Table of Contents

Gagandeep Thapar; AERO 560 HW1B

```
% housekeeping
clc;
clearvars;
close all;
```

Problem 2

```
% givens
body.omega = [0.1, 0.1, 0.1];   % [rad/s]

orbit.h = 55759;    % [kg/m2] angular momentum
orbit.ecc = 0.001;  % [~] eccentricity
orbit.raan = 10;    % [deg] raan
orbit.inc = 42;     % [deg] inc
orbit.omega = 22;   % [deg] arg of perigee
orbit.theta = 0;    % [deg] true anomaly
orbit.mu = 398600;

sat.mass = 3;   % [kg]
sat.l = 0.1;    % [m] length dimension
sat.w = 0.1;    % [m] width dimension
sat.h = 0.3;    % [m] height dimension

% initial conditions
[orbit.R, orbit.V] = COES2STATE(orbit.h, orbit.ecc, orbit.inc, orbit.raan,
 orbit.omega, orbit.theta, 398600);
orbit.state0 = [orbit.R;orbit.V];

sat.J = sat.mass/12 * [sat.l^2 + sat.h^2, 0, 0;
                       0, sat.w^2 + sat.h^2, 0;
                       0, 0, sat.l^2 + sat.w^2];    % principal inertial
 matrix

sat.w0 = [0.1;0.1;0.1]; % initial ang vel
sat.e0 = [0;0;0];   % initial quat
sat.n0 = 1;
sat.euls0 = quat2eul([sat.n0, sat.e0'])'; % initial euler

sat.state0 = [sat.w0;sat.e0;sat.n0;sat.euls0];  % initial state

% initial lvlh_eci conditions
orbit.lvlh_q0 = lvlh_eci(orbit.state0);
```

```
orbit.lvlh_eul0 = quat_eul(orbit.lvlh_q0);
r = orbit.state0(1:3);
v = orbit.state0(4:6);
orbit.lvlh_w0 = cross(r,v) / norm(r)^2;

orbit.lvlh_state0 = [orbit.lvlh_w0;orbit.lvlh_q0;orbit.lvlh_eul0];

% initial body_eci conditions
sat.q_eci0 = body_eci([sat.e0;sat.n0], orbit.lvlh_q0);
sat.e_eci0 = sat.q_eci0(1:3);
sat.n_eci0 = sat.q_eci0(4);
sat.eul_eci0 = quat_eul(sat.q_eci0);

sat.eci_state0 = [sat.q_eci0;sat.eul_eci0];

% call simulation and unpack data
out = sim('hw1b');
t = out.tout;

sat.state = squeeze(out.sat_state)';
sat.eci_state = squeeze(out.body_eci_state)';
orbit.state = squeeze(out.orbit_state)';
orbit.lvlh_state = squeeze(out.lvlh_eci_state)';

orbit.R = orbit.state(:,1:3);
orbit.V = orbit.state(:,4:6);
orbit.lvlh_w = orbit.lvlh_state(:,1:3);
orbit.lvlh_q = orbit.lvlh_state(:,4:6);
orbit.lvlh_n = orbit.lvlh_state(:,7);
orbit.lvlh_eul = orbit.lvlh_state(:,8:10);

sat.w = sat.state(:,1:3);
sat.e = sat.state(:,4:6);
sat.n = sat.state(:,7);
sat.euls = sat.state(:,8:10);
sat.eci_e = sat.eci_state(:,1:3);
sat.eci_n = sat.eci_state(:,4);
sat.eci_euls = sat.eci_state(:,5:7);

% plot figures
figure
subplot(3,3,1)
hold on
plot(t, sat.e(:,1))
plot(t, sat.e(:,2))
plot(t, sat.e(:,3))
plot(t, sat.n)
hold off
title('Quaternion from Body to LVLH in Body')
legend('\epsilon_1', '\epsilon_2', '\epsilon_3', '\eta')
xlabel('Time [sec]')
ylabel('Value')

subplot(3,3,2)
```

```
hold on
plot(t, orbit.lvlh_q(:,1))
plot(t, orbit.lvlh_q(:,2))
plot(t, orbit.lvlh_q(:,3))
plot(t, orbit.lvlh_n)
hold off
title('Quaternion from LVLH to ECI in LVLH')
legend('\epsilon_1', '\epsilon_2', '\epsilon_3', '\eta')
xlabel('Time [sec]')
ylabel('Value')

subplot(3,3,3)
hold on
plot(t, sat.eci_e(:,1))
