	

Dessa maneira temos o sistema representado em espaço de estados:

$m = 0.111;$

$R = 0.015;$

$g = -9.8;$

$J = 9.99e-6;$

$H_{ball} = -m \cdot g / (J / (R^2) + m);$

$A_{ball} = [0 \ 1 \ 0 \ 0$

$0 \ 0 \ H_{ball} \ 0$

$0 \ 0 \ 0 \ 1$

$0 \ 0 \ 0 \ 0];$

$B_{ball} = [0;0;0;1];$

$C_{ball} = [1 \ 0 \ 0 \ 0];$

$D_{ball} = [0];$

$ball_{ss} = ss(A_{ball}, B_{ball}, C_{ball}, D_{ball});$

$ball_{ss} =$

$A =$

	x1	x2	x3	x4
x1	0	1	0	0
x2	0	0	7	0
x3	0	0	0	1
x4	0	0	0	0

$B =$

	u1
x1	0
x2	0
x3	0
x4	1

$C =$

	x1	x2	x3	x4
y1	1	0	0	0

D =

	u1
y1	0

Continuous-time state-space model.

```
% Verificando a observabilidade e controlabilidade:
Matriz_Cntrl_ball = ctrb(A_ball,B_ball);
% Verificando o posto (rank) da matrizes de controlabilidade
Posto_Matriz_Cntrl_ball = rank(Matriz_Cntrl_ball);

% Matrizes de observabilidade:
Matriz_Observ_ball = obsv(A_ball,C_ball);
% Verificando o posto (rank) das matrizes de observabilidade
Posto_Matriz_Obsv_ball = rank(Matriz_Observ_ball);

% Ambos os ranks das matrizes de observabilidade e controlabilidade são 4,
% o mesmo número da ordem do sistema o que indica que o sistema é controlável e
observável.

% Estabilidade em Malha aberta:
Estabilidade_Sistema_ball = eig(A_ball);
% Todos os polos do sistema estão na origem indicando que o
% sistema é instável em malha aberta.
% Para a realimentação temos o seguinte K calculado tendo em vista os polos desejados de
MF:
p1_ball = -2+2i;
p2_ball = -2-2i;
p3_ball = -20;
p4_ball = -80;

K_ball = place(A_ball,B_ball,[p1_ball,p2_ball,p3_ball,p4_ball])

% Projeto do observador de ordem completa:
AObs_ball = A_ball;
BObs_ball = B_ball;
CObs_ball = C_ball;
DObs_ball = D_ball;

% Polos escolhidos duas vezes mais rápidos que os polos do sistema:
p1_ball = -4.5;
p2_ball = -4.5;
p3_ball = -50;
p4_ball = -170;

pr_obs=[p1_obs p2_obs p3_obs p4_obs];
```

```

eqco_obs = poly(pr_obs);
% Com isso os polos desejados são -4+/-2i, -50 e -100
% Calculando os valores de Ke:
ke_ball = acker(AObs_ball,BObs_ball,pr_obs);
% Ke_ball = 914.3    971.4    2004.0    104.0

```

```

% Projeto do Observador de ordem mínima
% Como x1 é relacionado a saída podemos assumir o observador com uma ordem
% a menos. Organizando as matrizes temos:

```

```

Ball_Aaa = [0];
Ball_Aab = [1, 0, 0];
Ball_Aba = [0; 0; 0];
Ball_Abb = [0,7,0;0,0,1;0,0,0];

```

```

Ball_Ba = [0];
Ball_Bb = [0;0;1];

```

```

Ball_Ca = [1];
Ball_Cb = [0,0,0];

```

```

% A equação característica do observador para o sistema ball será então:

```

```

% Os polos desejados são -4+/-2i, -40

```

```

p1_obs_ordmin = -4+2i;
p2_obs_ordmin = -4-2i;
p3_obs_ordmin = -40;

```

```

pr_obs_ordmin=[p1_obs_ordmin p2_obs_ordmin p3_obs_ordmin];
eqco_obs_ordmin = poly(pr_obs_ordmin);

```

```

% Resultando na seguinte equação desejada: s3+48s2+340s+800

```

```

% Calculando os valores de Ke:

