```
clear; close all; clc;
problem1()
problem2()
problem3()
```

Problems

```
function problem1()
fprintf('Problem 1:\n')
    a = 8378;
    Re = 6378.1;
    mu = 398600;
    Js = 1371;

    Te = 1938.1811130; % first long eclipse time in seconds
    [T, ecl_time, ecl_proportion] = eclipse(a, mu, Re);
    fprintf("The eclipse time from GMAT is %f minutes\n", Te/60)
    ecl_proportion = Te/T;
    Td = T - Te; % daylight time per orbit



    % Solar cells need to generate operating power plus excess to store for
    % eclipse operations
    x = [10, 25, 50, 75];
    for ii = 1:4
        Pd = 50; % daylight operating power
        Pe = x(ii)*Pd; % eclipse operating power
        eta_charge = 98; % charging efficiency Table 21-18 SME
        DOD = .8; % use 80 percent of energy of the battery every eclipse
        energy_battery = Pe*Te/(eta_charge*DOD); % energy stored in battery
        energy_battery_wh(ii) = energy_battery/3600; % stored energy in watt-
hours
        mass_battery(ii) = energy_battery_wh(ii)/200;
        Pnom = Pd + energy_battery/Td;
        eta_temperature = .9;% assume 90 percent efficiency due to temperature
        eta_rad = .95; % assume 95 percent efficiency due to radiation
        eta_angle = 1; % assume straight on light
        Psa = Pnom/(eta_angle*eta_rad*eta_temperature); % Power needed from
 solar array
        Acell = .04*.06; % solar cell area in m
        eta_pack = .9; % assume 90 percent packing efficiency
        eta_sa = .185; % assume GaAs (SJ) solar arrays
        Asa(ii) = Psa/(eta_pack*eta_sa*Js); % area of solar array
        Ncell(ii) = ceil(Asa(ii)/Acell) ; % number of cells needed
        msa(ii) = 2.7*Asa(ii); % 2.7 kg/m^2
        fprintf("For eclipse operations of %f percent of nominal power: \n",
 x(ii))
        fprintf("\tA solar array array of %f m^2 is required\n", Asa(ii))
        fprintf("\tA solar array array of %f kg is required\n", msa(ii))
        fprintf("\tA solar array array of %f cells are required\n", Ncell(ii))
        fprintf("\tA battery of %f W-hr is required\n", energy_battery_wh(ii))
```

```
        fprintf("\tA battery of %f kg is required\n", mass_battery(ii))
    end

    figure
    plot(x,Asa,'-o')
    xlabel('Percent of Operating Power Used During Eclipse')
    ylabel('Size of Solar Array [m^2]')

    figure
    plot(x,Ncell,'-o')
    xlabel('Percent of Operating Power Used During Eclipse')
    ylabel('Number of Solar Cells')

    figure
    plot(x,energy_battery_wh,'-o')
    xlabel('Percent of Operating Power Used During Eclipse')
    ylabel('Energy of Battery [W-hr]')

    figure
    plot(x,msa,'-o',x,mass_battery,'-o')
    xlabel('Percent of Operating Power Used During Eclipse')
    ylabel('Mass [kg]')
    legend('Solar Array', 'Battery', Location='East')
end

function problem2()
fprintf('Problem 2:\n')
    Ip = 2000; % moment of inertia about principal axis, kg*m^2
    t_s = 5; % 5 second
