```
function HW8()
clear ; close all ; clc ;

Part1()

Part2()

    function Part1()
        I = [ 1200 0 0 ; 0 2000 0 ; 0 0 2800 ] ;
        eps0 = [ -.5 ; -.5 ; .5 ] ;
        eta0 = .5 ;
        w0 = [ 0 0 0 ]' ;
        ts = 30 ;
        zeta = .65 ; % Damping coefficient
        wn = log( 0.02*sqrt( 1 - zeta^2 ) )/( -zeta*ts ) ;

        kp = 2.*I.*wn^2 ;
        kd = I.*2*zeta*wn ;

        state0 = [ eps0 ; eta0 ; w0 ] ;
        tmax = 60 ;
        tspan = [ 0 , tmax ] ;
        opts = odeset( 'RelTol' , 1e-8 , 'AbsTol' , 1e-8 ) ;

        [ t , statenew ] = ode45( @LinearControlne , tspan , state0 ,
 opts , I , kp , kd ) ;
        [ tn , statenewn ] = ode45( @NonLinearControlne , tspan ,
 state0 , opts , I , kp , kd ) ;

        figure
        subplot( 2 , 1 , 1 )
        hold on
        plot( t , statenew( : , 1 ) )
        plot( t , statenew( : , 2 ) )
        plot( t , statenew( : , 3 ) )
        plot( t , statenew( : , 4 ) )
        xlabel( 'Time (s)' )
        ylabel( 'Quaternion Value' )
        title( 'P1 Quaternions with Linear Control' )
        legend( 'Eps(1)' , 'Eps(2)' , 'Eps(3)' , 'Eta' )
        hold off

        subplot( 2 , 1 , 2 )
        hold on
        plot( tn , statenewn( : , 1 ) )
        plot( tn , statenewn( : , 2 ) )
        plot( tn , statenewn( : , 3 ) )
        plot( tn , statenewn( : , 4 ) )
        xlabel( 'Time (s)' )
        ylabel( 'Quaternion Value' )
        title( 'P1 Quaternions with Non-Linear Control' )
        legend( 'Eps(1)' , 'Eps(2)' , 'Eps(3)' , 'Eta' )
```

```
        hold off


        figure
        subplot( 2 , 1 , 1 )
        hold on
        plot( t , statenew( : , 5 ) )
        plot( t , statenew( : , 6 ) )
        plot( t , statenew( : , 7 ) )
        xlabel( 'Time (s)' )
        ylabel( 'Angular Velocity (rad/s)' )
        title( 'P1 Angular Velocity with Linear Control' )
        legend( 'x' , 'y' , 'z' )
        hold off

        subplot( 2 , 1 , 2 )
        hold on
        plot( tn , statenewn( : , 5 ) )
        plot( tn , statenewn( : , 6 ) )
        plot( tn , statenewn( : , 7 ) )
        xlabel( 'Time (s)' )
        ylabel( 'Angular Velocity (rad/s)' )
        title( 'P1 Angular Velocity with Non-Linear Control' )
        legend( 'x' , 'y' , 'z' )
        hold off

        disp( 'The linear control sytem controlled the attitude nearly as well as the ' )
        disp( 'non-linear system but the scalar portion of the quaternion never ' )
        disp( 'changed from its initial value while it went to 1 with the non-linear' )
        disp( 'control. The linear system also treated the change in the 2nd and 3rd ' )
        disp( 'component of the vector portion of the quaternion as the same but this ' )
        disp( 'was not true in the non-linear case. The non-linear case also showed ' )
        disp( 'a difference in the angular velocity around the y and z axes which was' )
        disp( 'not shown in the linear case' )
        disp( ' ' )

    end

    function Part2()
    I = [ 1200 0 0 ; 0 2000 0 ; 0 0 2800 ] ;
    eps01 = [ .10 ; 0 ; .1 ] ;
    eta01 = .9999 ;
    eps02 = [ .45 ; 0 ; .45 ] ;
    eta02 = .7777 ;
    w0 = [ 0 0 0 ]' ;
    ts = 30 ;
    zeta = .65 ; % Damping coefficient
```

```
   wn = log( 0.02*sqrt( 1 - zeta^2 ) )/( -zeta*ts ) ;

   ceta = 1 ;
   ceps = [ 0 ; 0 ; 0 ] ;

   kp = 2.*I.*wn^2 ;
   kd = I.*2*zeta*wn ;

   state01 = [ eps01 ; eta01 ; w0 ] ;
   state02 = [ eps02 ; eta02 ; w0 ] ;
   tmax = 60 ;
   tspan = [ 0 , tmax ] ;
   opts = odeset( 'RelTol' , 1e-8 , 'AbsTol' , 1e-8 ) ;

   [ t1 , statenew1 ] = ode45( @LinearControl , tspan , state01 ,
opts , I , kp , kd , ceta , ceps) ;
   [ tn1 , statenewn1 ] = ode45( @NonLinearControl , tspan ,
state01 , opts , I , kp , kd , ceta , ceps ) ;

   [ t2 , statenew2 ] = ode45( @LinearControl , tspan , state02 ,
opts , I , kp , kd , ceta , ceps) ;
   [ tn2 , statenewn2 ] = ode45( @NonLinearControl , tspan ,
state02 , opts , I , kp , kd , ceta , ceps ) ;

   % Linear 1
   figure
   subplot( 2 , 1 , 1 )
   hold on
   plot( t1 , statenew1( : , 1 ) )
   plot( t1 , statenew1( : , 2 ) )
   plot( t1 , statenew1( : , 3 ) )
   plot( t1 , statenew1( : , 4 ) )
   xlabel( 'Time (s)' )
   ylabel( 'Quaternion Value' )
   title( 'P2 Quaternions with Linear Control' )
   legend( 'Eps(1)' , 'Eps(2)' , 'Eps(3)' , 'Eta' , 'Location'
, 'east' )
   hold off

   subplot( 2 , 1 , 2 )
   hold on
   plot( tn1 , statenewn1( : , 1 ) )
   plot( tn1 , statenewn1( : , 2 ) )
   plot( tn1 , statenewn1( : , 3 ) )
   plot( tn1 , statenewn1( : , 4 ) )
   xlabel( 'Time (s)' )
   ylabel( 'Quaternion Value' )
   title( 'P2 Quaternions with Non-Linear Control' )
   legend( 'Eps(1)' , 'Eps(2)' , 'Eps(3)' , 'Eta' , 'Location'
, 'east' )
   hold off

   figure
   subplot( 2 , 1 , 1 )
```

```
hold on
plot( t1 , statenew1( : , 5 ) )
plot( t1 , statenew1( : , 6 ) )
plot( t1 , statenew1( : , 7 ) )
xlabel( 'Time (s)' )
ylabel( 'Angular Velocity (rad/s)' )
title( 'P2 Angular Velocity with Linear Control' )
legend( 'x' , 'y' , 'z' , 'Location' , 'east' )
hold off

subplot( 2 , 1 , 2 )
hold on
plot( tn1 , statenewn1( : , 5 ) )
plot( tn1 , statenewn1( : , 6 ) )
plot( tn1 , statenewn1( : , 7 ) )
xlabel( 'Time (s)' )
ylabel( 'Angular Velocity (rad/s)' )
title( 'P2 Angular Velocity with Non-Linear Control' )
legend( 'x' , 'y' , 'z' , 'Location' , 'east' )
hold off

% 2
figure
subplot( 2 , 1 , 1 )
hold on
plot( t2 , statenew2( : , 1 ) )
plot( t2 , statenew2( : , 2 ) )
plot( t2 , statenew2( : , 3 ) )
plot( t2 , statenew2( : , 4 ) )
xlabel( 'Time (s)' )
ylabel( 'Quaternion Value' )
title( 'P2 2nd Situation Quaternions with Linear Control' )
legend( 'Eps(1)' , 'Eps(2)' , 'Eps(3)' , 'Eta' )
hold off

subplot( 2 , 1 , 2 )
hold on
plot( tn2 , statenewn2( : , 1 ) )
plot( tn2 , statenewn2( : , 2 ) )
plot( tn2 , statenewn2( : , 3 ) )
plot( tn2 , statenewn2( : , 4 ) )
xlabel( 'Time (s)' )
ylabel( 'Quaternion Value' )
title( 'P2 2nd Situation Quaternions with Non-Linear Control' )
legend( 'Eps(1)' , 'Eps(2)' , 'Eps(3)' , 'Eta' )
hold off


figure
subplot( 2 , 1 , 1 )
hold on
plot( t2 , statenew2( : , 5 ) )
plot( t2 , statenew2( : , 6 ) )
plot( t2 , statenew2( : , 7 ) )
```

```
    xlabel( 'Time (s)' )
    ylabel( 'Angular Velocity (rad/s)' )
    title( 'P2 2nd Situation Angular Velocity with Linear Control' )
    legend( 'x' , 'y' , 'z' )
    hold off

    subplot( 2 , 1 , 2 )
    hold on
    plot( tn2 , statenewn2( : , 5 ) )
    plot( tn2 , statenewn2( : , 6 ) )
    plot( tn2 , statenewn2( : , 7 ) )
    xlabel( 'Time (s)' )
    ylabel( 'Angular Velocity (rad/s)' )
    title( 'P2 2nd Situation Angular Velocity with Non-Linear Control'
)
    legend( 'x' , 'y' , 'z' )
    hold off

        for ii = 1:length(t1)
            eps(1:3,1) = statenew1( ii , 1:3 ) ;
            eta = statenew1( ii , 4 ) ;
            quat = [ eta ; eps ]' ;
            cquat = [ ceta ; ceps ]' ;
            qstar = quatconj( cquat ) ;
            qerr = quatmultiply( qstar , quat ) ;
            epse(1:3,1) = qerr( 2:4 ) ;
            T1(:,ii) = -kp*epse - kd*statenew1( ii , 5:7 )' ;
        end
        for ii = 1:length(tn1)
            eps(1:3,1) = statenewn1( ii , 1:3 ) ;
            eta = statenewn1( ii , 4 ) ;
            quat = [ eta ; eps ]' ;
            cquat = [ ceta ; ceps ]' ;
            qstar = quatconj( cquat ) ;
            qerr = quatmultiply( qstar , quat ) ;
            epse(1:3,1) = qerr( 2:4 ) ;
            Tn1(:,ii) = -kp*epse - kd*statenewn1( ii , 5:7 )' ;
        end
        for ii = 1:length(t2)
            eps(1:3,1) = statenew2( ii , 1:3 ) ;
            eta = statenew2( ii , 4 ) ;
            quat = [ eta ; eps ]' ;
            cquat = [ ceta ; ceps ]' ;
            qstar = quatconj( cquat ) ;
            qerr = quatmultiply( qstar , quat ) ;
            epse(1:3,1) = qerr( 2:4 ) ;
            T2(:,ii) = -kp*epse - kd*statenew2( ii , 5:7 )' ;
        end
        for ii = 1:length(tn2)
            eps(1:3,1) = statenewn2( ii , 1:3 ) ;
            eta = statenewn2( ii , 4 ) ;
            quat = [ eta ; eps ]' ;
            cquat = [ ceta ; ceps ]' ;
            qstar = quatconj( cquat ) ;
```