```

```

ke_ball_ordmin = acker(Ball_Abb',Ball_Aab',pr_obs_ordmin);
ke1_ball_ordmin = ke_ball_ordmin(1);
ke2_ball_ordmin = ke_ball_ordmin(2);
ke3_ball_ordmin = ke_ball_ordmin(3);
ke_ball_ordmin = [ke1_ball_ordmin;ke2_ball_ordmin;ke3_ball_ordmin]

```

```

% Seguindo a equação apresentada por Ogata para isolar a saída dos estados temos:

```

```

Ball_Aobs = (Ball_Abb-(ke_ball_ordmin*Ball_Aab));
Ball_Bobs = ((Ball_Aobs*ke_ball_ordmin) + Ball_Aba - (ke_ball_ordmin*Ball_Aaa));
Ball_Fobs = (Ball_Bb - (ke_ball_ordmin*Ball_Ba));
Ball_cobs = [[0 0 0];[eye(3)]];
Ball_dobs = [1;ke_ball_ordmin];

```

```

% Necessário adição do termo N para corrigir o erro em regime permanente:

```

```

N_termo = rscale(A_ball,B_ball,C_ball,D_ball,K_ball);

```

```

% Analisando a resposta obtida:
t_sim = 150;
sim('Exercicio_1gball_sim',t_sim);

% Resposta ao degrau com observador de ordem reduzida do sistema
figure;
subplot(211);
plot(OrdMinCntrl_step_resp_ball,'-black','Linewidth',1);
xlabel('tempo (s)');
ylabel('amplitude');
axis([0 t_sim -0.4 1.2]);
title('Sistema com realimentação por observador de ordem reduzida');
legend({'resposta ao degrau'},'Location','southeast');
subplot(212);
plot(OrdMinCntrl_step_input_ball,'-b','Linewidth',1.5);
axis([0 t_sim -1 1.5]);
title('Resposta ao degrau com controle');
xlabel('tempo (s)');
ylabel('amplitude');
title('Step Input');
legend({'step input'},'Location','southeast');

%Comparação dos estados observados e do sistema
figure;
subplot(212);
plot(OrdMinCntrl_step_input_ball,'-b','Linewidth',1.5);
axis([1 5 -1 1.5]);
title('Step Input');
xlabel('tempo (s)');
ylabel('amplitude');
title('Sist Step Input');
legend({'step input'},'Location','southeast');

subplot(211); %estados obsv ordem min x estados modelo
hold on
plot(Sis_ball_OrdMin_ObsState1,'-magenta','Linewidth',1);
plot(Sis_ball_OrdMin_ObsState2,'-cyan','Linewidth',1);
plot(Sis_ball_OrdMin_ObsState3,'-yellow','Linewidth',1);
plot(Sis_ball_OrdMin_ObsState4,'-g','Linewidth',1);
plot(Sis_ball_OrdMin_State1,'--black','Linewidth',1);
plot(Sis_ball_OrdMin_State2,'--r','Linewidth',1);
plot(Sis_ball_OrdMin_State3,'--b','Linewidth',1);
plot(Sis_ball_OrdMin_State4,'--black','Linewidth',1);
axis([1 5 -10 10]);
title('Estados Sist x Obsrv Ordem Reduzida');
xlabel('tempo (s)');
ylabel('amplitude');
legend({'ObsvState1', 'ObsvState2', 'ObsvState3', 'ObsvState4', 'SistState1',
'SistState2', 'SistState3', 'SistState4'},'Location','southeast');
hold off

% Resposta ao degrau com observador de ordem plena do sistema
figure;
subplot(211);
plot(step_resp_ball,'-black','Linewidth',1);
xlabel('tempo (s)');