    damping(1) = 0.65;
    damping(2) = 0.80;

    wn(1) = 4/(t_s*damping(1));
    wn(2) = 4/(t_s*damping(2));

    Kp(1) = Ip*wn(1)^2;
    Kp(2) = Ip*wn(2)^2;
    Kd(1) = 2*Ip*wn(1)*damping(2);
    Kd(2) = 2*Ip*wn(1)*damping(2);

    opts = odeset('RelTol', 1e-4, 'AbsTol', 1e-4);
    tspan = [0, 10];
    state0 = [0; 0];
    parameters = [Ip, Kp(1), Kd(1), deg2rad(30)];
    [t, state] = ode45(@single_axis_control, tspan, state0, opts, parameters);

    fprintf('For Maximum Damping: \n')
    fprintf('\tProportional gain, Kp = %f\n', Kp(1))
    fprintf('\tDerivative gain, Kd = %f\n', Kd(1))
    fprintf('For Minimum Damping: \n')
    fprintf('\tProportional gain, Kp = %f\n', Kp(2))
    fprintf('\tDerivative gain, Kd = %f\n', Kd(2))

    figure
```

```
    subplot(2,1,1)
    plot(t, rad2deg(state(:,1)))
    xlabel('Time [s]')
    ylabel('Yaw Angle [degree]')
    title('Minimum Damping ')
    subplot(2,1,2)
    plot(t, rad2deg(state(:,2)))
    xlabel('Time [s]')
    ylabel('Yaw Rate [degree/s]')

    parameters = [Ip, Kp(2), Kd(2), deg2rad(30)];
    [t, state] = ode45(@single_axis_control, tspan, state0, opts, parameters);
    figure

    subplot(2,1,1)
    plot(t, rad2deg(state(:,1)))
    xlabel('Time [s]')
    ylabel('Yaw Angle [degree]')
    title('Maximum Damping ')
    subplot(2,1,2)
    plot(t, rad2deg(state(:,2)))
    xlabel('Time [s]')
    ylabel('Yaw Rate [degree/s]')
end

function problem3()
    fprintf('Problem 3:\n')
    Isat = 500;
    Iwheel = .06;
    Km = .04; % Torque constant
    Kb = .04; % back emf
    R = 3; % resistance for motor
    b = 6e-5; % friction for motor

    damping = .7;
    Ts = 250;
    wn = 4/(damping*Ts);

    ein_max = 28; % maximum voltage of reaction wheel
    psi_ref = pi/6;
    psi0 = 0;
    psid0 = 0;
    state0 = [psi0; psid0];
    Kp = R*Isat*wn^2/Km;
    Kd = 2*R*Isat*damping*wn/Km;

    opts = odeset('RelTol', 1e-4, 'AbsTol', 1e-4);
    tspan = [0, 400];
    parameters = [Isat, Kp, Kd, Km, R, psi_ref];
    [t, state] = ode45(@rxn_wheel_control, tspan, state0, opts, parameters);
    psi = state(:,1);
    psid = state(:,2);
    ein = [];
    for ii = 1:length(t)
```

```
        ein(ii) = Kp*(psi_ref-psi(ii)) - Kd*psid(ii);
    end
    fprintf('The maximum voltage required is %f which is less than the maximum
 \n', max(ein))
    fprintf('voltage of 28 V so the reaction wheels can complete a 30 degreee
 \n', ein_max)
    fprintf('slewing maneuver\n')
    figure
    subplot(2,1,1)
    plot(t, rad2deg(psi))
    xlabel('Time [s]')
    ylabel('Yaw Angle [degree]')
    subplot(2,1,2)
    plot(t, rad2deg(psid))
    xlabel('Time [s]')
    ylabel('Yaw Rate [degree/s]')

    figure
    hold on
    plot(t,ein)
    plot([t(1), t(end)], [ein_max, ein_max], '-r')
    plot([t(1), t(end)], [-ein_max, -ein_max], '-r')
    hold off
    xlabel('Time [s]')
    ylabel('Input Voltage [V]')
    title('Compare Input Voltage to Limits')
end
```

