

```

%Gustavo Grinsteins
%ASEN 5050
%HW6

%House Keeping
clc;
clear;
%% Problem 1
% %Given
mu_sun = 1.32712428*10^11;
%earth vectors
R1 = [1.00078*10^8, 1.09250*10^8, -5.29404*10^3];
V1 = [-22.46086, 20.00474, -1.79921*10^-4];
JulianDateEarth = 2459528.5;
%venus vectors
R2 = [-1.05048*10^8, -2.375576*10^7, 5.73539*10^6];
V2 = [7.49784, -34.31464, -0.90369];
r1 = norm(R1);
r2 = norm(R2);
JulianDateVenus = 2459640.5;
%STEP 1: Find the transfer angle (theta<180)
less180 = true;
delta_ThetaStar = abs(acos(dot(R1,R2)/(r1*r2)));
%STEP 2: Calculate Geometric quantities
c = sqrt(r1^2+r2^2-2*(r1*r2)*cos(delta_ThetaStar));
s = (1/2)*(r1+r2+c);
%STEP 3: Determine if the TOF is for an ellipse
TOFp = (1/3)*sqrt(2/mu_sun)*((s^(3/2))+((s-c)^(3/2)));
OrbitDays = abs(JulianDateVenus-JulianDateEarth);
OrbitSeconds = OrbitDays*86400;
if OrbitSeconds > TOFp
    fprintf('The transfer orbit is elliptical \n')
    TOF = OrbitSeconds;
else
    fprintf('The transfer orbit is not elliptical \n')
end
%STEP 4: Determine correct alpha and beta
a_m = (s/2);
n_m = sqrt(mu_sun/(a_m^3));
alpha_m = pi;
beta_m_0 = 2*asin(sqrt((s-c)/(s)));
less180 = true;
if less180
    beta_m = beta_m_0;%(theta<180)
else
    beta_m = -beta_m_0;%(theta>180)
end
TOFmin = (1/n_m)*((alpha_m-beta_m)-(sin(alpha_m)-sin(beta_m)));

at = EllOrbitLambertEqSolve(s,c,a_m,TOF,TOFmin,mu_sun,true);

%Recalc check
n = sqrt(mu_sun/(at^3));
alpha_0 = 2*asin(sqrt((s)/(2*at)));
beta_0 = 2*asin(sqrt((s-c)/(2*at)));
if less180
    beta = beta_0;
else

```

```

    beta = -beta_0;
end
if TOF > TOFmin
    alpha = 2*pi - alpha_0;
else
    alpha = alpha_0;
end
TOF_check = (1/n)*((alpha-beta)-(sin(alpha)-sin(beta)));
p = ((4*at*(s-r1)*(s-r2))/(c^2))*sin((alpha+beta)/(2)).^2;
e = sqrt(1-(p/at));
mech_e = -(mu_sun)/(2*at);
%Part D - Calculating true anomalies
thetaS_1 = abs(acos((1/e)*((p/r1)-1)));
thetaS_2 = abs(acos((1/e)*((p/r2)-1)));
%Calculating speeds using f and G functions
f = 1- (r2/p)*(1-cos(delta_ThetaStar));
g = (r2*r1*sin(delta_ThetaStar))/(sqrt(mu_sun*p));
f_dot = sqrt(mu_sun/p)*tan(delta_ThetaStar/2)*(((1-cos(delta_ThetaStar))/(p))-(1/r2)-(1/r1));
g_dot = 1-(r1/p)*(1-cos(delta_ThetaStar));
V1_f = (1/g)*(R2-f*R1);
v1_f = norm(V1_f);
V2_i = (f_dot)*R1+g_dot*V1_f;
v2_i = norm(V2_i);
delta_V1 = V1_f - V1;
delta_V2 = V2 - V2_i;

%% Problem 2
%Load imported data
%JD X Y Z Vx Vy Vz (columns)
load('EarthEphemMatrix.mat');
load('MarsEphemMatrix.mat');
mu_sun = 1.32712428*10^11;
less180 = true;

%Create a nested for loop to access the imported data
for i = 1:length(HW6EphemEarth1)
    EarthJD = HW6EphemEarth1(i,1);
    R1 = [HW6EphemEarth1(i,2),HW6EphemEarth1(i,3),HW6EphemEarth1(i,4)];
    V1 = [HW6EphemEarth1(i,5),HW6EphemEarth1(i,6),HW6EphemEarth1(i,7)];
    for j = 1:length(HW6EphemMars)
        MarsJD = HW6EphemMars(j,1);
        TOF = (MarsJD - EarthJD)*86400; %days to seconds
        R2 = [HW6EphemMars(j,2),HW6EphemMars(j,3),HW6EphemMars(j,4)];
        V2 = [HW6EphemMars(j,5),HW6EphemMars(j,6),HW6EphemMars(j,7)];
        [EarthVinf,MarsVinf] = EllOrbitLambertEqSolve2(R1,R2,V1,V2,TOF,mu_sun,less180);
        EarthDepartingVelocities(i,j) = EarthVinf;
        MarsArravingVelocities(i,j) = MarsVinf;
    end
end

%creating the pork chop plot for earth departing
figure(1)
X = 0:2:(HW6EphemEarth1(end,1)-HW6EphemEarth1(1,1));
Y = 0:5:(HW6EphemMars(end,1)-HW6EphemMars(1,1));
%peaks = EarthDepartingVelocities(:);
%peaks = fix(peaks);
%peaks = unique(peaks);