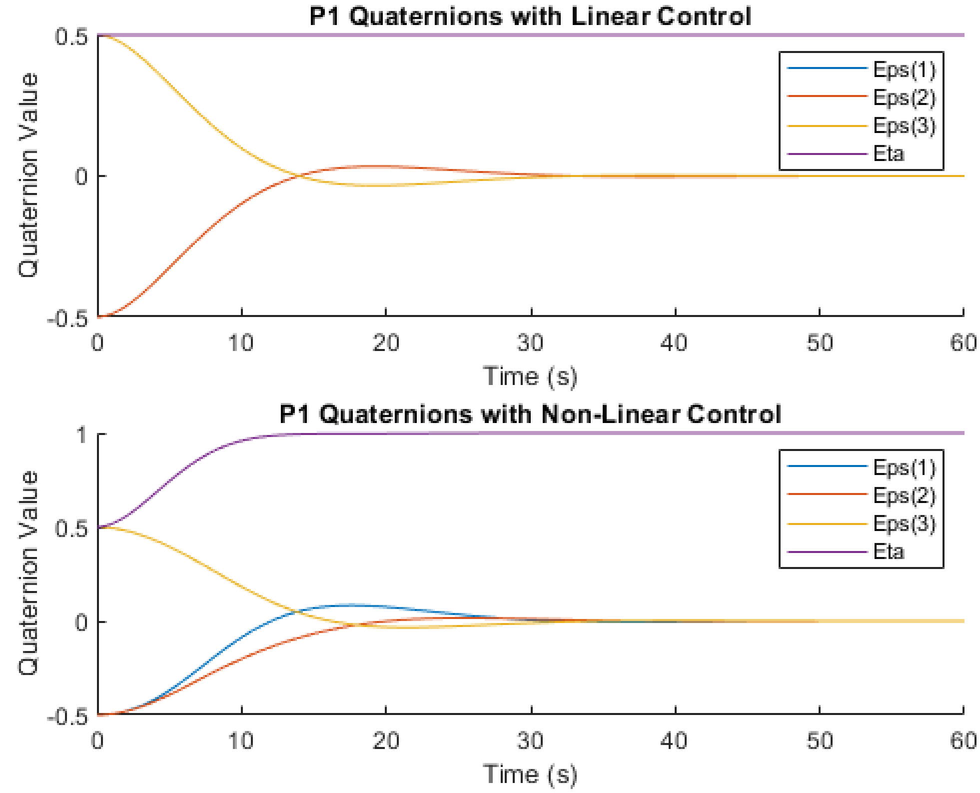
```
        qerr = quatmultiply( qstar , quat ) ;
        epse(1:3,1) = qerr( 2:4 ) ;
        Tn2(:,ii) = -kp*epse - kd*statenewn2( ii , 5:7 )' ;
    end

    figure
    subplot( 2 , 1 , 1 )
    hold on
    plot( t1 , T1( 1 , : ) )
    plot( t1 , T1( 2 , : ) )
    plot( t1 , T1( 3 , : ) )
    xlabel( 'Time (s)' )
    ylabel( 'Torque (N/m)' )
    title( 'P1 1st Situation Torque with Linear Control' )
    legend( 'x' , 'y' , 'z' )
    hold off

    subplot( 2 , 1 , 2 )
    hold on
    plot( tn1 , Tn1( 1 , : ) )
    plot( tn1 , Tn1( 2 , : ) )
    plot( tn1 , Tn1( 3 , : ) )
    xlabel( 'Time (s)' )
    ylabel( 'Torque (Nm)' )
    title( 'P1 1st Situation Torque with Non-Linear Control' )
    legend( 'x' , 'y' , 'z' )
    hold off

    figure
    subplot( 2 , 1 , 1 )
    hold on
    plot( t2 , T2( 1 , : ) )
    plot( t2 , T2( 2 , : ) )
    plot( t2 , T2( 3 , : ) )
    xlabel( 'Time (s)' )
    ylabel( 'Torque (N/m)' )
    title( 'P2 2nd Situation Torque with Linear Control' )
    legend( 'x' , 'y' , 'z' )
    hold off

    subplot( 2 , 1 , 2 )
    hold on
    plot( tn2 , Tn2( 1 , : ) )
    plot( tn2 , Tn2( 2 , : ) )
    plot( tn2 , Tn2( 3 , : ) )
    xlabel( 'Time (s)' )
    ylabel( 'Torque (Nm)' )
    title( 'P2 2nd Situation Torque with Non-Linear Control' )
    legend( 'x' , 'y' , 'z' )
    hold off

    disp( 'The linear control law has no torque about the y-axis
while there is ' )
```