Problem 1:

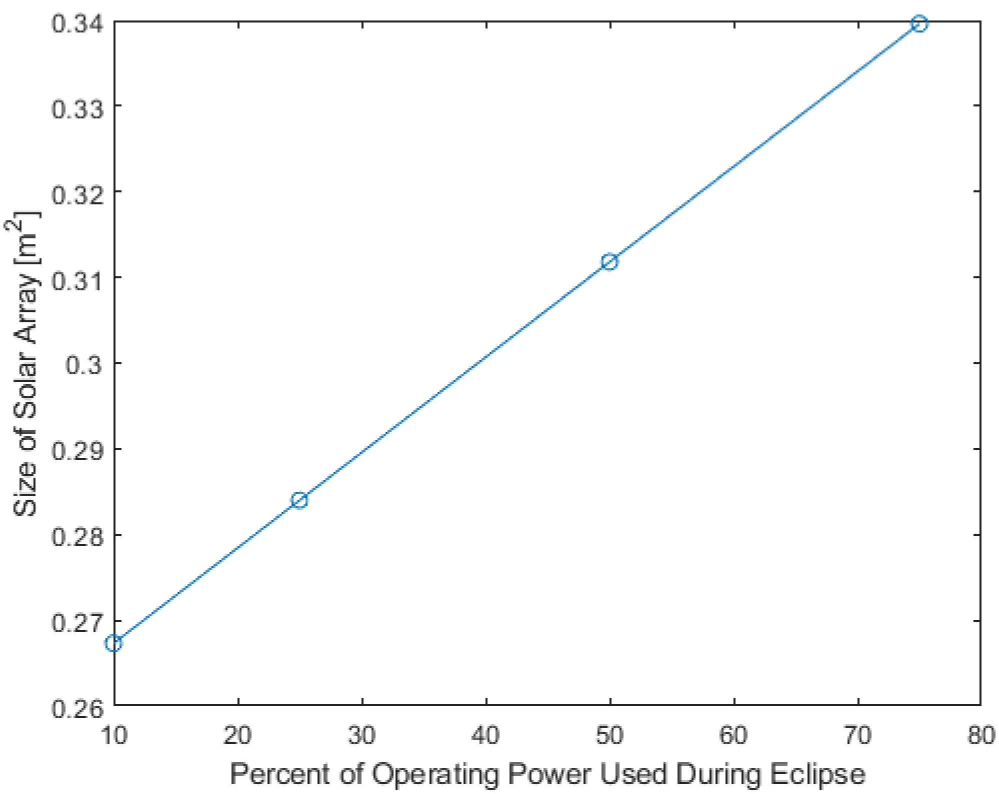
Functions

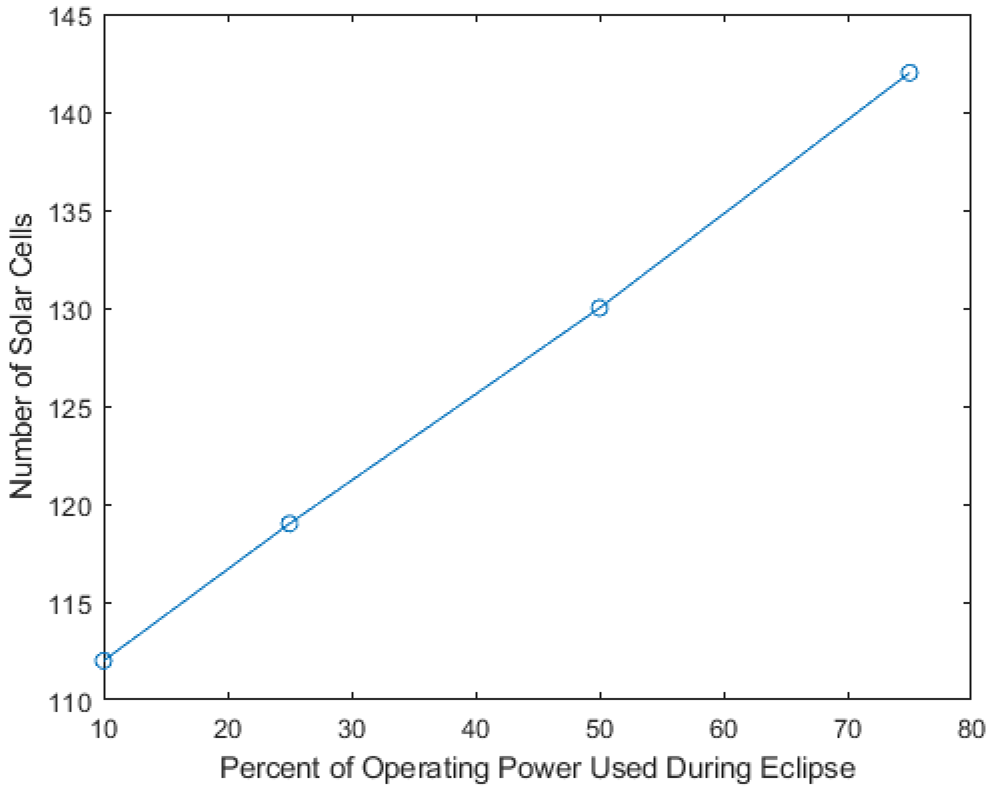
```
function [T, ecl_time, ecl_proportion] = eclipse(a, mu, Rb)

% Finds orbital period, eclipse time in seconds, and eclipse proportion of
% orbit
% Needs inputs of semi-major axis, mu of central body, radius of central
% body

    beta = asin(Rb/a);
    T = 2*pi*sqrt(a^3/mu);
    ecl_proportion = 2*beta/(2*pi);
    ecl_time = ecl_proportion*T;
    ecl_min = ecl_time/60;

    fprintf("The period is %f minutes\n", T/60)
    fprintf("The proportion of maximum eclipse is %f \n", ecl_proportion)
    fprintf("The maximum eclipse is %f minutes\n", ecl_min)
end

function dstate = single_axis_control(t, state, parameters)

    I = parameters(1);
```

```
    Kp = parameters(2);
    Kd = parameters(3);
    psi_ref = parameters(4);
%     t
    psi = state(1);
    psid = state(2);
    psidd = Kp*(psi_ref-psi)/I-Kd*psid/I;
    dstate = [psid; psidd];
end

function dstate = rxn_wheel_control(t, state, parameters)
    I = parameters(1);
    Kp = parameters(2);
    Kd = parameters(3);
    Km = parameters(4);
    R = parameters(5);
    psi_ref = parameters(6);
    C = (R*I)/Km;
%     t
    psi = state(1);
    psid = state(2);
    psidd = Kp*(psi_ref-psi)/C - Kd*psid/C;
    dstate = [psid; psidd];
end
```

