
Table of Contents

Clear Workspace	1
Given Data	1
Maneuver 1	1
Maneuver 2	5

Clear Workspace

```
close all;
clear;
clc;
```

Given Data

```
m = 1637; % kg
g = 9.81; % m/s^2
Iz = 3326; % kg-m^2
l = 2.736; % m
t = 1.7; % m
a = 0.4*1; % 60% front load distribution
b = 0.6*1; % 60% front load distribution
G = 15;
h = 2.4/3.281; % m
```

Maneuver 1

```
% Simulation parameters
v = 74/2.237; % m/s
dt = 0.001; % s
X = [];
X(:,1) = [0;0];
Xdot = [];
Xint(:,1) = [0;0];
Bdot = 0;
V = [];
P = [0;0];
Sf = (m*g*0.6)/2; % 60% front (static) load distribution
Sr = (m*g*0.4)/2; % 60% front (static) load distribution
j = 1;

% Simulate from t=0 to t=7
U = 0; % Control input (@ 0 deg steering wheel angle)
for i = 0+dt:dt:7
    ay = v*(Bdot+X(2,j));
    FZfl = Sf - ((m*ay*h)/t)*0.6;
    FZfr = Sf + ((m*ay*h)/t)*0.6;
    FZrl = Sr - ((m*ay*h)/t)*0.4;
    FZrr = Sr + ((m*ay*h)/t)*0.4;
    af = U - (X(2,j)*a)/v - X(1,j);
```

```

    ar = (X(2,j)*b)/v - X(1,j);
    FYfl = -nonlintire(af,FZfl,v);
    FYfr = -nonlintire(af,FZfr,v);
    FYf = FYfl+FYfr;
    FYrl = -nonlintire(ar,FZrl,v);
    FYrr = -nonlintire(ar,FZrr,v);
    FYr = FYrl+FYrr;
    Xdot(1,j) = (FYf+FYr)/(m*v) - X(2,j);
    Xdot(2,j) = (FYf*a-FYr*b)/(Iz);