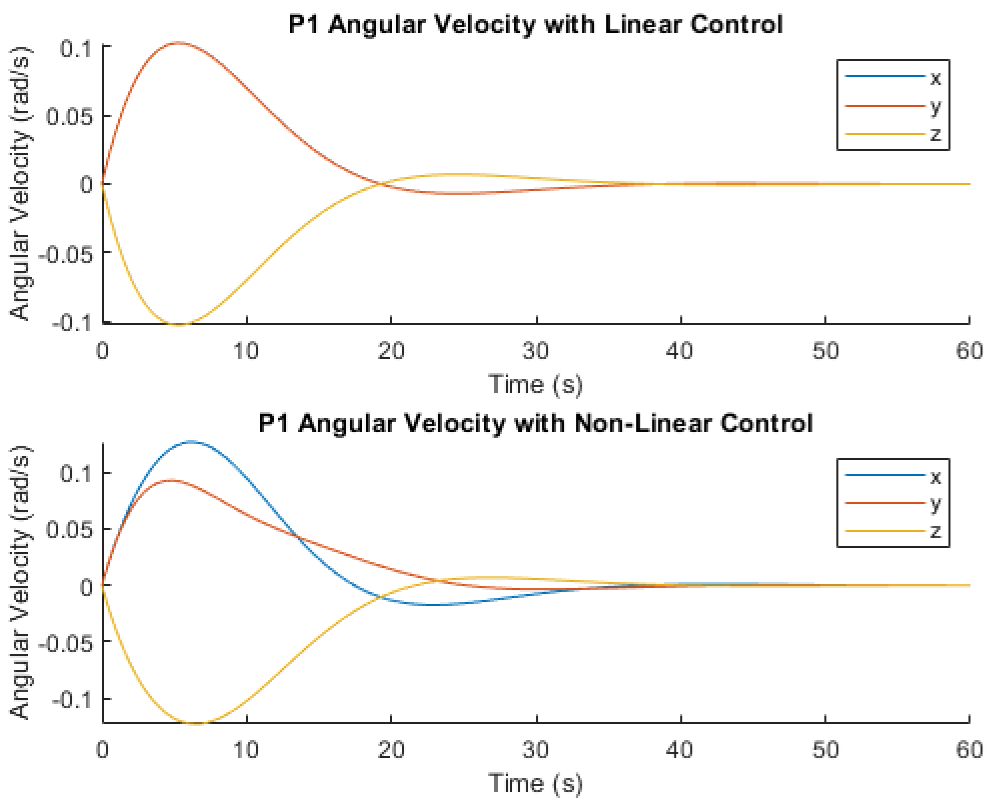
```
        disp( 'significant torque about the y-axis in the non-linear
 case, especially ' )
        disp( 'in case 2. The torques about the x and z axis are
 similar in both cases ' )
    end

function [ dstate , t ] = LinearControl( t , state , I , kp , kd ,
 ceta , ceps )
eps(1:3,1) = state( 1:3 ) ;
eta = state( 4 ) ;
quat = [ eta ; eps ]' ;
cquat = [ ceta ; ceps ]' ;
qstar = quatconj( cquat ) ;
qerr = quatmultiply( qstar , quat ) ;
etae = qerr( 1 ) ;
epse(1:3,1) = qerr( 2:4 ) ;
w(1:3,1) = state( 5:7 ) ;

deps = w./2 ;
deta = 0 ;
dw = inv( I )*( -kp*epse - kd*w ) ;

dstate = [ deps ; deta ; dw ] ;

end

function [ dstate , t ] = LinearControlne( t , state , I , kp , kd )
eps(1:3,1) = state( 1:3 ) ;
eta = state( 4 ) ;
w(1:3,1) = state( 5:7 ) ;

deps = w./2 ;
deta = 0 ;
dw = inv( I )*( -kp*eps - kd*w ) ;

dstate = [ deps ; deta ; dw ] ;

end

function [ dstate , t ] = NonLinearControl( t , state , I , kp , kd ,
 ceta , ceps )
eps(1:3,1) = state( 1:3 ) ;
eta = state( 4 ) ;
quat = [ eta ; eps ]' ;
cquat = [ ceta ; ceps ]' ;
qstar = quatconj( cquat ) ;
qerr = quatmultiply( qstar , quat ) ;
etae = qerr( 1 ) ;
epse(1:3,1) = qerr( 2:4 ) ;
w(1:3,1) = state( 5:7 ) ;

w(1:3,1) = state( 5:7 ) ;
wcross = [ 0 -w(3) w(2) ; w(3) 0 -w(1) ; -w(2) w(1) 0 ] ;
```

