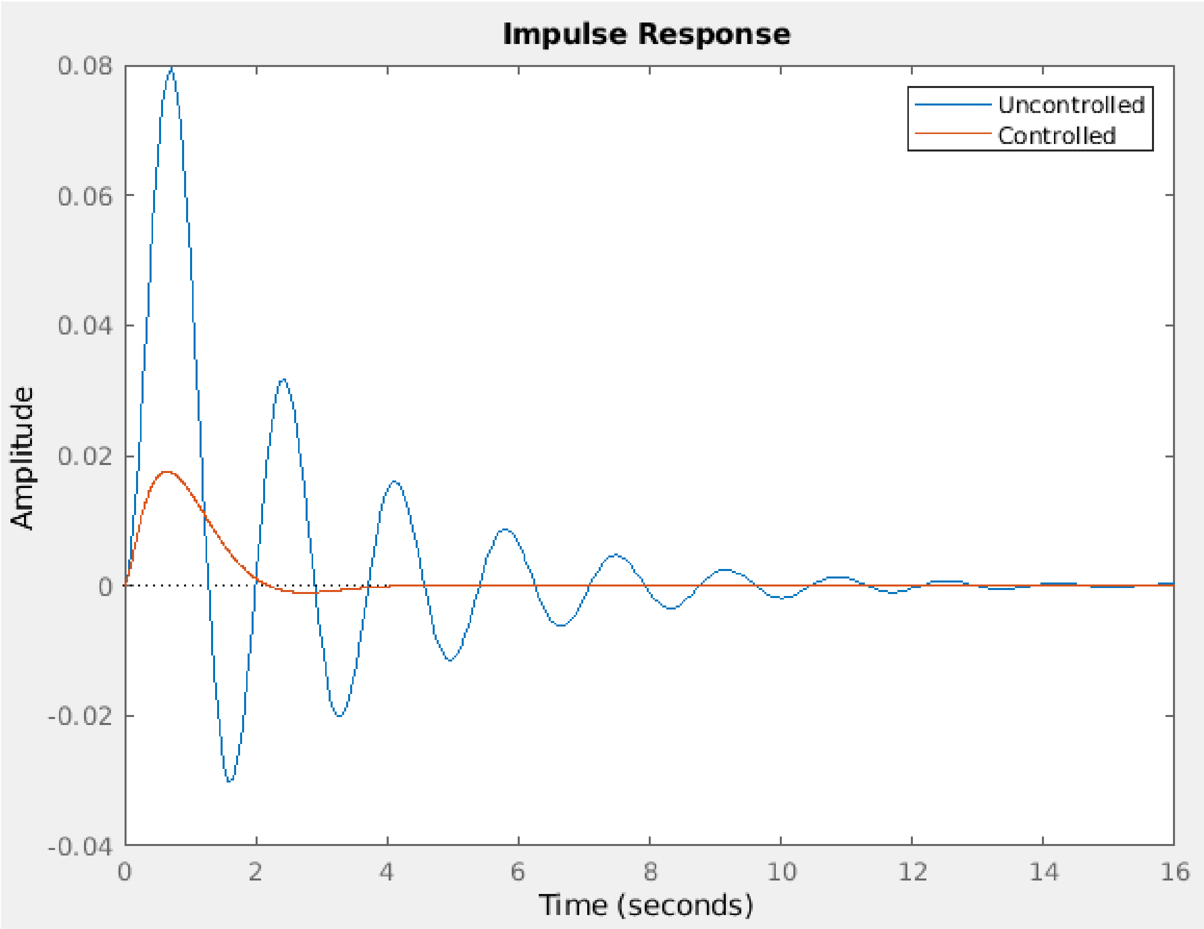
Formative

1.