    X(:,j+1) = X(:,j) + Xdot(:,j)*dt;
    Xint(:,j+1) = Xint(:,j) + X(:,j)*dt;
    V(:,j) = [v*cos(X(1,j)+Xint(2,j)); v*sin(X(1,j)+Xint(2,j))];
    P(:,j+1) = P(:,j) + V(:,j)*dt;
    Bdot = Xdot(1,j);
    j = j+1;
end

% Simulate from t=7 to t=67
U = (45*(pi/180))/G; % Control input (@ 45 deg steering wheel angle)
for i = 7+dt:dt:67
    ay = v*(Bdot+X(2,j));
    FZfl = Sf - ((m*ay*h)/t)*0.6;
    FZfr = Sf + ((m*ay*h)/t)*0.6;
    FZrl = Sr - ((m*ay*h)/t)*0.4;
    FZrr = Sr + ((m*ay*h)/t)*0.4;
    af = U - (X(2,j)*a)/v - X(1,j);
    ar = (X(2,j)*b)/v - X(1,j);
    FYfl = -nonlintire(af,FZfl,v);
    FYfr = -nonlintire(af,FZfr,v);
    FYf = FYfl+FYfr;
    FYrl = -nonlintire(ar,FZrl,v);
    FYrr = -nonlintire(ar,FZrr,v);
    FYr = FYrl+FYrr;
    Xdot(1,j) = (FYf+FYr)/(m*v) - X(2,j);
    Xdot(2,j) = (FYf*a-FYr*b)/(Iz);
    X(:,j+1) = X(:,j) + Xdot(:,j)*dt;
    Xint(:,j+1) = Xint(:,j) + X(:,j)*dt;
    V(:,j) = [v*cos(X(1,j)+Xint(2,j)); v*sin(X(1,j)+Xint(2,j))];
    P(:,j+1) = P(:,j) + V(:,j)*dt;
    Bdot = Xdot(1,j);
    j = j+1;
end

% Simulate from t=67 to t=100
U = 0; % Control input (@ 0 deg steering wheel angle)
for i = 67+dt:dt:100
    ay = v*(Bdot+X(2,j));
    FZfl = Sf - ((m*ay*h)/t)*0.6;
