
Table of Contents

.....	1
EOM	2
Part a	2
Part b	3
Part c	4
Part d	5
Part e	7

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

```
% ASEN 5010 - HW 3, Problem 4  
% Spring 2025  
% Jash Bhalavat
```

```
mass = 1; % kg  
v_CN_P_0 = [0, 0, 0]'; % m/s  
v_CN_N_0 = v_CN_P_0;  
omega_PN_P_0 = [0, 0, 0.85]'; % rad/sec  
euler_PN_0 = [0, 0, 0]'; % rad  
g = 9.81; % km/s2  
I_c_P = [32.5, 32.5, 5]'; % kg-m2  
r_NozzleC_P = [0, 0, -3]'; % m  
r_NoseC_P = [0, 0, 4]'; % m
```

```
r_CN_N_0 = [0, 0, 0]'; % [m]
```

```
tspan = [0, 60]; % sec
```

```
function thrust_P = thrust(time)  
    % Thrust in Principal frame  
  
    thruster_misalignment = deg2rad(2.5); % rad  
    thruster_mag = 15; % N  
    thruster_duration = 10; % sec  
    thrust_parallel = 0; % [N] Passes CoM  
    thrust_perp = 0; % [N]  
    if time <= thruster_duration  
        thrust_parallel = thruster_mag * cos(thruster_misalignment);  
        thrust_perp = thruster_mag * sin(thruster_misalignment);  
    end  
    thrust_P = [0, -thrust_perp, thrust_parallel]'; % [N]  
end
```

```
function out = L_c_P(time, r)  
    % Torque at CoM in P frame  
    thrust_at_time_P = thrust(time);  
    out = cross(r, thrust_at_time_P); % [Nm]  
end
```

EOM

```
state0 = [r_CN_N_0; v_CN_N_0; omega_PN_P_0; euler_PN_0];

[tout_state, state_out] = ode45(@(t, state)state_eom(t, state, mass, g,
I_c_P, r_NozzleC_P), [tspan(1), tspan(2)], state0);

function state_dot = state_eom(time, state, mass, g, I, r_NozzleC_P)
    % state = [x_N, y_N, z_N, x_N_dot, y_N_dot, z_N_dot, omega_PN_P(1),
omega_PN_P(2),
    % omega_PN_P(3), alpha, beta, gamma]'
    % state_dot = [x_N_dot, y_N_dot, z_N_dot, x_N_dotdot, y_N_dotdot,
z_N_dotdot, omega_PN_P(1), omega_PN_P(2),
    % omega_PN_P(3), alpha, beta, gamma]'

    gravity_N = [0 0 -g]';

    r_CN_N = state(1:3);
    v_CN_N = state(4:6);
    omega_PN_P = state(7:9);
    euler_PN = state(10:12);

    r_CN_N_dot = v_CN_N;

    PN = R3(euler_PN(3))*R2(euler_PN(2))*R1(euler_PN(1));

    thrust_at_time_P = thrust(time);
    thrust_at_time_N = PN' * thrust_at_time_P;
    v_CN_N_dot = (thrust_at_time_N + mass*gravity_N)/mass;

    L_c_P_at_time = L_c_P(time, r_NozzleC_P);
    omega_PN_P_dot(1,1) = -1/I(1) * (I(3) - I(2)) *
omega_PN_P(2)*omega_PN_P(3) + L_c_P_at_time(1)/I(1);
    omega_PN_P_dot(2,1) = -1/I(2) * (I(1) - I(3)) *
omega_PN_P(1)*omega_PN_P(3) + L_c_P_at_time(2)/I(2);
    omega_PN_P_dot(3,1) = -1/I(3) * (I(2) - I(1)) *
omega_PN_P(1)*omega_PN_P(2) + L_c_P_at_time(3)/I(3);

    B_theta = 1/cos(euler_PN(2)) .* [cos(euler_PN(3)), -sin(euler_PN(3)), 0;
cos(euler_PN(2))*sin(euler_PN(3)),
cos(euler_PN(2))*cos(euler_PN(3)), 0;
-sin(euler_PN(2))*cos(euler_PN(3)),
sin(euler_PN(2))*sin(euler_PN(3)), cos(euler_PN(2))];
    euler_PN_dot = B_theta * omega_PN_P;

    state_dot = [r_CN_N_dot; v_CN_N_dot; omega_PN_P_dot; euler_PN_dot];
end
```