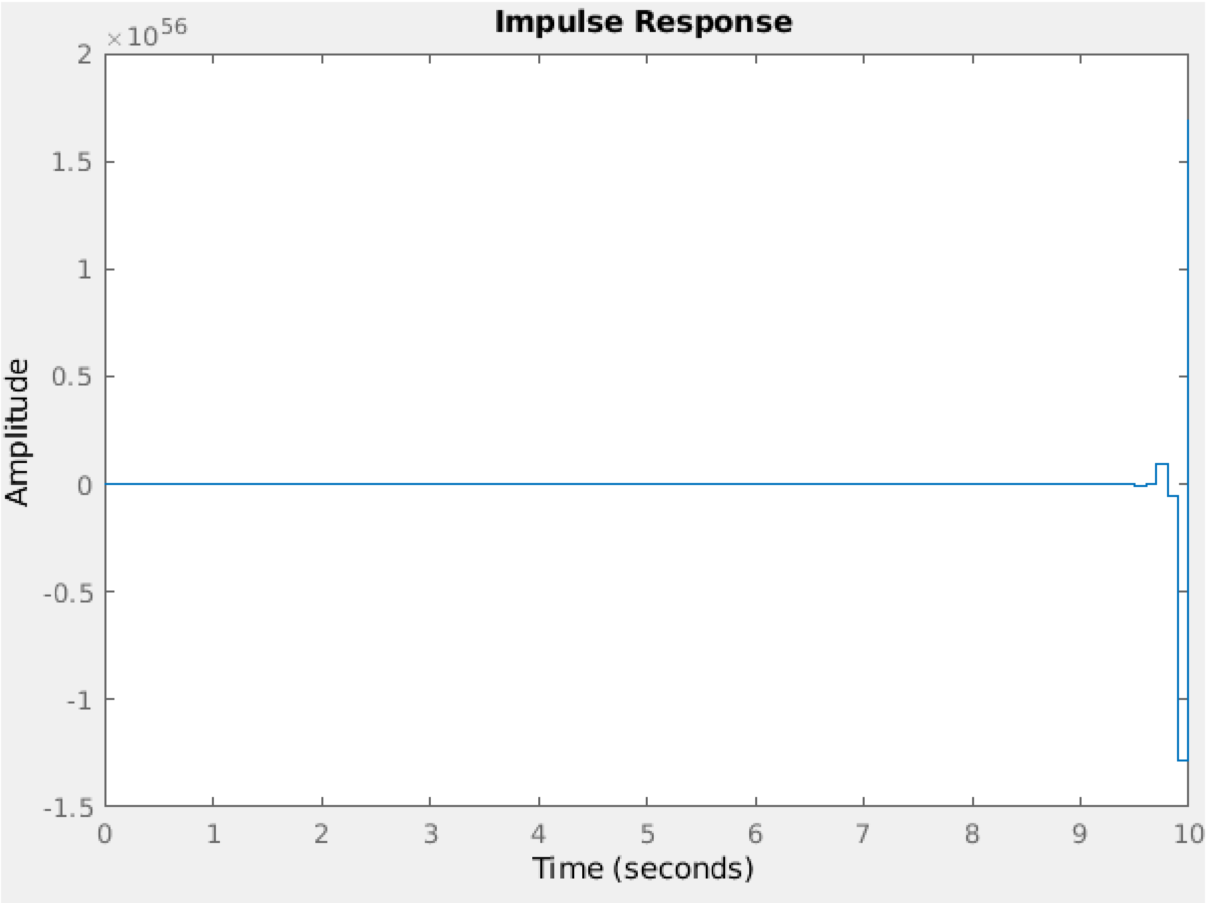
Given the continuous time system:

$$A = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ -18 & -15 & -2 \end{bmatrix}$$
$$B = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}$$
$$C = \begin{bmatrix} 1 & 0 & 0 \end{bmatrix}$$
$$D = [0]$$

That is controlled with 3 poles at locations: [-1.33+1.49j, -1.33-1.49j, -13.3];
Giving it a continuous time response of:

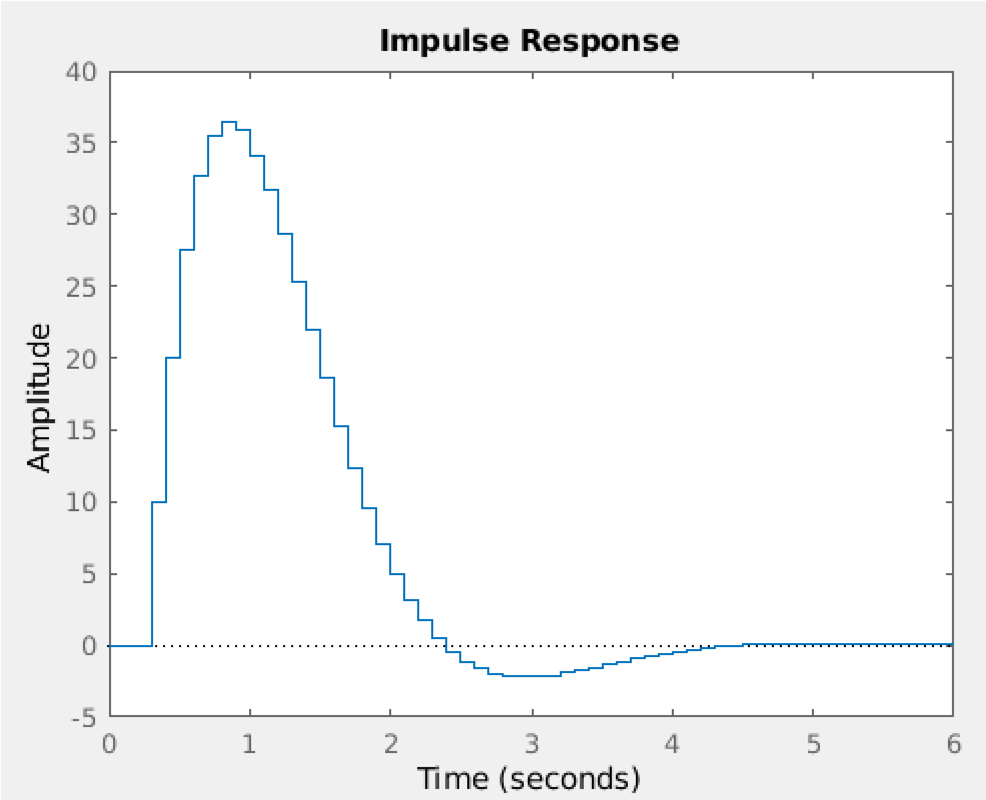


This system was discretized with a sample time of 0.1s using the following command
sys = ss(A,B,C,D,ts);
Which provides the following unregulated system response, which is contrary to the continuous time system which is stable.



So we can create a similar discrete time regulator by converting the poles to the z domain by using the equation $poles_Z = e^{poles_s \times t_s}$ which provides the following poles and then response when we model the regulated system.

[
0.865764957849127 + 0.129962168578086i,
0.865764957849127 - 0.129962168578086i,
0.264477261299824
]



From this we can see that the main difference is the amplitude invoked by the impulse, as the settling time remains about the same. There is a delay of 2 samples before the impulse kicks in though.

Summative

(a)

Show that the system is not controllable, but is stabilisable. You should identify which eigenvalues/modes are controllable and which are uncontrollable.

1	0	-10	0	100	0	-1000	0	10000	0	-100000	0
0	2	0	-100	0	5000	0	-250000	0	12500000	0	-625000...
0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0

The controllability matrix shown above was found to have a rank of 2, which is less than the number of rows in the A matrix. This means that it is uncontrollable.

-10	0	0	0	0	0
0	-50	0	0	0	0
0	0	-100	500	0	0
0	0	-500	-100.0000	0	0
0	0	0	0	-200	500.0000
0	0	0	0	-500.0000	-200

The eigen values shown in the canonical form of the A matrix above show that this system is in fact stabilisable as all the real components of the eigenvectors (the diagonal) are negative, indicating that any exponential components will decay to 0.