    FZfr = Sf + ((m*ay*h)/t)*0.6;
    FZrl = Sr - ((m*ay*h)/t)*0.4;
    FZrr = Sr + ((m*ay*h)/t)*0.4;
    af = U - (X(2,j)*a)/v - X(1,j);
    ar = (X(2,j)*b)/v - X(1,j);
    FYfl = -nonlintire(af,FZfl,v);

```

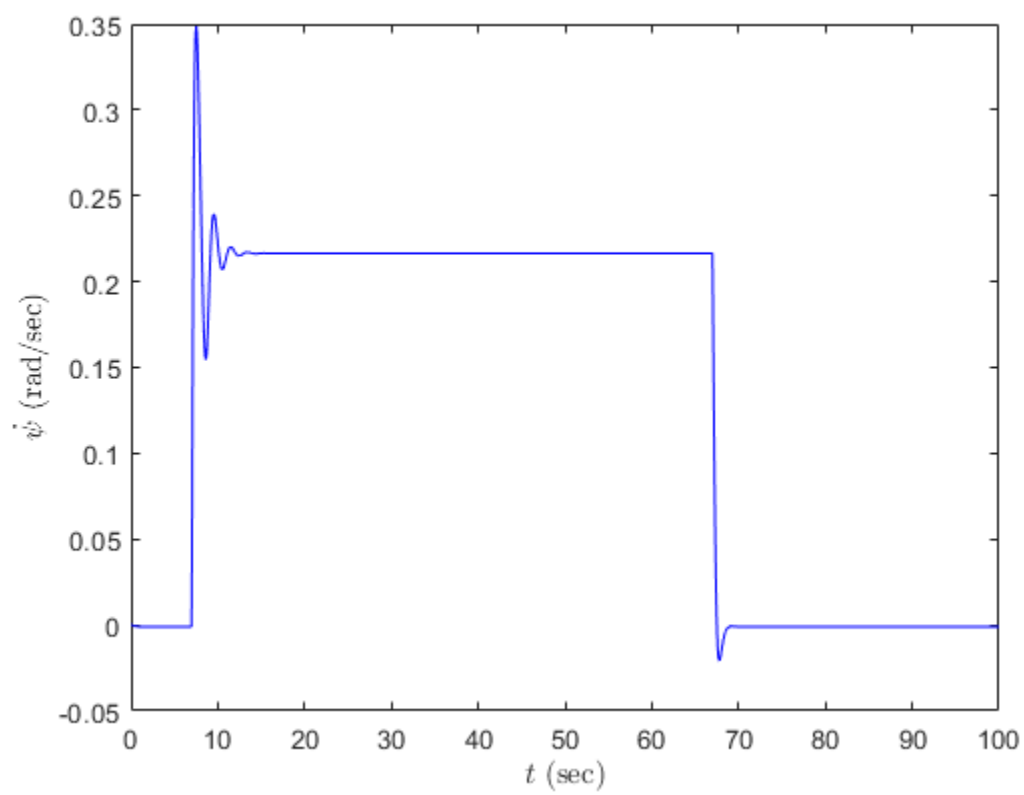
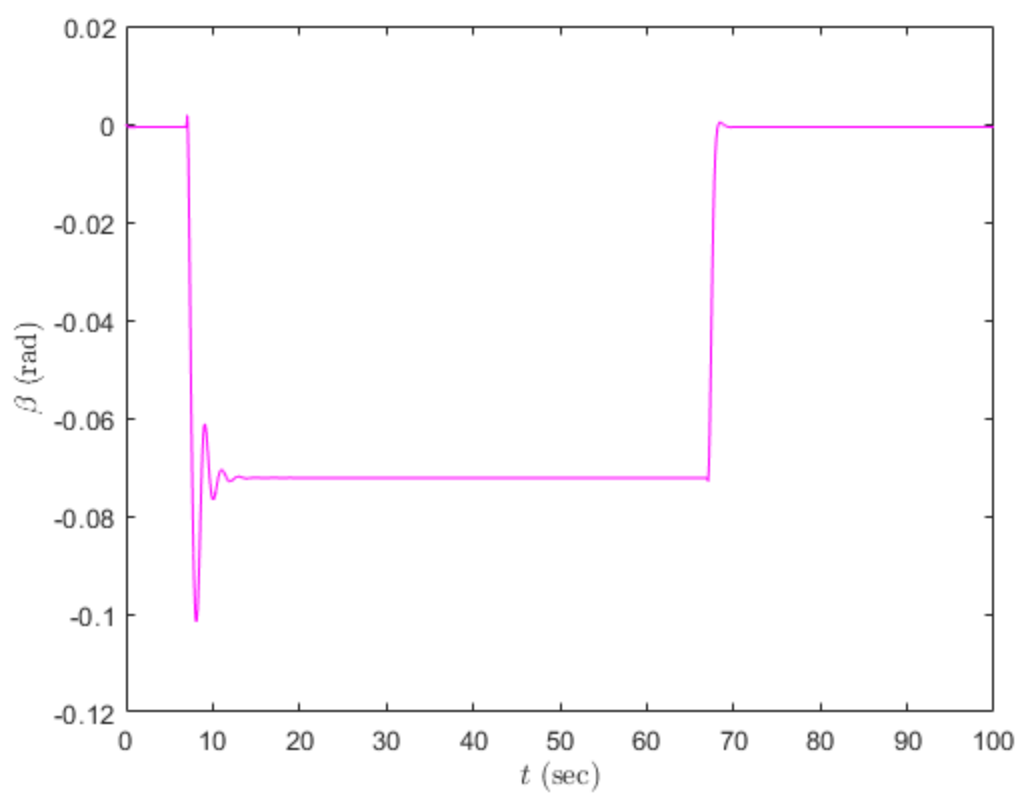
```

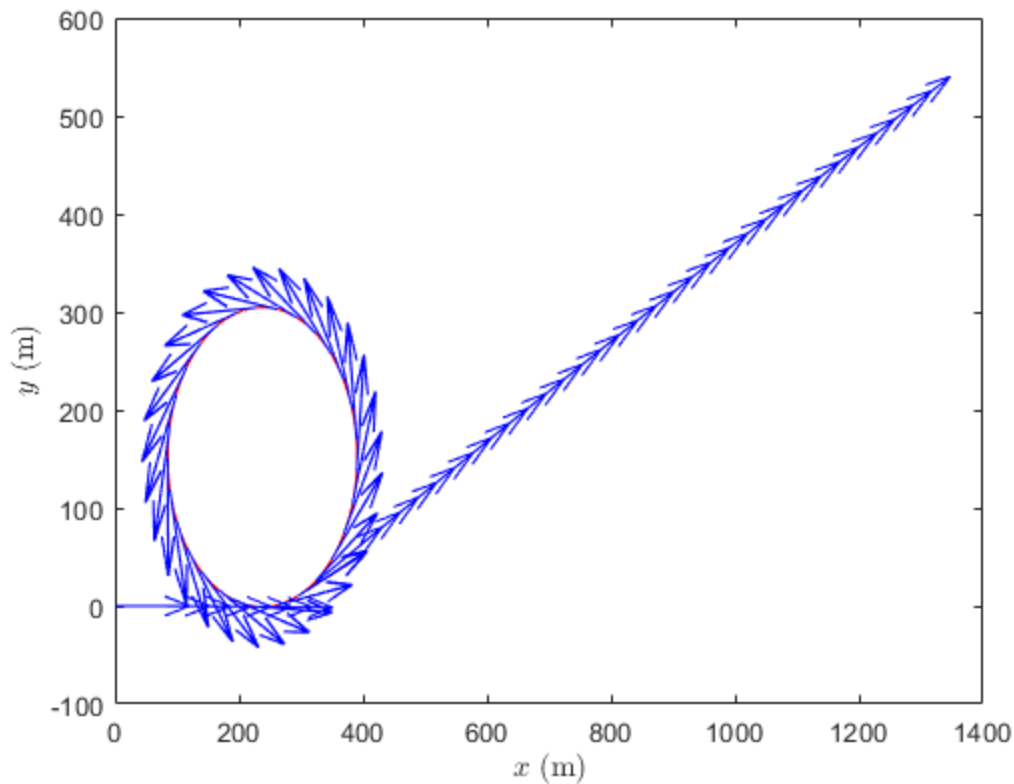
    FYfr = -nonlintire(af,FZfr,v);
    FYf = FYfl+FYfr;
    FYrl = -nonlintire(ar,FZrl,v);
    FYrr = -nonlintire(ar,FZrr,v);
    FYr = FYrl+FYrr;
    Xdot(1,j) = (FYf+FYr)/(m*v) - X(2,j);
    Xdot(2,j) = (FYf*a-FYr*b)/(Iz);
    X(:,j+1) = X(:,j) + Xdot(:,j)*dt;
    Xint(:,j+1) = Xint(:,j) + X(:,j)*dt;
    V(:,j) = [v*cos(X(1,j)+Xint(2,j)); v*sin(X(1,j)+Xint(2,j))];
    P(:,j+1) = P(:,j) + V(:,j)*dt;
    Bdot = Xdot(1,j);
    j = j+1;
end

% Plot states
figure(1)
plot((0:dt:100),X(1,:), 'color', 'magenta')
xlabel('$\{t\}$ (sec)', 'interpreter', 'latex')
ylabel('$\{\beta\}$ (rad)', 'interpreter', 'latex')
figure(2)
plot((0:dt:100),X(2,:), 'color', 'blue')
xlabel('$\{t\}$ (sec)', 'interpreter', 'latex')
ylabel('$\{\dot{\psi}\}$ (rad/sec)', 'interpreter', 'latex')

% Plot trajectory
PosX = P(1,:);
PosX_sec = PosX(1:1000:end-1000);
PosY = P(2,:);
PosY_sec = PosY(1:1000:end-1000);