Part a

```
omega_PN_P_out = state_out(:,7:9);
```

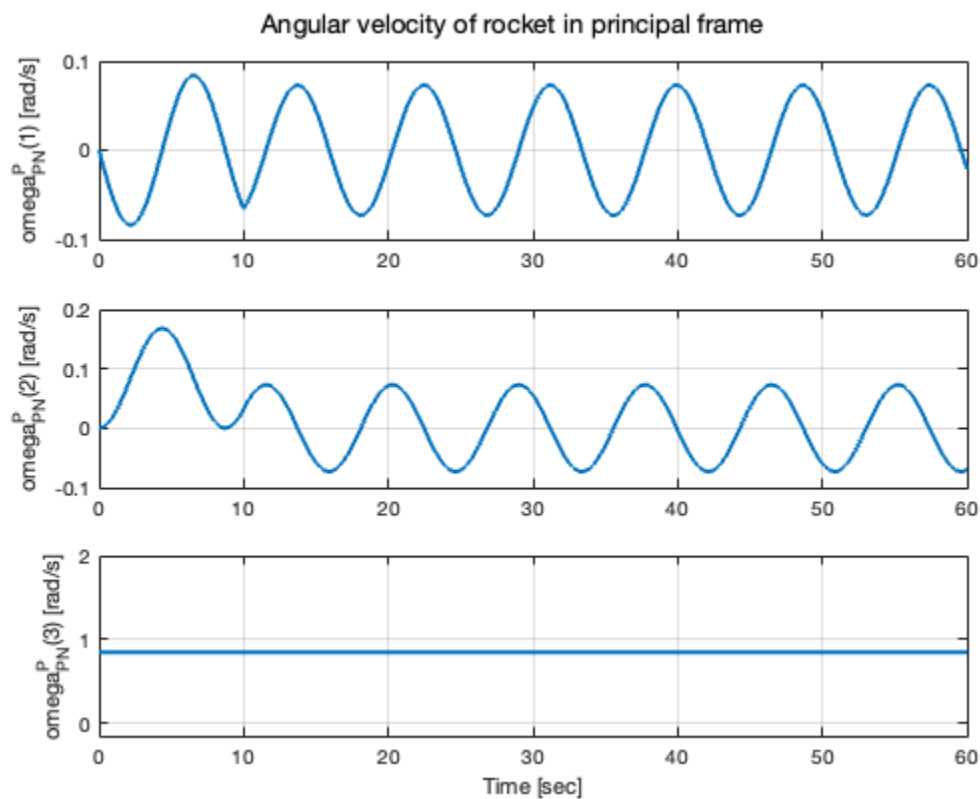
```

figure()
subplot(3,1,1)
plot(tout_state, omega_PN_P_out(:,1), 'LineWidth',2)
ylabel("omega_{PN}^P(1) [rad/s]")
grid on

subplot(3,1,2)
plot(tout_state, omega_PN_P_out(:,2), 'LineWidth',2)
ylabel("omega_{PN}^P(2) [rad/s]")
grid on

subplot(3,1,3)
plot(tout_state, omega_PN_P_out(:,3), 'LineWidth',2)
ylabel("omega_{PN}^P(3) [rad/s]")
xlabel("Time [sec]")
grid on
sgtitle("Angular velocity of rocket in principal frame")