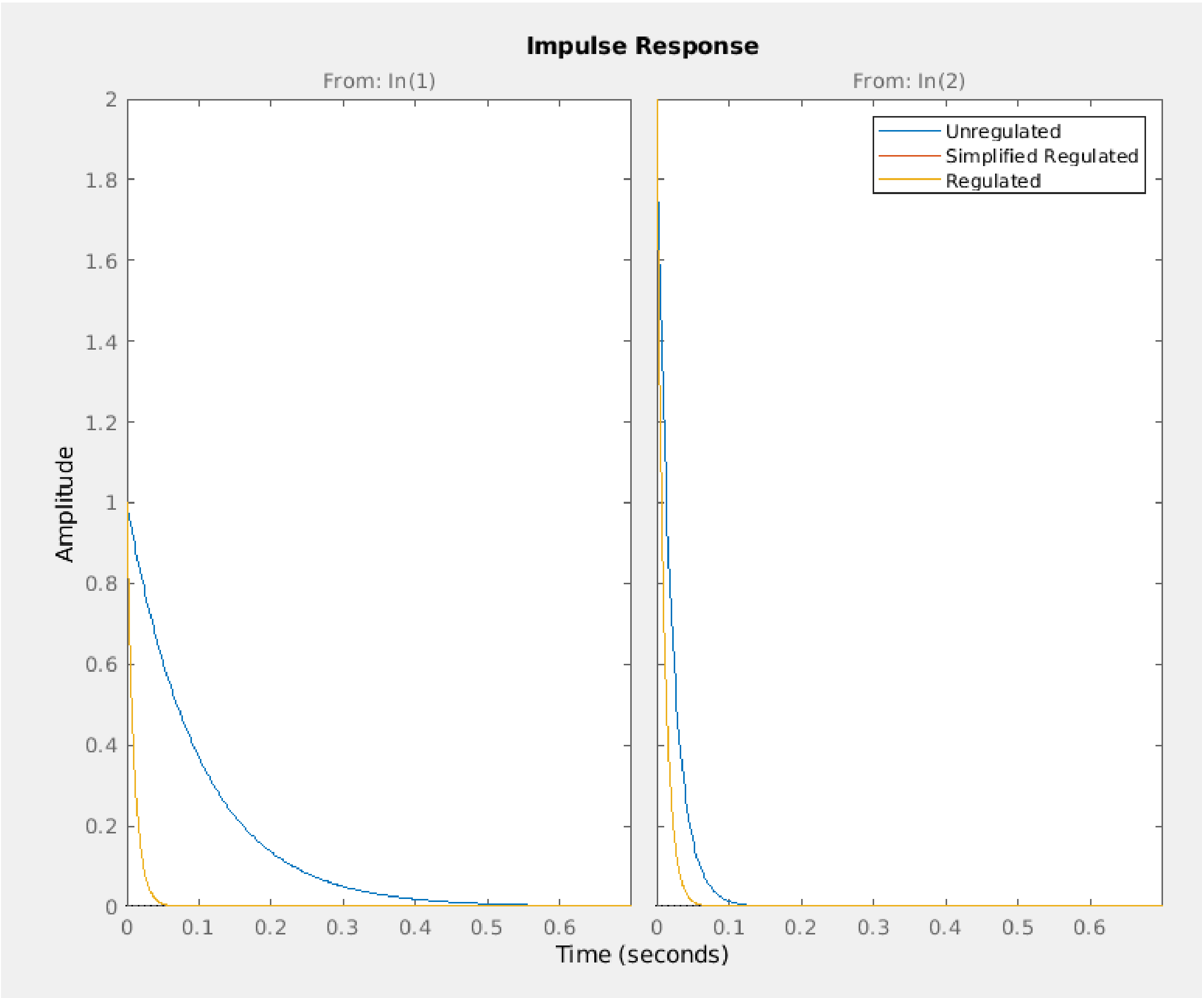
(b)

Design a regulator that ensures that the controllable modes decay at least as quickly as the slowest uncontrollable modes. You should demonstrate the performance of your system using appropriate simulation.

As shown from the canonical form of the A matrix, above, the slowest uncontrollable modes occur at a real value of -100, so this will be the two pole locations when we simplify the model to the controllable state variables.

```
A_shrink = csys.A(1:2,1:2);
B_shrink = csys.B(1:2,1:2);

K_shrink = place(A_shrink,B_shrink,p);
K = zeros(size(B.'));
K(1:2,1:2) = K_shrink

reg_sys = ss(A-B*K, B, C, D)
```

The above code was used to shrink the A and B matrices and place the poles for the generation of a K matrix, which was then fed into the sys to regulate it.