VelX = V(1,:);
VelX_sec = VelX(1:1000:end);
VelY = V(2,:);
VelY_sec = VelY(1:1000:end);
figure(3)
plot(PosX,PosY, 'color', 'red')
%comet(PosX,PosY)
hold on
quiver(PosX_sec,PosY_sec,VelX_sec,VelY_sec, 'color', 'blue')
xlabel('$\{x\}$ (m)', 'interpreter', 'latex')
ylabel('$\{y\}$ (m)', 'interpreter', 'latex')

```





Maneuver 2

```
% Simulation parameters
v = 50/2.237; % m/s
dt = 0.001; % s
X = [];
X(:,1) = [0;0];
Xdot = [];
Xint(:,1) = [0;0];
Bdot = 0;
V = [];
P = [0;0];
Sf = (m*g*0.6)/2; % 60% front (static) load distribution
Sr = (m*g*0.4)/2; % 60% front (static) load distribution
j = 1;

% Simulate from t=0 to t=5
U = 0; % Control input (@ 0 deg steering wheel angle)
for i = 0+dt:dt:5
    ay = v*(Bdot+X(2,j));
    FZfl = Sf - ((m*ay*h)/t)*0.6;
    FZfr = Sf + ((m*ay*h)/t)*0.6;
    FZrl = Sr - ((m*ay*h)/t)*0.4;
    FZrr = Sr + ((m*ay*h)/t)*0.4;
    af = U - (X(2,j)*a)/v - X(1,j);
```

```

    ar = (X(2,j)*b)/v - X(1,j);
    FYfl = -nonlintire(af,FZfl,v);
    FYfr = -nonlintire(af,FZfr,v);
    FYf = FYfl+FYfr;
    FYrl = -nonlintire(ar,FZrl,v);
    FYrr = -nonlintire(ar,FZrr,v);
    FYr = FYrl+FYrr;
    Xdot(1,j) = (FYf+FYr)/(m*v) - X(2,j);
    Xdot(2,j) = (FYf*a-FYr*b)/(Iz);