```



Part b

```

v_CN_N_out = state_out(:, 4:6);

for i = 1:length(tout_state)
    euler_PN = state_out(i, 10:12);
    PN = R3(euler_PN(3))*R2(euler_PN(2))*R1(euler_PN(1));
    v_CN_P_out(i,:) = (PN * v_CN_N_out(i, :))';

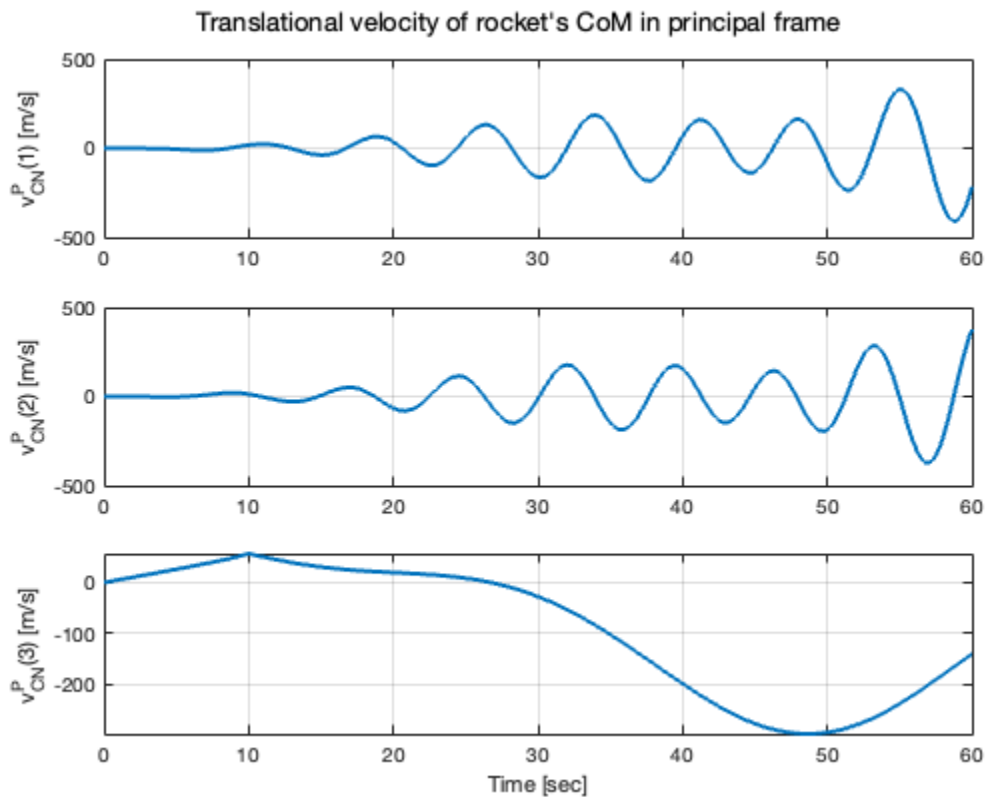
```

```
end
```

```
figure()
subplot(3,1,1)
plot(tout_state, v_CN_P_out(:,1), 'LineWidth',2)
ylabel("v_{CN}^P(1) [m/s]")
grid on

subplot(3,1,2)
plot(tout_state, v_CN_P_out(:,2), 'LineWidth',2)
ylabel("v_{CN}^P(2) [m/s]")
grid on

subplot(3,1,3)
plot(tout_state, v_CN_P_out(:,3), 'LineWidth',2)
ylabel("v_{CN}^P(3) [m/s]")
xlabel("Time [sec]")
grid on
sgtitle("Translational velocity of rocket's CoM in principal frame")
```