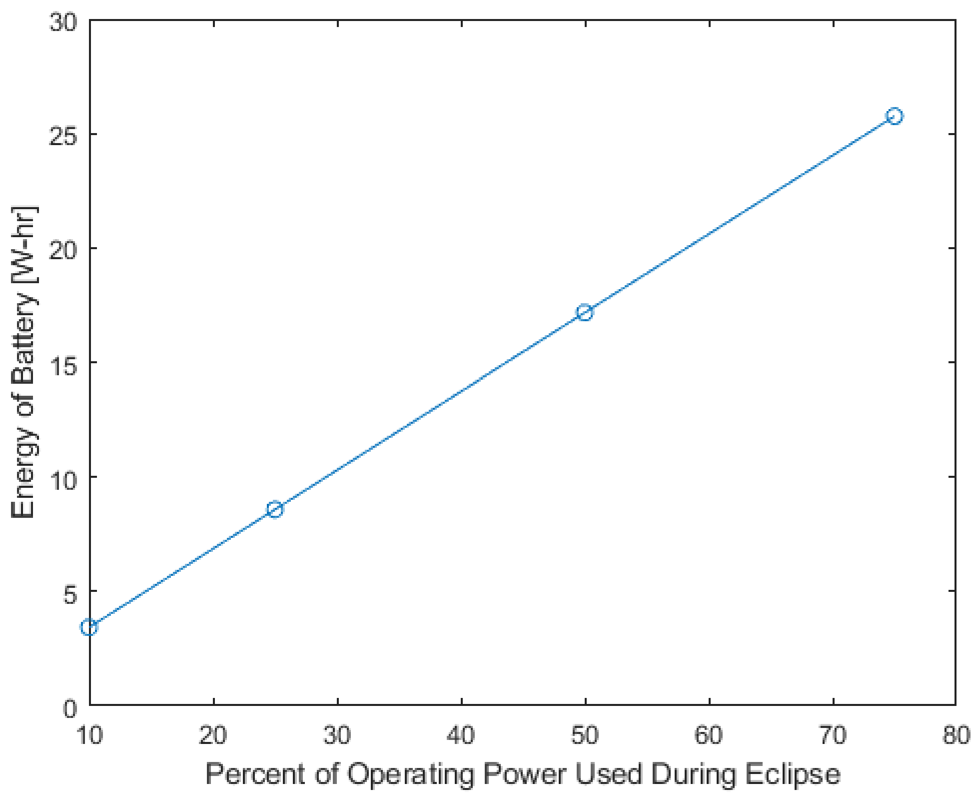
```
The period is 127.195136 minutes
The proportion of maximum eclipse is 0.275434
The maximum eclipse is 35.033911 minutes
The eclipse time from GMAT is 32.303019 minutes
For eclipse operations of 10.000000 percent of nominal power:
 A solar array array of 0.267308 m^2 is required
 A solar array array of 0.721731 kg is required
 A solar array array of 112.000000 cells are required
 A battery of 3.433569 W-hr is required
 A battery of 0.017168 kg is required
For eclipse operations of 25.000000 percent of nominal power:
 A solar array array of 0.283993 m^2 is required
 A solar array array of 0.766782 kg is required
 A solar array array of 119.000000 cells are required
 A battery of 8.583923 W-hr is required
 A battery of 0.042920 kg is required
For eclipse operations of 50.000000 percent of nominal power:
 A solar array array of 0.311803 m^2 is required
 A solar array array of 0.841867 kg is required
 A solar array array of 130.000000 cells are required
 A battery of 17.167846 W-hr is required
 A battery of 0.085839 kg is required
For eclipse operations of 75.000000 percent of nominal power:
 A solar array array of 0.339612 m^2 is required
 A solar array array of 0.916952 kg is required
 A solar array array of 142.000000 cells are required
 A battery of 25.751769 W-hr is required
 A battery of 0.128759 kg is required
Problem 2:
```

For Maximum Damping:
Proportional gain, Kp = 3029.585799
Derivative gain, Kd = 3938.461538
For Minimum Damping:
Proportional gain, Kp = 2000.000000
Derivative gain, Kd = 3938.461538
Problem 3:
The maximum voltage required is 10.258262 which is less than the maximum voltage of 28 V so the reaction wheels can complete a 30 degreee slewing maneuver