    X(:,j+1) = X(:,j) + Xdot(:,j)*dt;
    Xint(:,j+1) = Xint(:,j) + X(:,j)*dt;
    V(:,j) = [v*cos(X(1,j)+Xint(2,j)); v*sin(X(1,j)+Xint(2,j))];
    P(:,j+1) = P(:,j) + V(:,j)*dt;
    Bdot = Xdot(1,j);
    j = j+1;
end

% Simulate from t=5 to t=23.621
U = 0:((14.5*(pi/180))/G)*dt:(270*(pi/180))/G; % Control input (@ 0-270 deg
steering wheel angle)
for i = 5+dt:dt:23.621
    ay = v*(Bdot+X(2,j));
    FZfl = Sf - ((m*ay*h)/t)*0.6;
    FZfr = Sf + ((m*ay*h)/t)*0.6;
    FZrl = Sr - ((m*ay*h)/t)*0.4;
    FZrr = Sr + ((m*ay*h)/t)*0.4;
    af = U(j-5000) - (X(2,j)*a)/v - X(1,j);
    ar = (X(2,j)*b)/v - X(1,j);
    FYfl = -nonlintire(af,FZfl,v);
    FYfr = -nonlintire(af,FZfr,v);
    FYf = FYfl+FYfr;
    FYrl = -nonlintire(ar,FZrl,v);
    FYrr = -nonlintire(ar,FZrr,v);
    FYr = FYrl+FYrr;
    Xdot(1,j) = (FYf+FYr)/(m*v) - X(2,j);
    Xdot(2,j) = (FYf*a-FYr*b)/(Iz);
    X(:,j+1) = X(:,j) + Xdot(:,j)*dt;
    Xint(:,j+1) = Xint(:,j) + X(:,j)*dt;
    V(:,j) = [v*cos(X(1,j)+Xint(2,j)); v*sin(X(1,j)+Xint(2,j))];
    P(:,j+1) = P(:,j) + V(:,j)*dt;
    Bdot = Xdot(1,j);
    j = j+1;
end

% Simulate from t=23.621 to t=27.621
U = (270*(pi/180))/G; % Control input (@ 270 deg steering wheel angle)
for i = 23.621+dt:dt:27.621