Part c

```
euler_PN_out = state_out(:, 10:12);

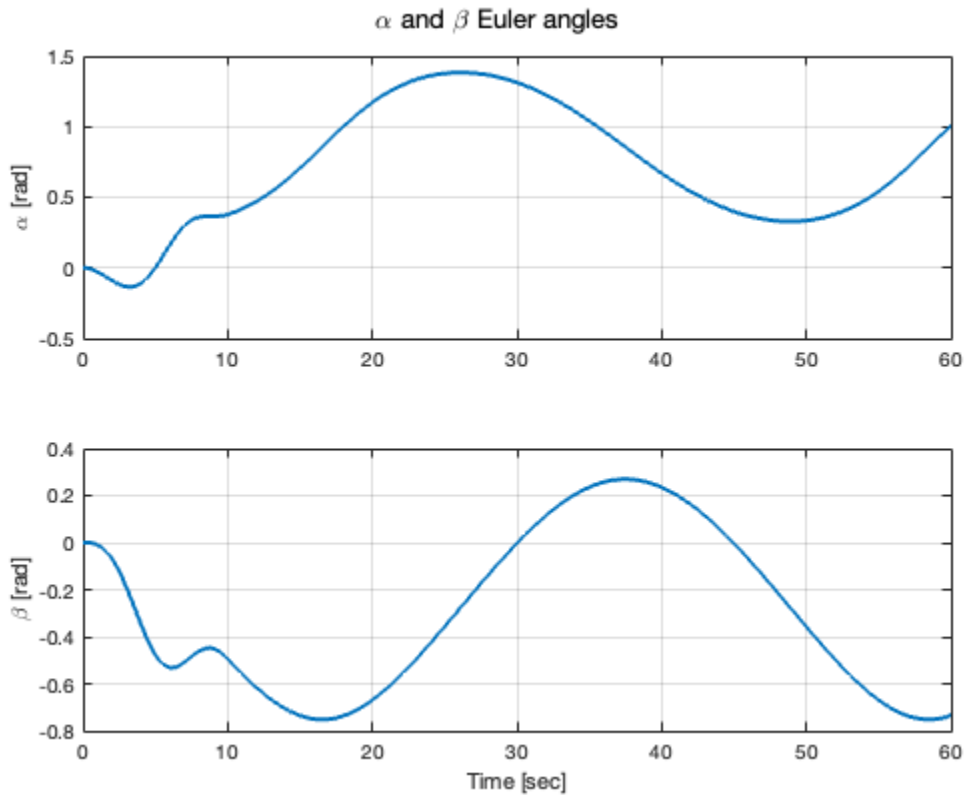
figure()
subplot(2,1,1)
```

```

plot(tout_state, euler_PN_out(:,1), 'LineWidth',2)
ylabel("\alpha [rad]")
grid on

subplot(2,1,2)
plot(tout_state, euler_PN_out(:,2), 'LineWidth',2)
xlabel("Time [sec]")
ylabel("\beta [rad]")
grid on
sgtitle("\alpha and \beta Euler angles")