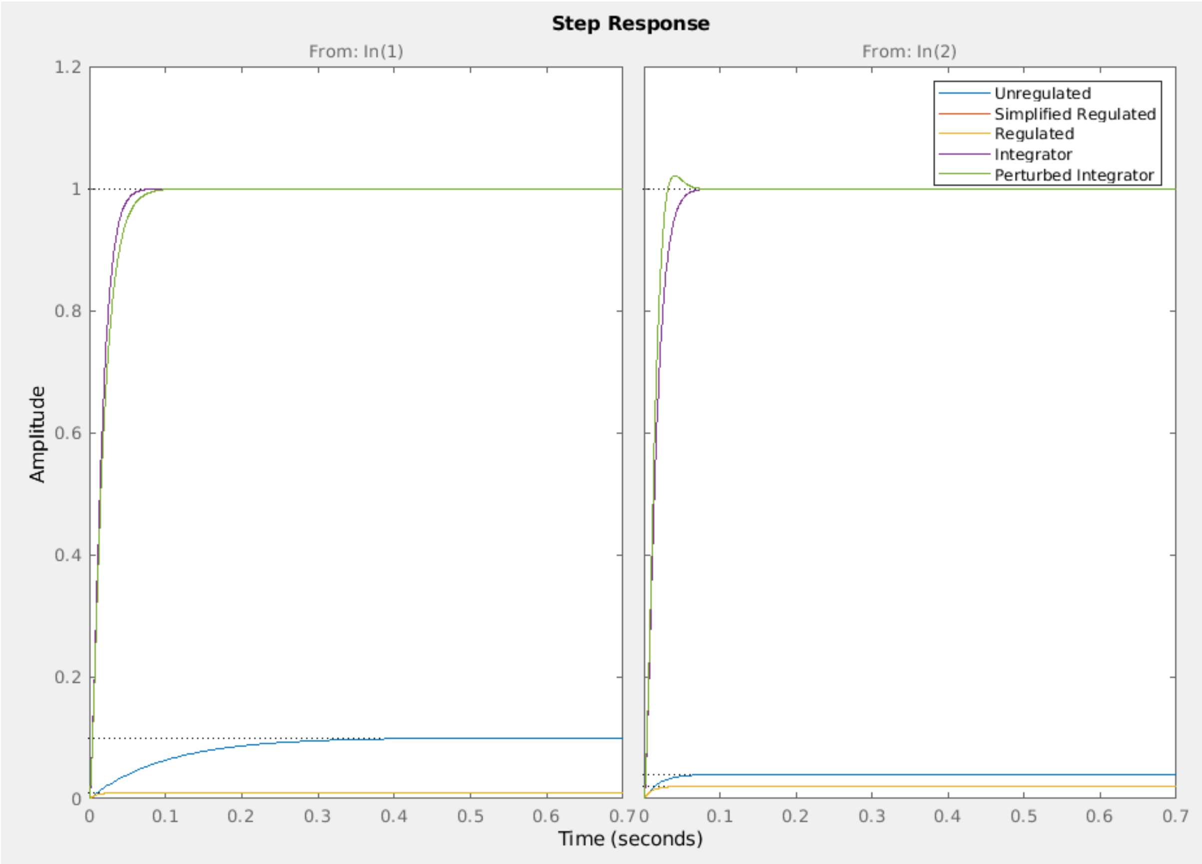
The response of this regulator can be seen in this figure which shows the impulse response of the unregulated system, the regulated simplified system and the yet to be discussed regulation of the full system. It is the orange line which is aligned with the yellow line (directly on top of it). From this it can be seen that the implementation of this regulator provides a significant speed increase to the settling time.

The above figure showing impulse responses also shows that when controlling the full system, there is no noticeable change in response for this system, when setting our speed controlling poles to the same -100. The other poles are set to the uncontrollable modes of the system, as shown by the below code.

```
p = [-100+500j -100-500j -200+500j -200-500j -100 -100];
k_adv = place(A, B, p);
sys_adv = ss(A-B*k_adv, B, C, D)
```

(c)

Add integrators for both controllable states, so that the system achieves x1 = r1 and x2 = r2 in steady state. You should demonstrate the effectiveness of your integrator solution by perturbing the system somewhat from the nominal plant model. You may wish to incorporate a prefilter, but are not required to.



The above figure shows the unregulated system and the regulated system, to emphasise the steady state error present with a step input. The purple line is then the implementation of an integrator on both inputs of the system which is shown to respond in approximately the same time as the systems without integrators, and removes the steady state error.

The system was then perturbed by modifying the A matrix to:

```
A_perturbed = [-50 0 -90 355 -205 0;
               0 10 -50 125 -375 1500;
               0 0 -100 200 -350 500;
               0 0 -600 150 200 200;
               0 0 500 -350 -400 500;
               0 0 0 -250 -650 -200;
              ];
```

And from the green line in the above figure it can be seen that the integrator is still effective at removing the steady state error, however some overshoot has been introduced.