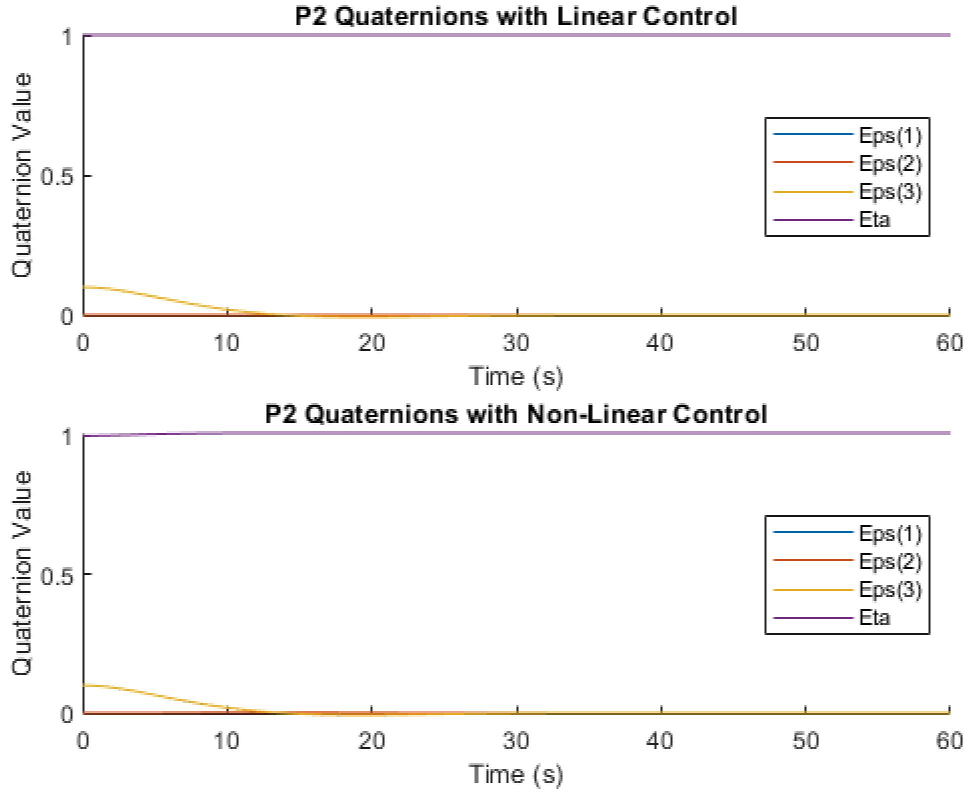
```
epscrosse = [ 0 -epse(3) epse(2) ; epse(3) 0 -epse(1) ; -epse(2)
 epse(1) 0 ] ;

T = -kp*epse - kd*w ;

deps = .5*( etae*eye( 3 ) + epscrosse )*w ;
deta = -.5*epse'*w ;
dw = inv( I )*( -wcross*I*w + T ) ;

dstate = [ deps ; deta ; dw ] ;

end

function [ dstate , t ] = NonLinearControlne( t , state , I , kp ,
 kd )
eps(1:3,1) = state( 1:3 ) ;
eta = state( 4 ) ;
w(1:3,1) = state( 5:7 ) ;
wcross = [ 0 -w(3) w(2) ; w(3) 0 -w(1) ; -w(2) w(1) 0 ] ;
epscross = [ 0 -eps(3) eps(2) ; eps(3) 0 -eps(1) ; -eps(2) eps(1)
 0 ] ;

T = -kp*eps - kd*w ;

deps = .5*( eta*eye( 3 ) + epscross )*w ;
deta = -.5*eps'*w ;
dw = inv( I )*( -wcross*I*w + T ) ;

dstate = [ deps ; deta ; dw ] ;

end

end
```