plot(t, sat.eci_e(:,2))
plot(t, sat.eci_e(:,3))
plot(t, sat.eci_n)
hold off
title('Quaternion from Body to ECI in Body')
legend('\epsilon_1', '\epsilon_2', '\epsilon_3', '\eta')
xlabel('Time [sec]')
ylabel('Value [~]')

subplot(3,3,4)
hold on
plot(t, sat.euls(:,1))
plot(t, sat.euls(:,2))
plot(t, sat.euls(:,3))
hold off
title('Euler Angles from Body to LVLH in Body')
legend('\Phi', '\Theta', '\Psi')
ylabel('Angle [rad]')
xlabel('Time [sec]')

subplot(3,3,5)
hold on
plot(t, orbit.lvlh_eul(:,1))
plot(t, orbit.lvlh_eul(:,2))
plot(t, orbit.lvlh_eul(:,3))
hold off
title('Euler Angles from LVLH to ECI in LVLH')
legend('\Phi', '\Theta', '\Psi')
ylabel('Angle [rad]')
xlabel('Time [sec]')

subplot(3,3,6)
hold on
plot(t, sat.eci_euls(:,1))
plot(t, sat.eci_euls(:,2))
plot(t, sat.eci_euls(:,3))
hold off
title('Euler Angles from Body to ECI in Body')
legend('\Phi', '\Theta', '\Psi')
ylabel('Angle [rad]')
```

```
xlabel('Time [sec]')

subplot(3,3,7)
hold on
plot(t, sat.w(:,1))
plot(t, sat.w(:,2))
plot(t, sat.w(:,3))
hold off
title('Angular Rates from Body to LVLH in Body')
legend('\omega_x', '\omega_y', '\omega_z')
ylabel('Rate [rad/sec]')
xlabel('Time [sec]')

subplot(3,3,8)
hold on
plot(t, orbit.lvlh_w(:,1))
plot(t, orbit.lvlh_w(:,2))
plot(t, orbit.lvlh_w(:,3))
hold off
title('Angular Rates from LVLH to ECI in LVLH')
legend('\omega_x', '\omega_y', '\omega_z')
ylabel('Rate [rad/sec]')
xlabel('Time [sec]')

subplot(3,3,9)
hold on
plot(t, sat.w(:,1))
plot(t, sat.w(:,2))
plot(t, sat.w(:,3))
hold off
title('Angular Rates from Body to ECI in Body')
legend('\omega_x', '\omega_y', '\omega_z')
ylabel('Rate [rad/sec]')
xlabel('Time [sec]')

% plot figures
figure
hold on
plot(t, sat.e(:,1))
plot(t, sat.e(:,2))
plot(t, sat.e(:,3))
plot(t, sat.n)
hold off
title('Quaternion from Body to LVLH in Body')
legend('\epsilon_1', '\epsilon_2', '\epsilon_3', '\eta')
xlabel('Time [sec]')
ylabel('Value')

figure
hold on
plot(t, orbit.lvlh_q(:,1))
plot(t, orbit.lvlh_q(:,2))
plot(t, orbit.lvlh_q(:,3))
plot(t, orbit.lvlh_n)
```

```
hold off
title('Quaternion from LVLH to ECI in LVLH')
legend('\epsilon_1', '\epsilon_2', '\epsilon_3', '\eta')
xlabel('Time [sec]')
ylabel('Value')

figure
hold on
plot(t, sat.eci_e(:,1))
plot(t, sat.eci_e(:,2))
plot(t, sat.eci_e(:,3))
plot(t, sat.eci_n)
hold off
title('Quaternion from Body to ECI in Body')
legend('\epsilon_1', '\epsilon_2', '\epsilon_3', '\eta')
xlabel('Time [sec]')
ylabel('Value [~]')

figure
hold on
plot(t, sat.euls(:,1))
plot(t, sat.euls(:,2))
plot(t, sat.euls(:,3))
hold off
title('Euler Angles from Body to LVLH in Body')
legend('\Phi', '\Theta', '\Psi')
ylabel('Angle [rad]')
xlabel('Time [sec]')

figure
hold on
plot(t, orbit.lvlh_eul(:,1))
plot(t, orbit.lvlh_eul(:,2))
plot(t, orbit.lvlh_eul(:,3))
hold off
title('Euler Angles from LVLH to ECI in LVLH')
legend('\Phi', '\Theta', '\Psi')
ylabel('Angle [rad]')
xlabel('Time [sec]')