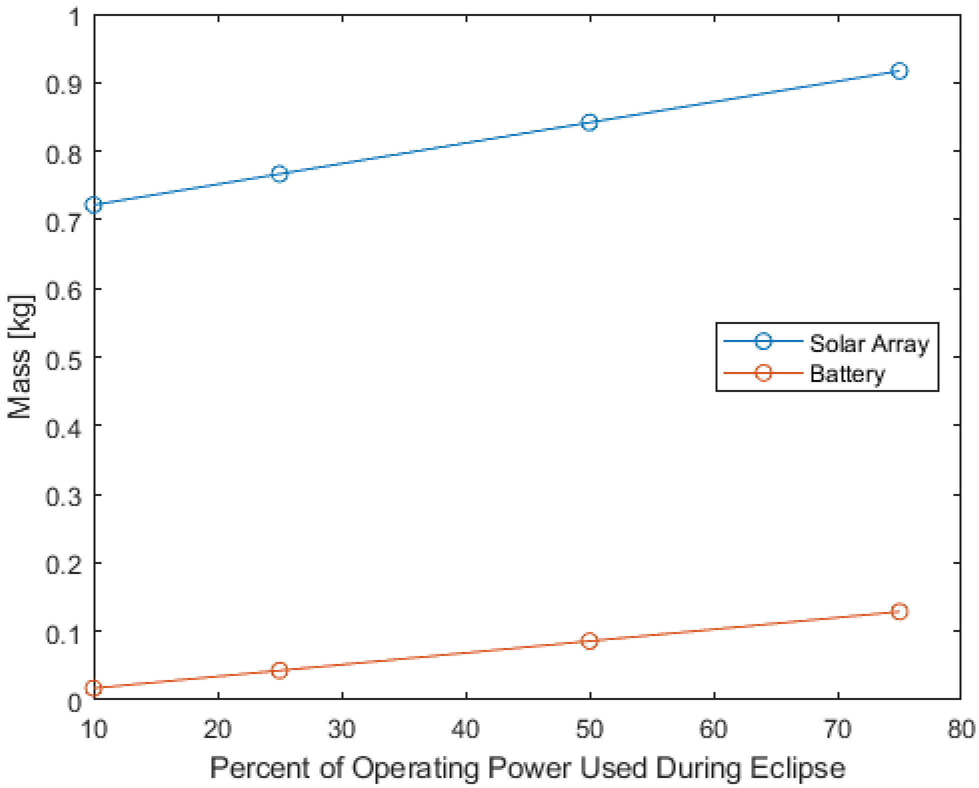




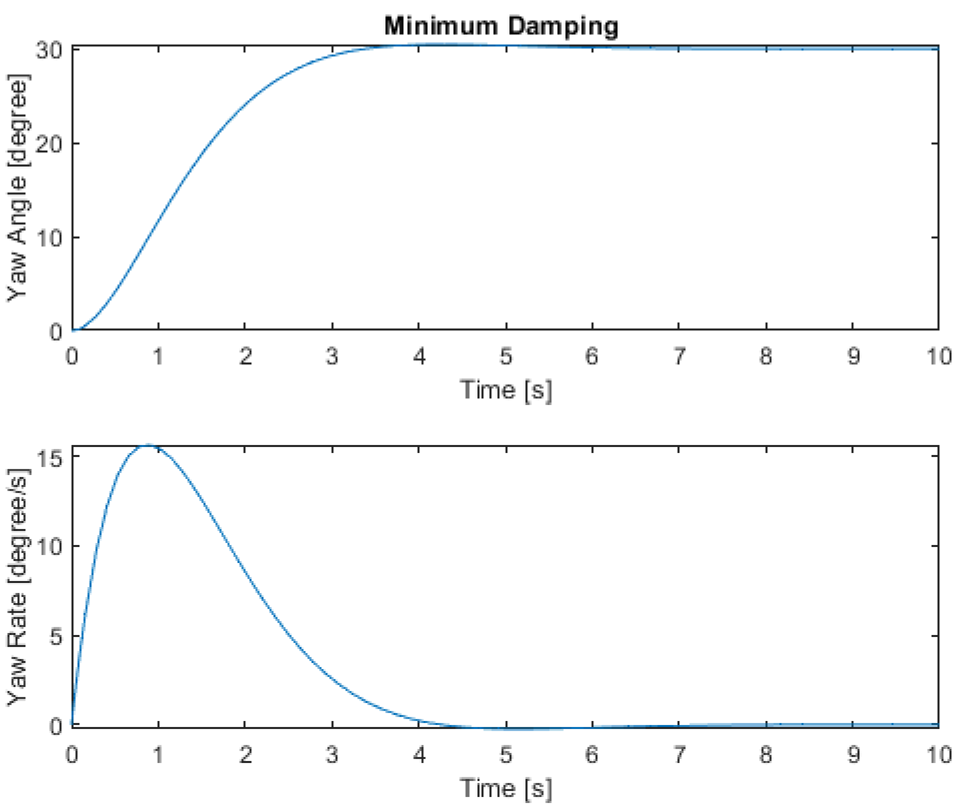
145
140
135
130
125
120
115
110
Number of Solar Cells
10
20
30
40
50
60
70
80
Percent of Operating Power Used During Eclipse