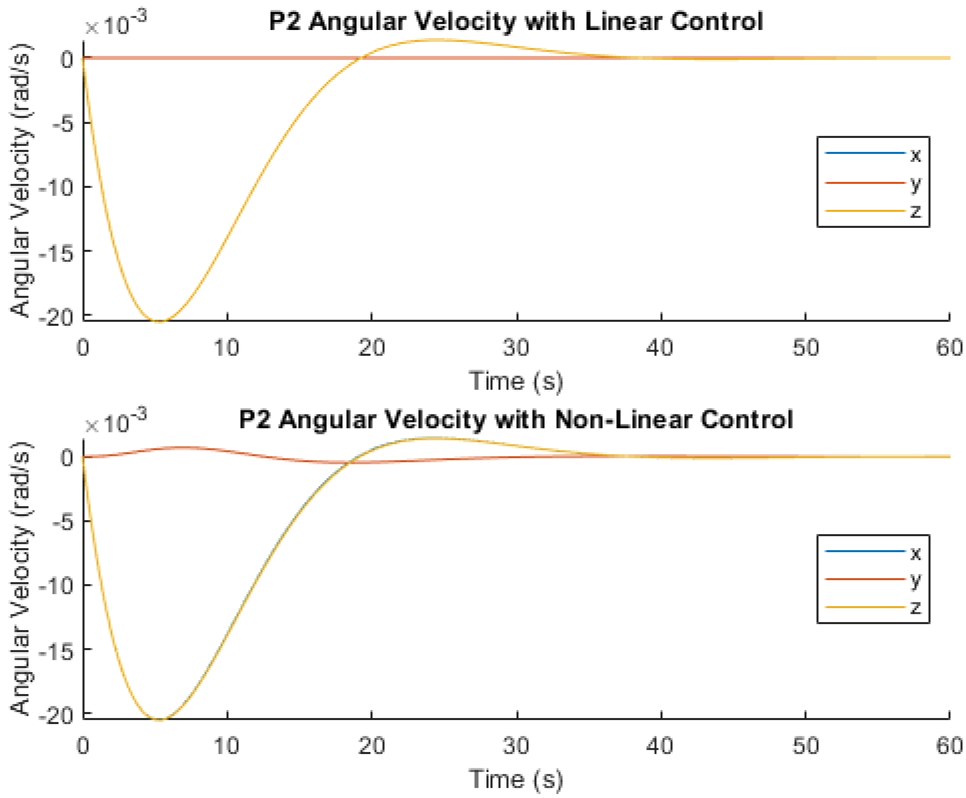
The linear control sytem controlled the attitude nearly as well as the non-linear system but the scalar portion of the quaternion never changed from its initial value while it went to 1 with the non-linear control. The linear system also treated the change in the 2nd and 3rd component of the vector portion of the quaternion as the same but this was not true in the non-linear case. The non-linear case also showed a difference in the angular velocity around the y and z axes which was not shown in the linear case