figure
hold on
plot(t, sat.eci_euls(:,1))
plot(t, sat.eci_euls(:,2))
plot(t, sat.eci_euls(:,3))
hold off
title('Euler Angles from Body to ECI in Body')
legend('\Phi', '\Theta', '\Psi')
ylabel('Angle [rad]')
xlabel('Time [sec]')

figure
hold on
plot(t, sat.w(:,1))
```

```
plot(t, sat.w(:,2))
plot(t, sat.w(:,3))
hold off
title('Angular Rates from Body to LVLH in Body')
legend('\omega_x', '\omega_y', '\omega_z')
ylabel('Rate [rad/sec]')
xlabel('Time [sec]')

figure
hold on
plot(t, orbit.lvlh_w(:,1))
plot(t, orbit.lvlh_w(:,2))
plot(t, orbit.lvlh_w(:,3))
hold off
title('Angular Rates from LVLH to ECI in LVLH')
legend('\omega_x', '\omega_y', '\omega_z')
ylabel('Rate [rad/sec]')
xlabel('Time [sec]')

figure
hold on
plot(t, sat.w(:,1))
plot(t, sat.w(:,2))
plot(t, sat.w(:,3))
hold off
title('Angular Rates from Body to ECI in Body')
legend('\omega_x', '\omega_y', '\omega_z')
ylabel('Rate [rad/sec]')
xlabel('Time [sec]')
```

answering questions

```
fprintf('~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~\n')
fprintf('2. Initial Quaternion of Body to ECI [e, n]:\n\t[%.3f %.3f %.3f
 %.3f]\n\n', sat.eci_state(1,1:4));
fprintf('3. Initial Quaternion of Body to LVLH [e, n]:\n\t[%.3f %.3f %.3f
 %.3f]\n\n', sat.state(1,4:7))
fprintf('5. Nadir Direction after 200sec [deg]:\n\t%.3f\n', 360-
rad2deg(sat.eci_euls(end,3)))

function q = body_eci(body_lvlh_q,lvlh_eci_q)

    function qp = quatmult(q, p)

        function wx = skewSymmetric(w)
            wx = [0, -1*w(3), w(2);
                  w(3), 0, -1*w(1);
                  -1*w(2), w(1), 0];
        end

        qn = q(4);
        qe = q(1:3);
```

```
        pn = p(4);
        pe = p(1:3);

        n = pn * qn - pe'*qe;
        e = pn * qe + qn*pe + skewSymmetric(pe)*qe;

        qp = [e(1);e(2);e(3);n];

    end

    q = quatmult(body_lvlh_q, lvlh_eci_q);


end

function q = lvlh_eci(state)

    function Q = ECILVLH(R,V)

        if size(R) == [1 3]
            R = R';
        end

        if size(V) == [1 3]
            V = V';
        end

        z = -1*R / norm(R);
        y = -1 * cross(R,V) / norm(cross(R,V));

        x = cross(y,z);

        Q = [x,y,z]';

    end

    function [e, n] = c2quat(C)

        e = zeros(3,1);
        n = 0.5 * sqrt(1 + trace(C));
        e(1) = 0.25 * (C(2,3) - C(3,2))/n;
        e(2) = 0.25 * (C(3,1) - C(1,3))/n;
        e(3) = 0.25 * (C(1,2) - C(2,1))/n;

    end

    R = state(1:3);
    V = state(4:6);

    Q = ECILVLH(R,V);
    [e,n] = c2quat(Q);

    e = -1*e;
    n = -1*n;
```

```
    q = [e(1);e(2);e(3);n];

end

function eul = quat_eul(q)

    n = q(4);
    e = q(1:3);

    q = [n, e(1), e(2), e(3)];

    phi = atan2(2*(q(1)*q(2) + q(3)*q(4)), 1 - 2*(q(2)^2 + q(3)^2));
    theta = asin(2*(q(1)*q(3) - q(4)*q(2)));
    psi = atan2(2*(q(1)*q(4) + q(2)*q(3)), 1-2*(q(3)^2 + q(4)^2));

    eul = [phi; theta; psi];

end
```

Published with MATLAB® R2022b