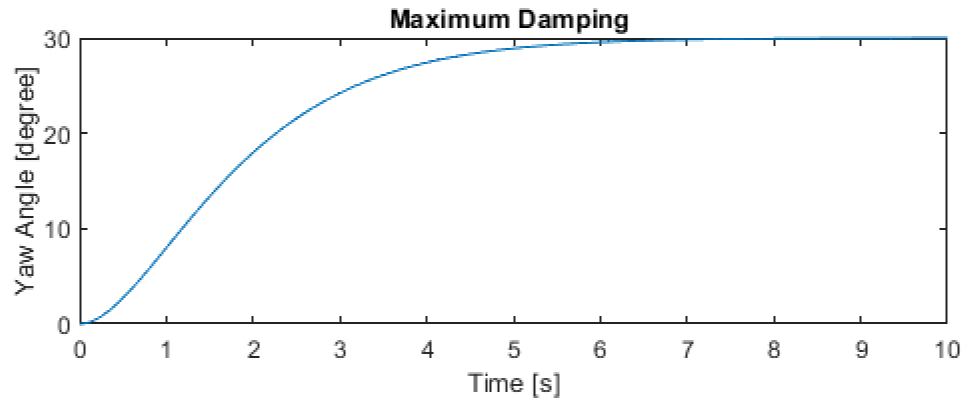
30
25
20
15
10
5
0
Energy of Battery [W-hr]
10
20
30
40
50
60
70
80
Percent of Operating Power Used During Eclipse