```

```

%peaks = peaks(1:2:end);
%peaks = peaks(~isnan(peaks));
peaks = [3.9811 4.5 5 5.5 6 10 50 60 62 63 64 65 66];
%peaks = fix(peaks);
%peaks = unique(peaks);
contour(X,Y,EarthDepartingVelocities.',peaks,'ShowText','on','LabelSpacing',300)
colormap winter
title('V-infinity at Earth departure','FontSize',20)
xlabel('Earth Departure (Days Past 2005-06-20)','FontSize',15)
ylabel('Mars Arrival (Days Past 2005-12-01)','FontSize',15)

```

```

figure(2)
X = 0:2:(HW6EphemEarth1(end,1)-HW6EphemEarth1(1,1));
Y = 0:5:(HW6EphemMars(end,1)-HW6EphemMars(1,1));
contour(X,Y,EarthDepartingVelocities.',400)
colormap winter
title('V-infinity at Earth departure','FontSize',20)
xlabel('Earth Departure (Days Past 2005-06-20)','FontSize',15)
ylabel('Mars Arrival (Days Past 2005-12-01)','FontSize',15)

```

```

%creating the pork chop plot for Mars Arrival
figure(3)
X = 0:2:(HW6EphemEarth1(end,1)-HW6EphemEarth1(1,1));
Y = 0:5:(HW6EphemMars(end,1)-HW6EphemMars(1,1));
peaks = MarsArravingVelocities(:);
peaks = fix(peaks);
peaks = unique(peaks);
%peaks = peaks(1:2:end);
%peaks = peaks(~isnan(peaks));
%peaks = fix(peaks);
%peaks = unique(peaks);
contour(X,Y,MarsArravingVelocities.',200)
colormap jet
title('V-infinity at Mars Arrival','FontSize',20)
xlabel('Earth Departure (Days Past 2005-06-20)','FontSize',15)
ylabel('Mars Arrival (Days Past 2005-12-01)','FontSize',15)

```

```

%creating the pork chop plot for Mars Arrival
figure(4)
X = 0:2:(HW6EphemEarth1(end,1)-HW6EphemEarth1(1,1));
Y = 0:5:(HW6EphemMars(end,1)-HW6EphemMars(1,1));
contour(X,Y,MarsArravingVelocities.',[2.36 3 4 5 10 20 40], 'ShowText','on','LabelSpacing',100)
colormap jet
title('V-infinity at Mars Arrival','FontSize',20)
xlabel('Earth Departure (Days Past 2005-06-20)','FontSize',15)
ylabel('Mars Arrival (Days Past 2005-12-01)','FontSize',15)

```