```

```

ylabel('amplitude');
axis([0 t_sim -0.4 1.2]);
title('Sistema com realimentação por observador de ordem plena');
legend({'resposta ao degrau'}, 'Location', 'southeast');
subplot(212);
plot(step_input_ball, '-b', 'Linewidth', 1.5);
axis([0 t_sim -1 1.5]);
title('Resposta ao degrau com controle');
xlabel('tempo (s)');
ylabel('amplitude');
title('Step Input');
legend({'step input'}, 'Location', 'southeast');

%Comparação dos estados observados e do sistema
figure;
subplot(212);
plot(step_input_ball, '-b', 'Linewidth', 1.5);
axis([0 t_sim -1 1.5]);
title('Step Input');
xlabel('tempo (s)');
ylabel('amplitude');
title('Sist Step Input');
legend({'step input'}, 'Location', 'southeast');

subplot(211); %estados obsv ordem plena x estados modelo
hold on
plot(Sis_ball_ObsState1, '-magenta', 'Linewidth', 1);
plot(Sis_ball_ObsState2, '-cyan', 'Linewidth', 1);
plot(Sis_ball_ObsState3, '-yellow', 'Linewidth', 1);
plot(Sis_ball_ObsState4, '-g', 'Linewidth', 1);
plot(Sis_ball_State1, '--black', 'Linewidth', 1);
plot(Sis_ball_State2, '--r', 'Linewidth', 1);
plot(Sis_ball_State3, '--b', 'Linewidth', 1);
plot(Sis_ball_State4, '--black', 'Linewidth', 1);
axis([0 t_sim -10 10]);
title('Estados Sist x Obsrv Ordem Reduzida');
xlabel('tempo (s)');
ylabel('amplitude');
legend({'ObsvState1', 'ObsvState2', 'ObsvState3', 'ObsvState4', 'SistState1',
'SistState2', 'SistState3', 'SistState4'}, 'Location', 'southeast');
hold off

```

DIAGRAMA DE BLOCOS DO SISTEMA COM REALIMENTAÇÃO POR OBSERVADOR DE ESTADOS DE ORDEM PLENA E DE ORDEM REDUZIDA:

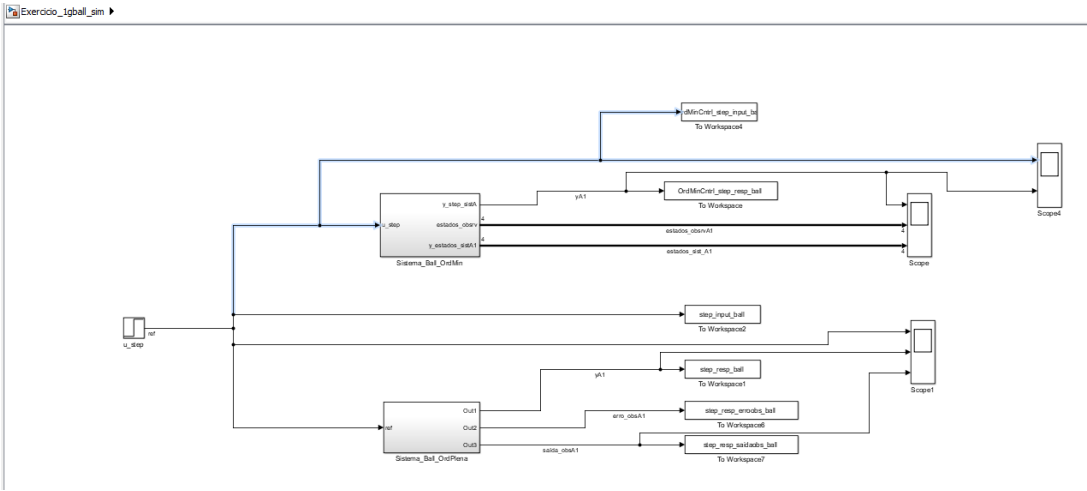


DIAGRAMA DE BLOCOS DO SISTEMA:

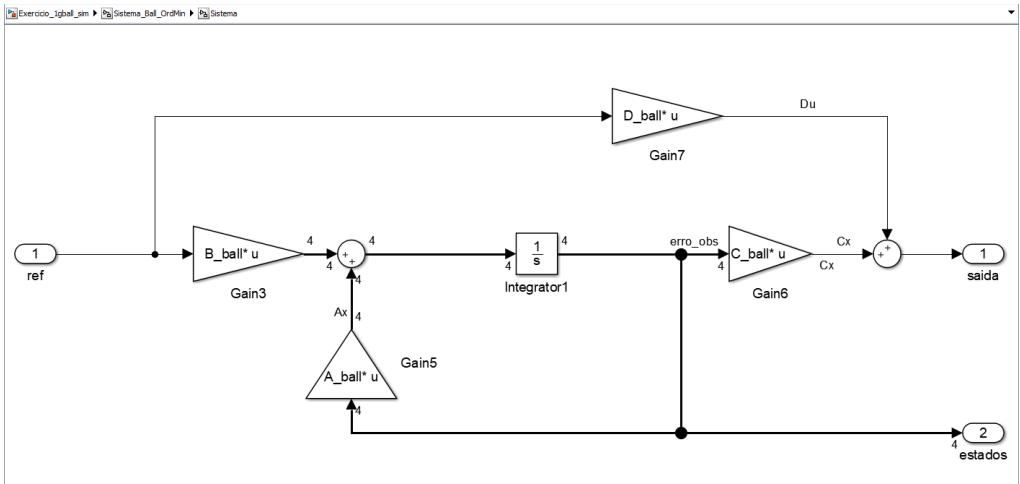


DIAGRAMA DE BLOCOS DO OBSERVADOR DE ORDEM REDUZIDA:

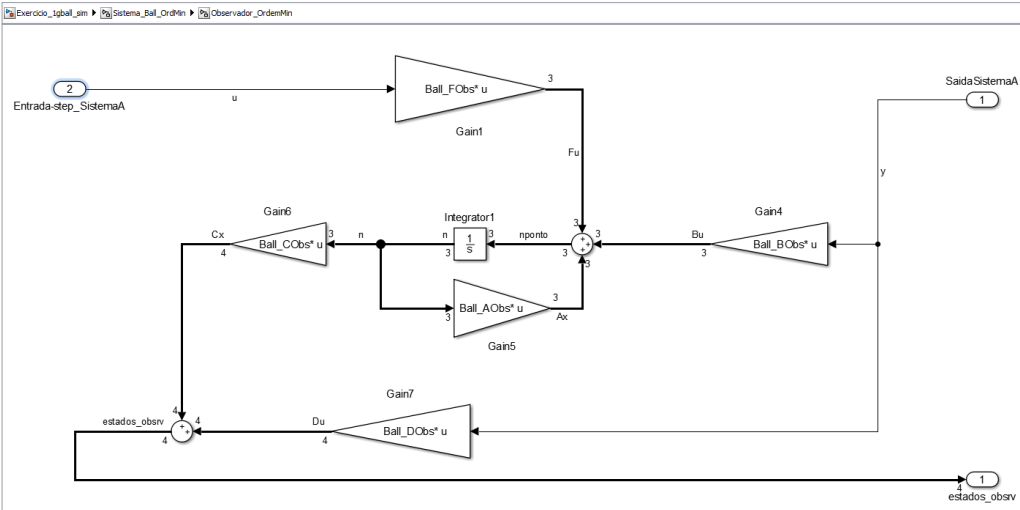


DIAGRAMA DE BLOCOS DO CONTROLE POR REALIMENTAÇÃO DE ESTADOS COM OBSERVADOR DE ORDEM REDUZIDA:

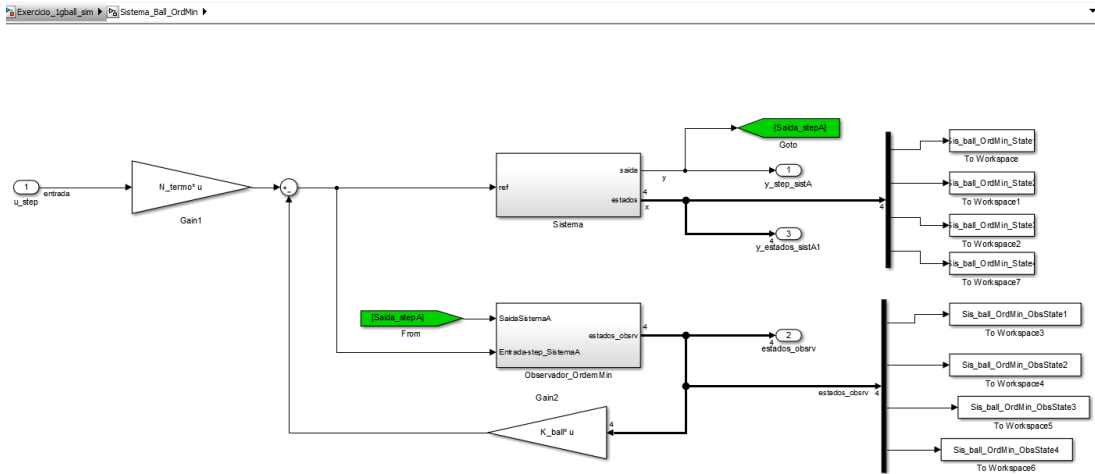


DIAGRAMA DE BLOCOS DO OBSERVADOR DE ORDEM COMPLETA:

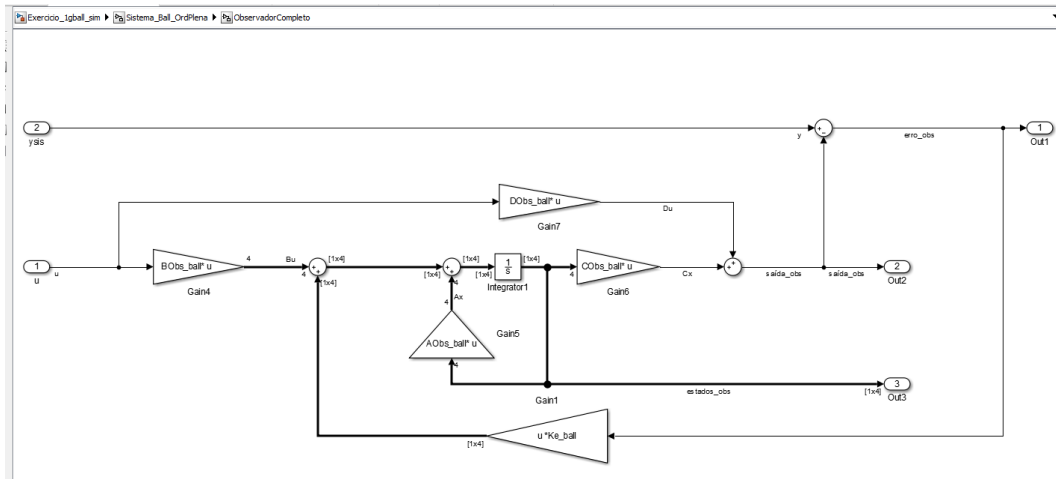
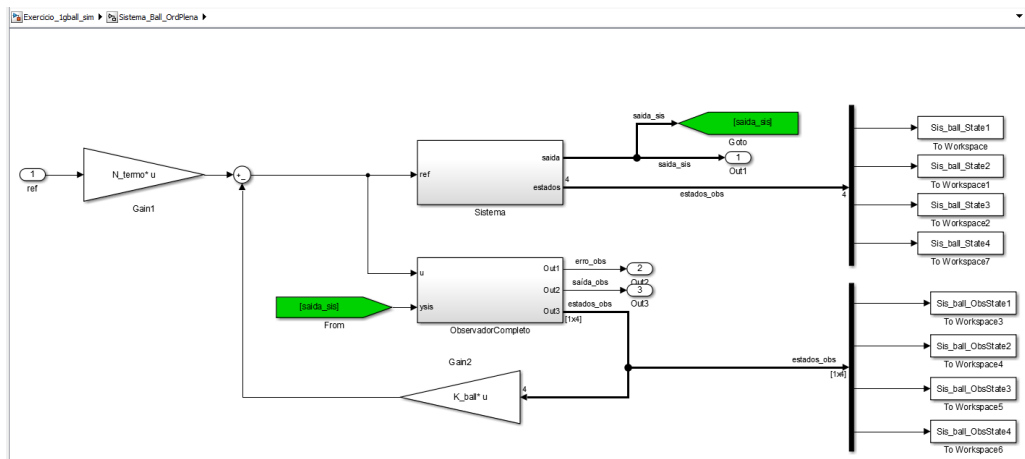
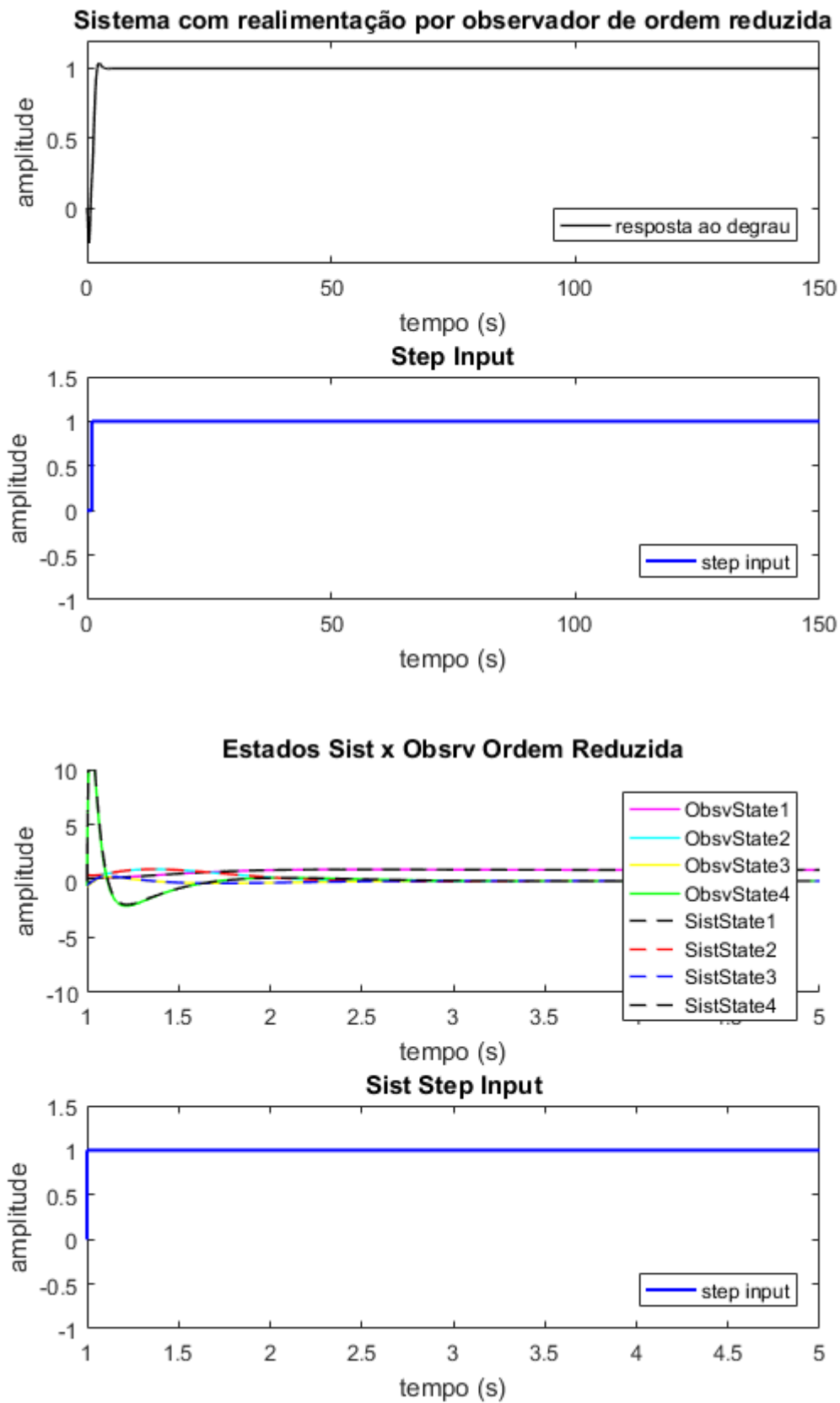
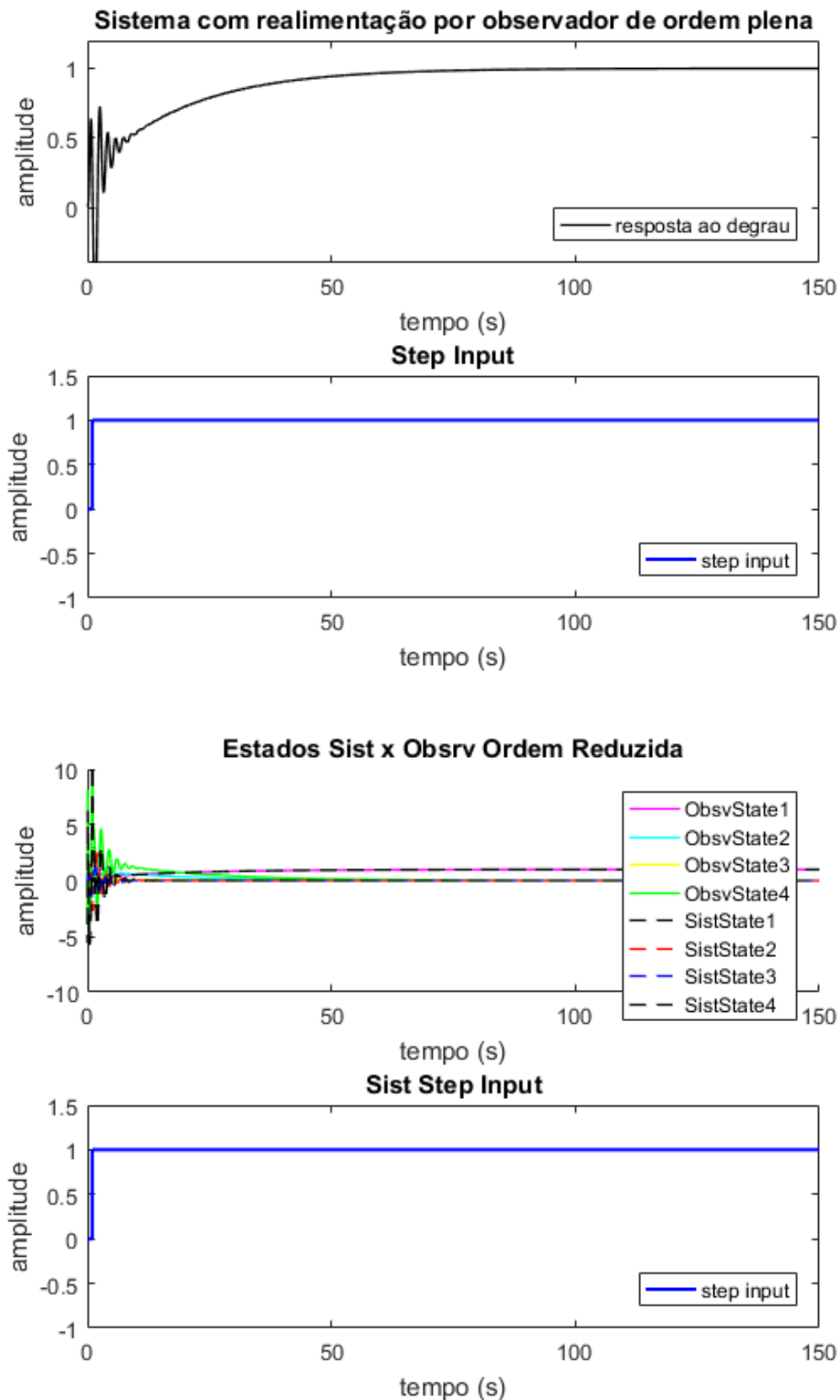


DIAGRAMA DE BLOCOS DO CONTROLE POR REALIMENTAÇÃO DE ESTADOS COM OBSERVADOR DE ORDEM COMPLETA:



Resultados obtidos após o controle aplicado:





R) Notamos que o observador e ordem reduzida apresentou um melhor comportamento para o controle por realimentação de estados.

Obs: Todos os modelos, scripts e referências utilizadas estão disponíveis no GITHUB pelo link: <https://github.com/well-nogueira/PME5009>