    ay = v*(Bdot+X(2,j));
    FZfl = Sf - ((m*ay*h)/t)*0.6;
    FZfr = Sf + ((m*ay*h)/t)*0.6;
    FZrl = Sr - ((m*ay*h)/t)*0.4;
    FZrr = Sr + ((m*ay*h)/t)*0.4;
    af = U - (X(2,j)*a)/v - X(1,j);
    ar = (X(2,j)*b)/v - X(1,j);

```

```

    FYfl = -nonlintire(af,FZfl,v);
    FYfr = -nonlintire(af,FZfr,v);
    FYf = FYfl+FYfr;
    FYrl = -nonlintire(ar,FZrl,v);
    FYrr = -nonlintire(ar,FZrr,v);
    FYr = FYrl+FYrr;
    Xdot(1,j) = (FYf+FYr)/(m*v) - X(2,j);
    Xdot(2,j) = (FYf*a-FYr*b)/(Iz);
    X(:,j+1) = X(:,j) + Xdot(:,j)*dt;
    Xint(:,j+1) = Xint(:,j) + X(:,j)*dt;
    V(:,j) = [v*cos(X(1,j)+Xint(2,j)); v*sin(X(1,j)+Xint(2,j))];
    P(:,j+1) = P(:,j) + V(:,j)*dt;
    Bdot = Xdot(1,j);
    j = j+1;
end

% Simulate from t=27.621 to t=40
U = 0; % Control input (@ 0 deg steering wheel angle)
for i = 27.621+dt:dt:40
    ay = v*(Bdot+X(2,j));
    FZfl = Sf - ((m*ay*h)/t)*0.6;
    FZfr = Sf + ((m*ay*h)/t)*0.6;
    FZrl = Sr - ((m*ay*h)/t)*0.4;
    FZrr = Sr + ((m*ay*h)/t)*0.4;
    af = U - (X(2,j)*a)/v - X(1,j);
    ar = (X(2,j)*b)/v - X(1,j);
    FYfl = -nonlintire(af,FZfl,v);
    FYfr = -nonlintire(af,FZfr,v);
    FYf = FYfl+FYfr;
    FYrl = -nonlintire(ar,FZrl,v);
    FYrr = -nonlintire(ar,FZrr,v);