```



Part d

```

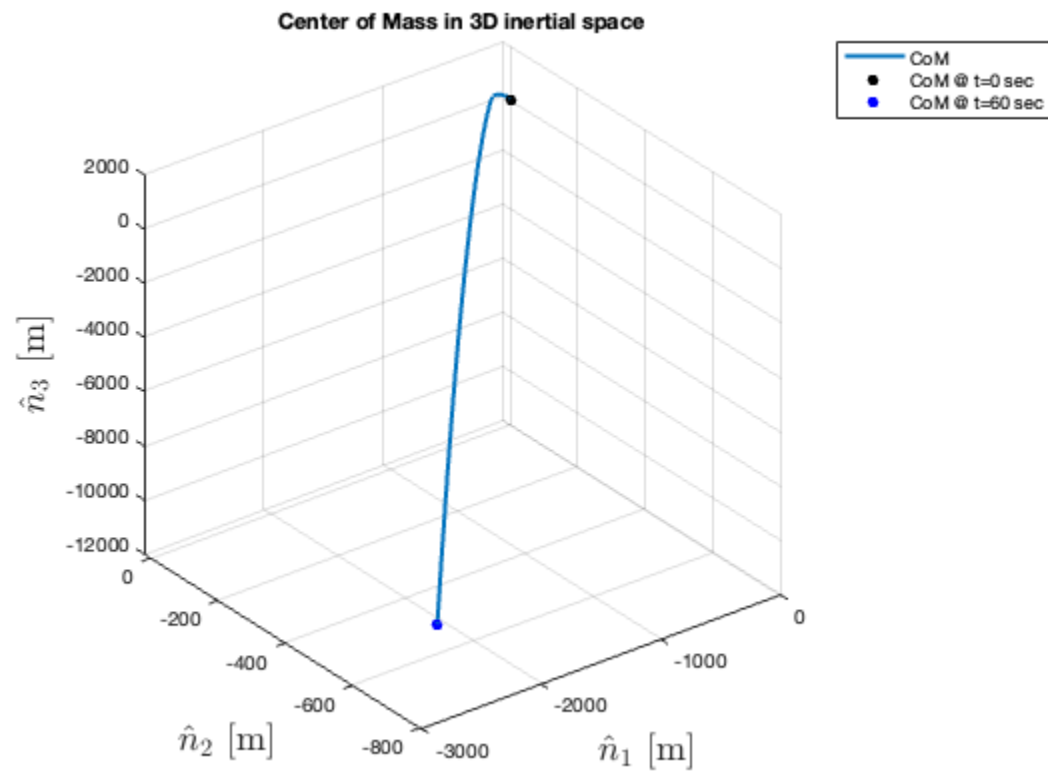
figure()
plot3(state_out(:,1), state_out(:,2), state_out(:,3), 'LineWidth',2)
xlabel('$$\hat{n}_1$$ [m]', 'Interpreter', 'Latex', 'FontSize',18)
ylabel('$$\hat{n}_2$$ [m]', 'Interpreter', 'Latex', 'FontSize',18)
zlabel('$$\hat{n}_3$$ [m]', 'Interpreter', 'Latex', 'FontSize',18)
hold on
scatter3(state_out(1,1), state_out(1,2), state_out(1,3), 'filled', 'black')
scatter3(state_out(end,1), state_out(end,2), state_out(end,3), 'filled',
'blue')
legend("CoM", "CoM @ t=0 sec", "CoM @ t=60 sec")
grid on
title("Center of Mass in 3D inertial space")

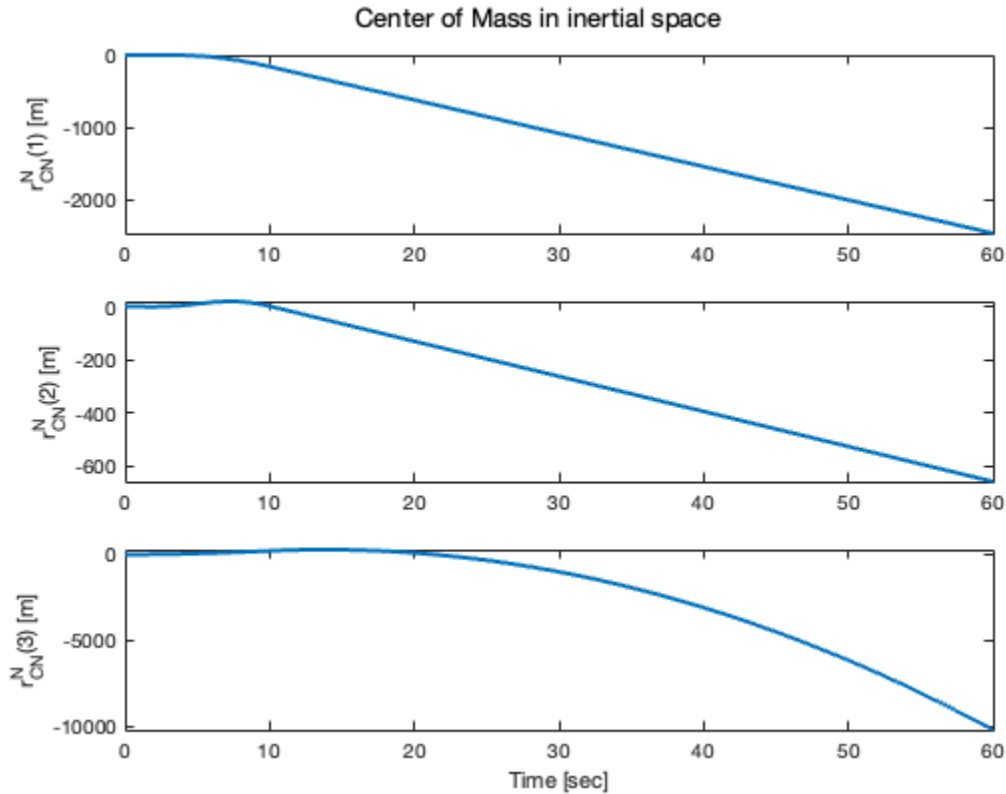
```

```

figure()
subplot(3,1,1)
plot(tout_state, state_out(:,1), 'LineWidth',2)
ylabel("r_{CN}^N(1) [m]")
subplot(3,1,2)
plot(tout_state, state_out(:,2), 'LineWidth',2)
ylabel("r_{CN}^N(2) [m]")
subplot(3,1,3)
plot(tout_state, state_out(:,3), 'LineWidth',2)
ylabel("r_{CN}^N(3) [m]")
xlabel("Time [sec]")
sgtitle("Center of Mass in inertial space")