There was no method to changing the A matrix, values were changed at random, however the link between the inputs and controllable state variables were intentionally switched in magnitude.

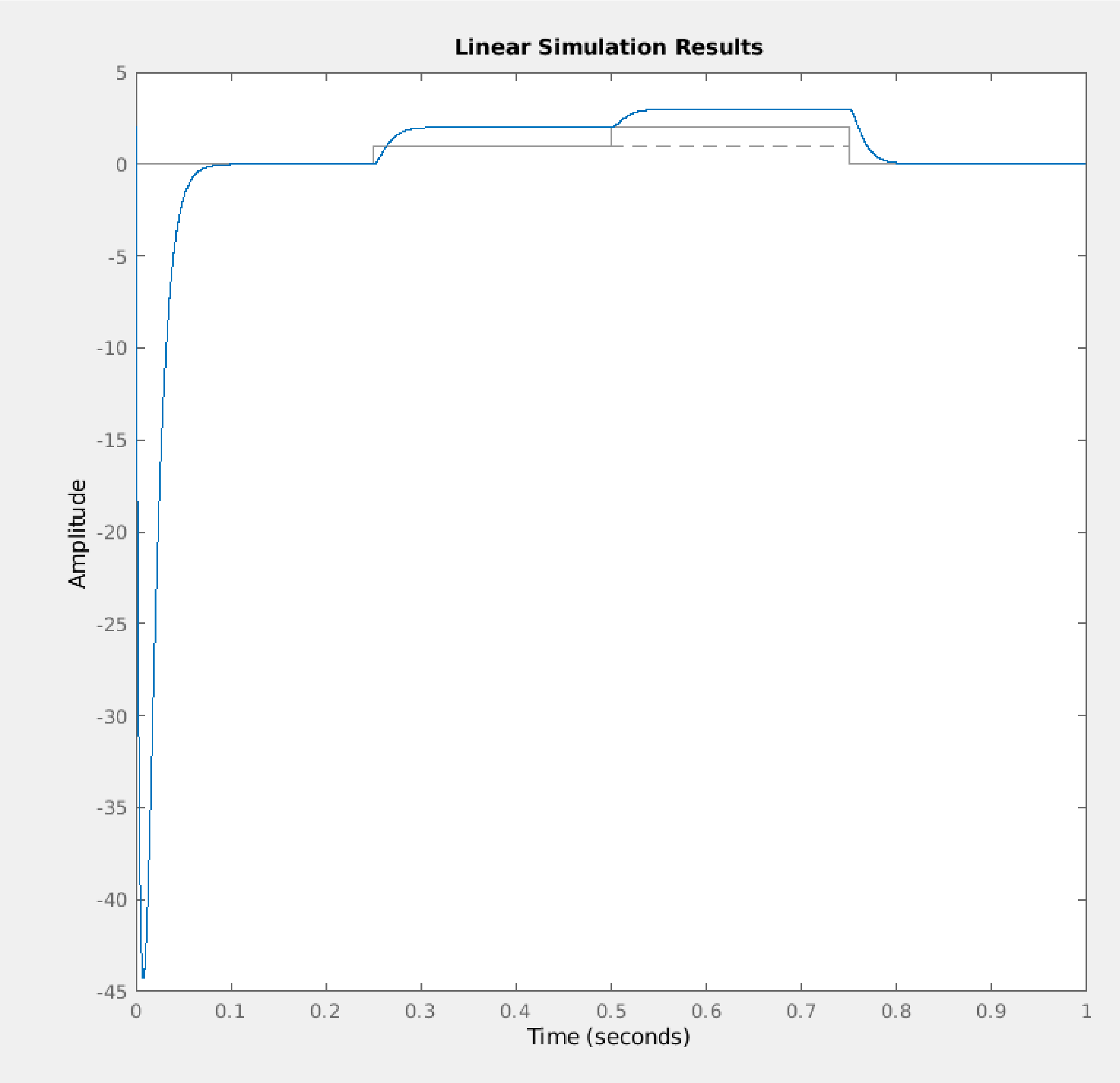
```
A_i = [A zeros(length(A),1) zeros(length(A),1) ; -1 0 0 0 0 0 0 0 0; 0 -1 0 0 0 0 0 0 0];
B_i = [B ; 0 0; 0 0];

alpha_c = [p -150 -150];

k_i = place(A_i, B_i, alpha_c);
sys_i = ss(A_i-B_i*k_i, [zeros(size(B)); 1 0; 0 1], [C 0 0], D);
```