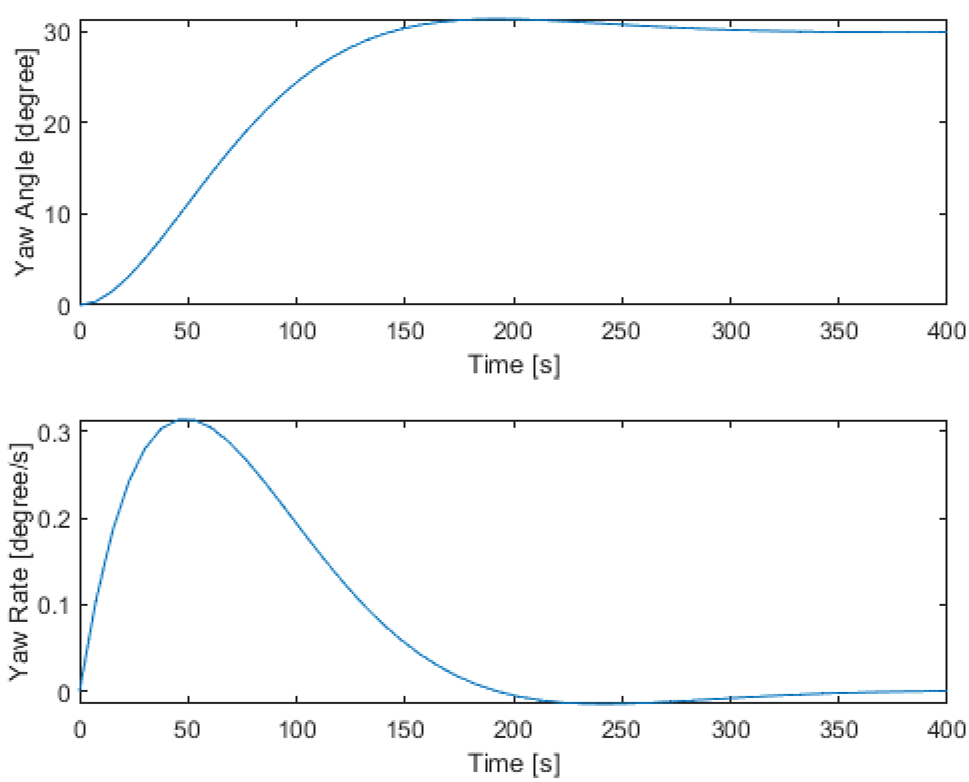


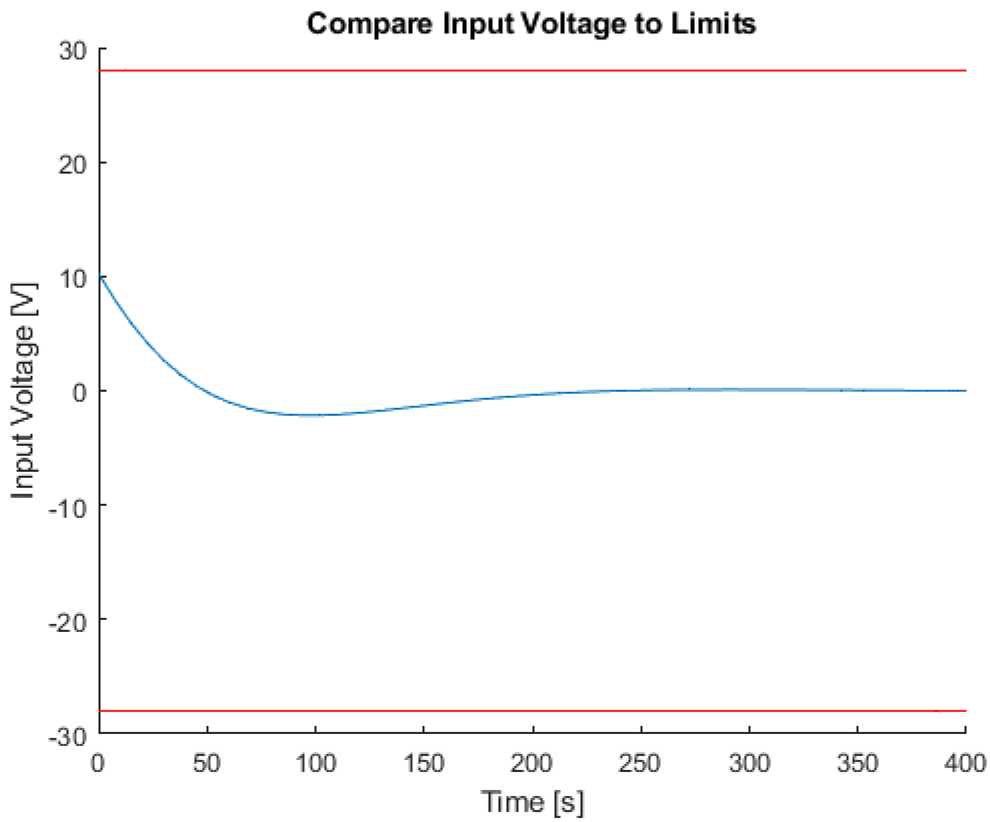
Solar Array
Battery
Mass [kg]
Percent of Operating Power Used During Eclipse



Minimum Damping
Yaw Angle [degree]
Time [s]
Yaw Rate [degree/s]
Time [s]


Maximum Damping
Yaw Angle [degree]
Time [s]
Yaw Rate [degree/s]
Time [s]


Yaw Angle [degree]
Time [s]
Yaw Rate [degree/s]
Time [s]



Published with MATLAB® R2021b