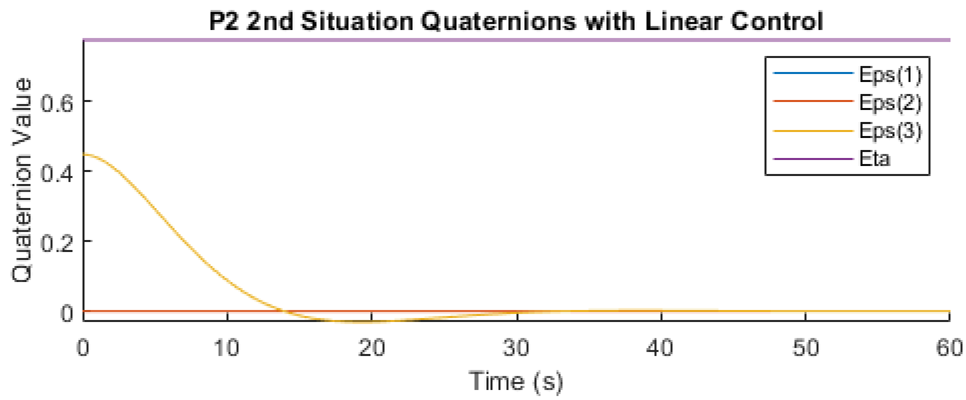
The linear control law has no torque about the y-axis while there is significant torque about the y-axis in the non-linear case, especially in case 2. The torques about the x and z axis are similar in both cases

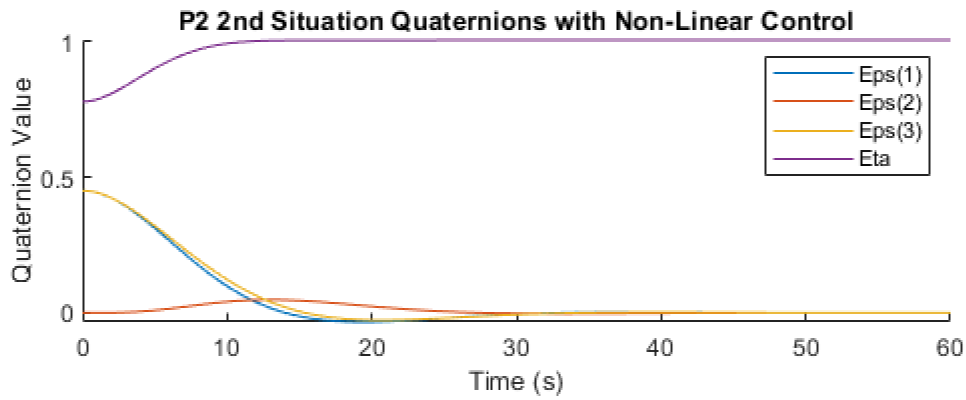

P1 Quaternions with Linear Control
Quaternion Value
Time (s)
Eps(1)
Eps(2)
Eps(3)
Eta
P1 Quaternions with Non-Linear Control
Quaternion Value
Time (s)
Eps(1)
Eps(2)
Eps(3)
Eta