    FYr = FYrl+FYrr;
    Xdot(1,j) = (FYf+FYr)/(m*v) - X(2,j);
    Xdot(2,j) = (FYf*a-FYr*b)/(Iz);
    X(:,j+1) = X(:,j) + Xdot(:,j)*dt;
    Xint(:,j+1) = Xint(:,j) + X(:,j)*dt;
    V(:,j) = [v*cos(X(1,j)+Xint(2,j)); v*sin(X(1,j)+Xint(2,j))];
    P(:,j+1) = P(:,j) + V(:,j)*dt;
    Bdot = Xdot(1,j);
    j = j+1;
end

% Plot states
figure(4)
plot((0:dt:40),X(1,:), 'color', 'magenta')
xlabel('$t$ (sec)', 'interpreter', 'latex')
ylabel('$\beta$ (rad)', 'interpreter', 'latex')
figure(5)
plot((0:dt:40),X(2,:), 'color', 'blue')
xlabel('$t$ (sec)', 'interpreter', 'latex')
ylabel('$\dot{\psi}$ (rad/sec)', 'interpreter', 'latex')

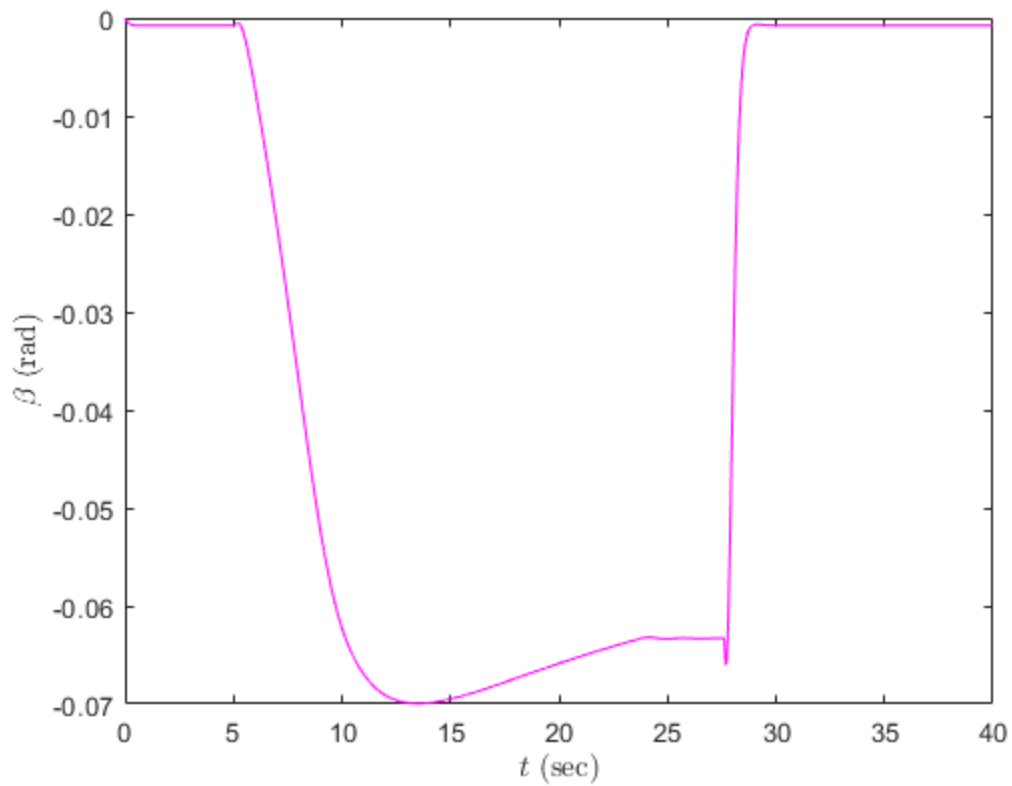
% Plot trajectory
PosX = P(1,:);

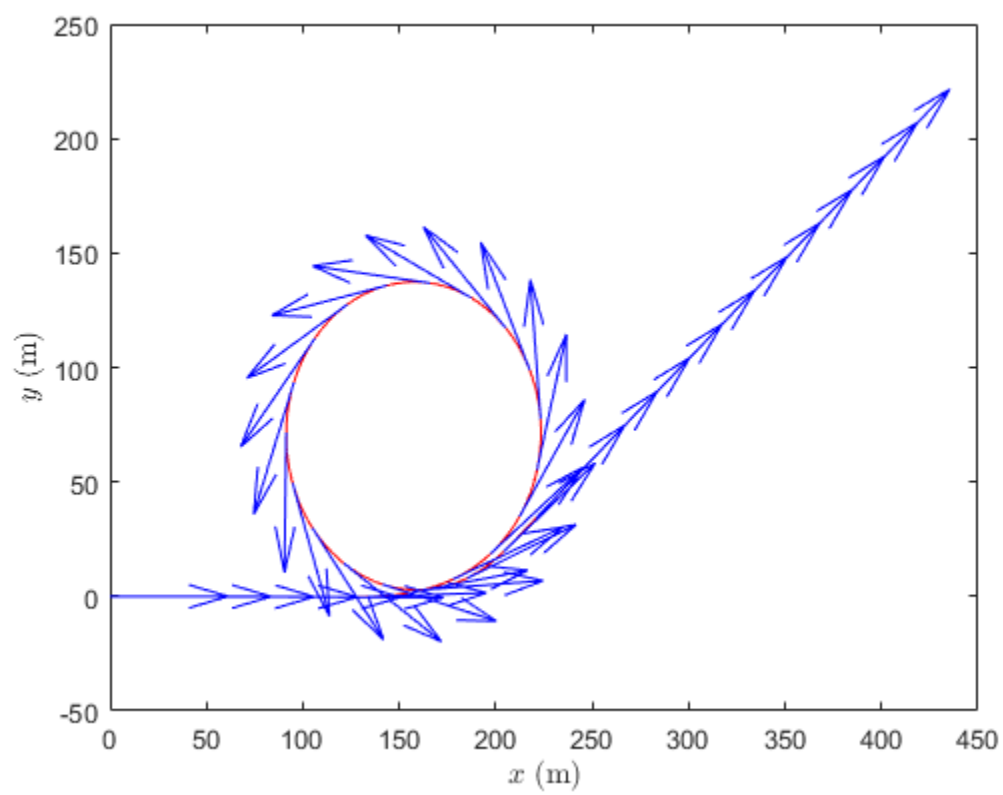
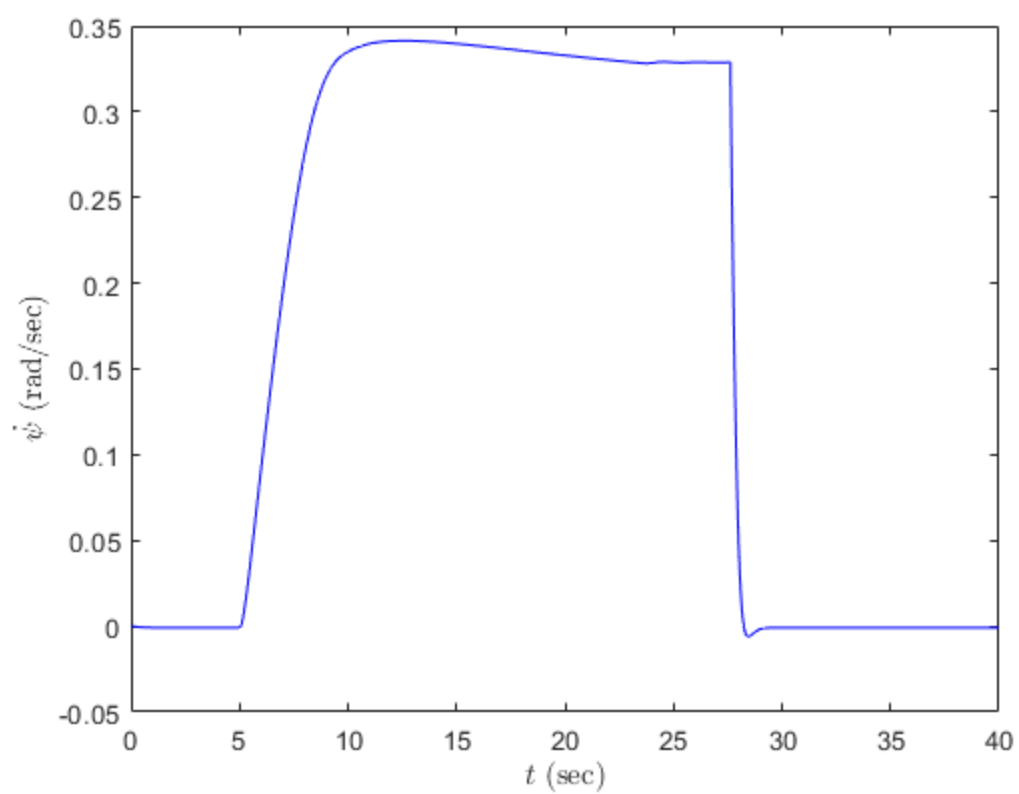
```

```

PosX_sec = PosX(1:1000:end-1000);
PosY = P(2,:);
PosY_sec = PosY(1:1000:end-1000);
VelX = V(1,:);
VelX_sec = VelX(1:1000:end);
VelY = V(2,:);
VelY_sec = VelY(1:1000:end);
figure(6)
plot(PosX,PosY,'color','red')
%comet(PosX,PosY)
hold on
quiver(PosX_sec,PosY_sec,VelX_sec,VelY_sec,'color','blue')
xlabel('$\{x\}$ (m)','interpreter','latex')
ylabel('$\{y\}$ (m)','interpreter','latex')

```





Published with MATLAB® R2022a