(d)

Plot the performance of the full system when step inputs are applied to r1 and r2. You should apply a step to r1, wait for the transient to die down and then apply a step to r2. Note that during your simulation you should set some non-trivial initial conditions that cause the uncontrollable modes to be non-zero. As they are uncontrollable, the steps that you apply will have no effect upon them.

The following initial conditions were set
x0 = [1 1 0 1 0 0 -1 0];

Which resulted in the following response to 4 step inputs



From this graph it can be seen that the uncontrollable and controllable state variables create an initial transient response. The negative value creates a particularly large transient, which would be physically stressful on any application. However, this quickly settles within 100ms. After the initial transient from the initial state, the following step inputs can be seen to have very little transients aside from a slight delay in settling.

Appendix

All the code used for this

```
close all
clear

A = [-10 0 -90 295 -205 0;
     0 -50 -50 125 -375 1000;
     0 0 -100 200 -300 500;
     0 0 -500 100 200 500;
     0 0 500 -300 -400 500;
     0 0 0 -250 -250 -200;
    ];

B = [1 0;
     0 2;
     0 0;
     0 0;
     0 0;
     0 0;
     ];

C = [1 1 -2 0 0 0];

D = [0 0];

sys = ss(A, B, C, D);

% a - non-Controllable but Stabilisable
Co = ctrb(A,B)
rank_Co = rank(Co)
%x1 and x2 are controllable
csys = canon(sys,'modal')
% All are stable as all have a -ive exponential decay, so go to 0

% b
% Slowest is -100+-500j
p = [-100 -100];

A_shrink = csys.A(1:2,1:2);
B_shrink = csys.B(1:2,1:2);

K_shrink = place(A_shrink,B_shrink,p);
K = zeros(size(B.'));
K(1:2,1:2) = K_shrink
```

```
reg_sys = ss(A-B*K, B, C, D)


p = [-100+500j -100-500j -200+500j -200-500j -100 -100];
k_adv = place(A, B, p);
sys_adv = ss(A-B*k_adv, B, C, D)

figure(1)
impulse(sys)
hold on
impulse(reg_sys)
impulse(sys_adv)
hold off
legend("Unregulated", "Simplified Regulated", "Regulated")

% c integrators
A_i = [A zeros(length(A),1) zeros(length(A),1) ; -1 0 0 0 0 0 0 0; 0 -1 0 0 0 0 0 0];
B_i = [B ; 0 0; 0 0];

alpha_c = [p -150 -150];

k_i = place(A_i, B_i, alpha_c);
sys_i = ss(A_i-B_i*k_i, [zeros(size(B)); 1 0; 0 1], [C 0 0], D);

A_perturbed = [-50 0 -90 355 -205 0;
               0 10 -50 125 -375 1500;
               0 0 -100 200 -350 500;
               0 0 -600 150 200 200;
               0 0 500 -350 -400 500;
               0 0 0 -250 -650 -200;
              ];
A_i_perturbed = [A_perturbed zeros(length(A_perturbed),1) zeros(length(A_perturbed),1) ;
-1 0 0 0 0 0 0 0; 0 -1 0 0 0 0 0 0];
sys_i_perturbed = ss(A_i_perturbed-B_i*k_i, [zeros(size(B)); 1 0; 0 1], [C 0 0], D);


figure(2)
step(sys)
hold on
step(reg_sys)
step(sys_adv)
step(sys_i)
step(sys_i_perturbed)
hold off
legend("Unregulated", "Simplified Regulated", "Regulated", "Integrator", "Perturbed
Integrator")

% d
```

```
x0 = [1 1 0 1 0 0 -1 0];
t = linspace(0,1,10000).';
u = [repmat([0 0], length(t)/4, 1);
    repmat([1 1], length(t)/4, 1);
    repmat([2 1], length(t)/4, 1);
    repmat([0 0], length(t)/4, 1);];

figure(3)
lsim(sys_i, u, t, x0)
```