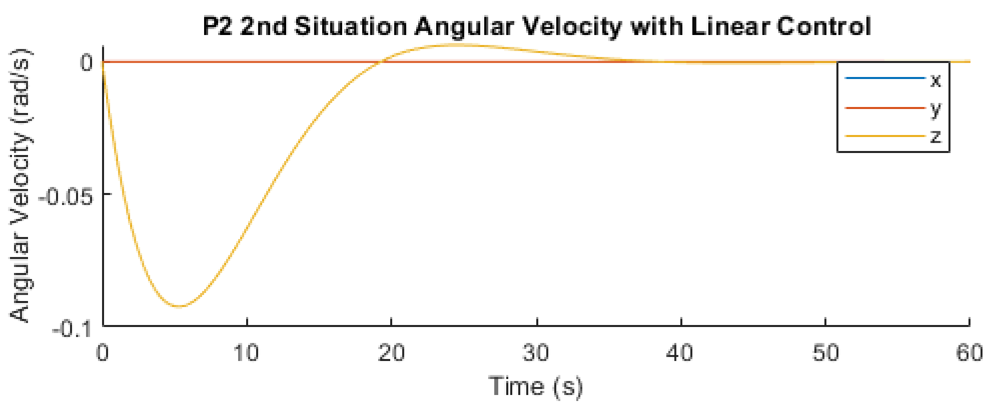

P1 Angular Velocity with Linear Control
Angular Velocity (rad/s)
Time (s)
x
y
z
P1 Angular Velocity with Non-Linear Control
Angular Velocity (rad/s)
Time (s)
x
y
z

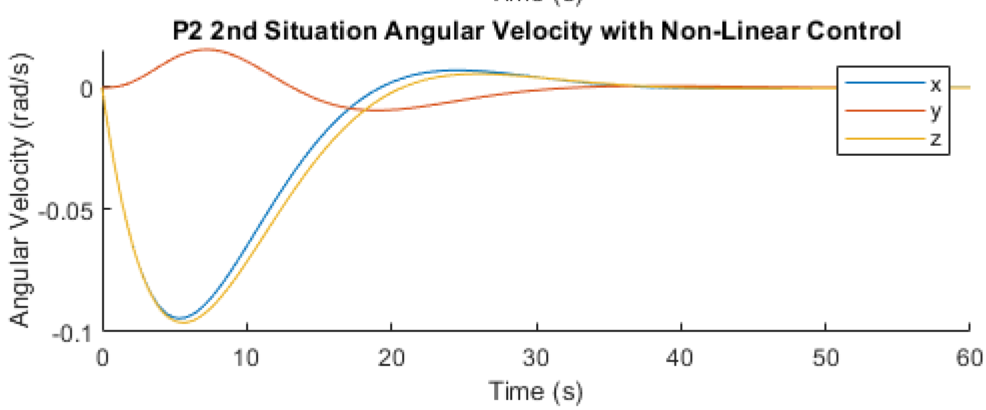

P2 Quaternions with Linear Control
Quaternion Value
Time (s)
Eps(1)
Eps(2)
Eps(3)
Eta
P2 Quaternions with Non-Linear Control
Quaternion Value
Time (s)
Eps(1)
Eps(2)
Eps(3)
Eta

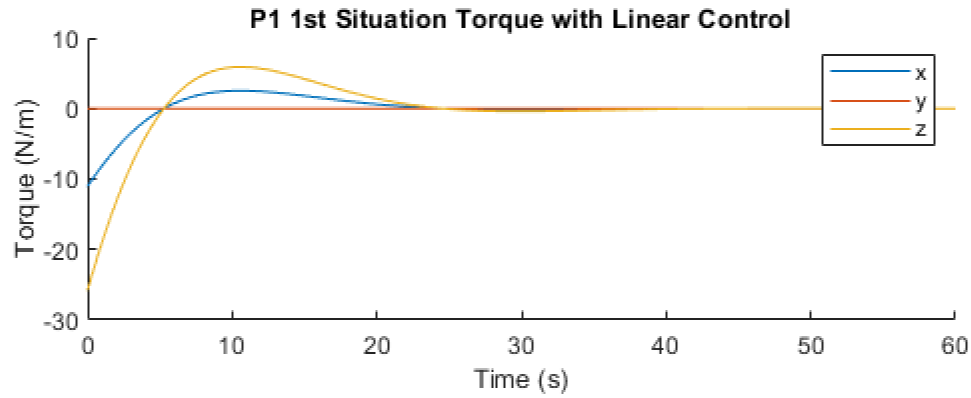

P2 Angular Velocity with Linear Control
$\times 10^{-3}$
Angular Velocity (rad/s)
Time (s)
x
y
z
P2 Angular Velocity with Non-Linear Control
$\times 10^{-3}$
Angular Velocity (rad/s)
Time (s)
x
y
z