```



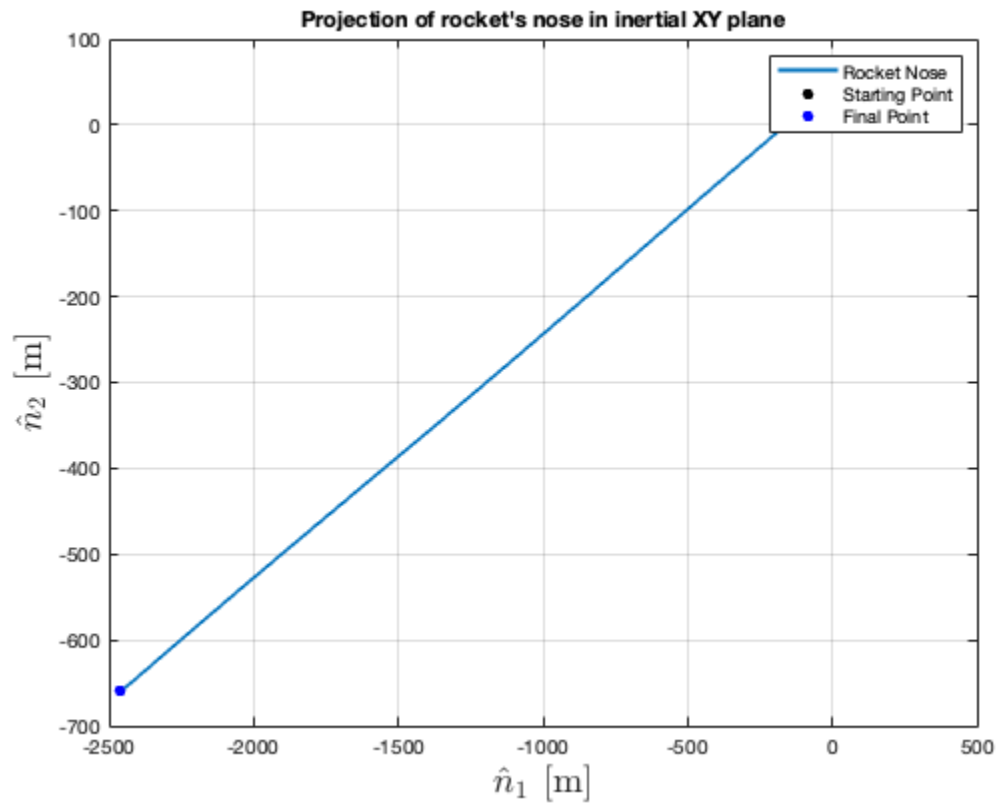


Part e

```
euler_PN = state_out(:,10:12);
r_CN_N = state_out(:,1:3);

for i = 1:length(tout_state)
    PN = R3(euler_PN(i, 3))*R2(euler_PN(i, 2))*R1(euler_PN(i, 1));
    r_NoseN_N(i,:) = (r_CN_N(i,:) + PN * r_NoseC_P)';
end

figure()
plot(r_NoseN_N(:,1), r_NoseN_N(:,2), 'LineWidth',2)
hold on
scatter(r_NoseN_N(1,1), r_NoseN_N(1,2), 'filled', 'black')
scatter(r_NoseN_N(end,1), r_NoseN_N(end,2), 'filled', 'blue')
legend("Rocket Nose", "Starting Point", "Final Point")
xlabel('$$\hat{n}_{1}$$ [m]','Interpreter','Latex', 'FontSize',18)
ylabel('$$\hat{n}_{2}$$ [m]','Interpreter','Latex', 'FontSize',18)
title("Projection of rocket's nose in inertial XY plane")
grid on
```



Published with MATLAB® R2024a