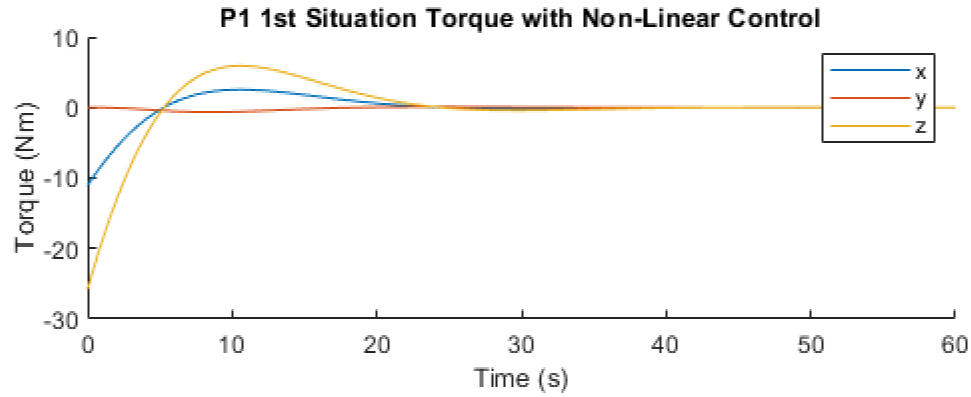

P2 2nd Situation Quaternions with Linear Control
Quaternion Value
Time (s)
Eps(1)
Eps(2)
Eps(3)
Eta

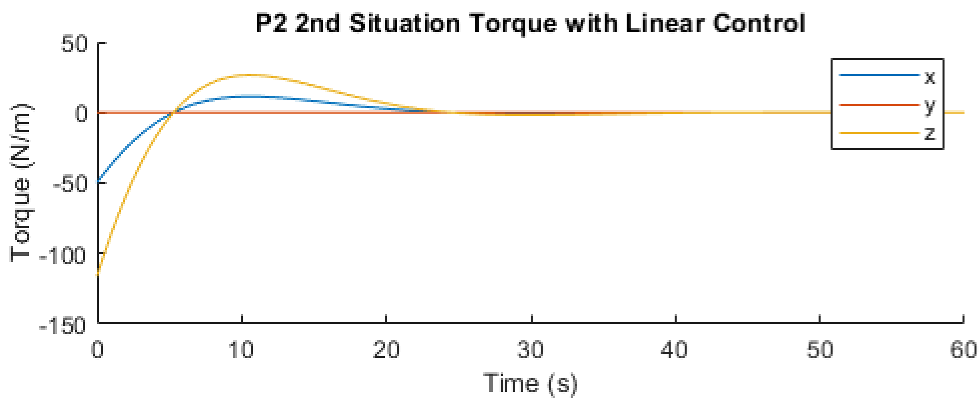

P2 2nd Situation Quaternions with Non-Linear Control
Quaternion Value
Time (s)
Eps(1)
Eps(2)
Eps(3)
Eta

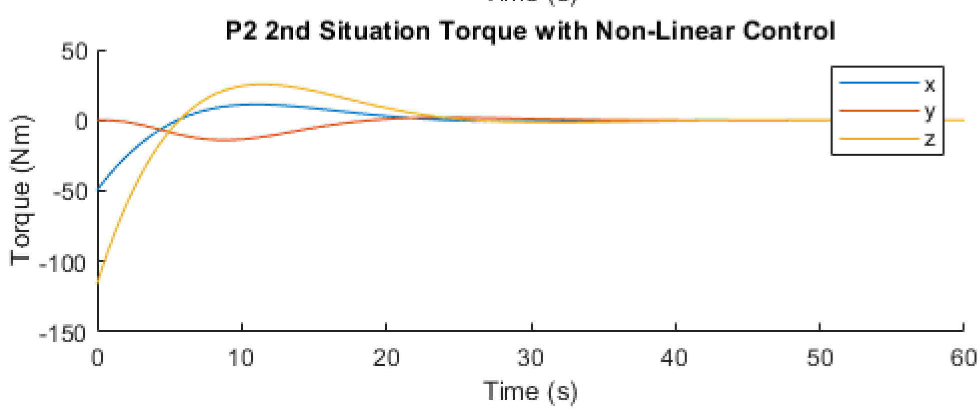

P2 2nd Situation Angular Velocity with Linear Control
Angular Velocity (rad/s)
Time (s)
x
y
z


P2 2nd Situation Angular Velocity with Non-Linear Control
Angular Velocity (rad/s)
Time (s)
x
y
z


P1 1st Situation Torque with Linear Control
Torque (N/m)
Time (s)
10
0
-10
-20
-30
0
10
20
30
40
50
60
x
y
z


P1 1st Situation Torque with Non-Linear Control
Torque (Nm)
Time (s)
10
0
-10
-20
-30
0
10
20
30
40
50
60
x
y
z


P2 2nd Situation Torque with Linear Control
Torque (N/m)
Time (s)
50
0
-50
-100
-150
0
10
20
30
40
50
60
x
y
z


P2 2nd Situation Torque with Non-Linear Control
Torque (Nm)
Time (s)
50
0
-50
-100
-150
0
10
20
30
40
50
60
x
y
z

Published with MATLAB® R2019a