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

Homework 6 -- Resonant Orbits

```
mu_S = 1.32712440018e11;
mu_E = 3.98600433e5;
AU = 1.49597870700e8;
Re = 6378.14;

Event.Launch = 2458951.5;
Event.VGA1 = 2459121.5;
Event.EGA1 = 2459438.5;
Event.EGA2 = 2460534.226567;
Event.JOI = 2461300.75;
% Calculate hyperbolic excess velocity of S/C upon arrivalt at EGA1
VGA1Ephem = Meeus(Event.VGA1);
EGA1Ephem = Meeus(Event.EGA1);

[R_VGA1,Vp_VGA1] =
 calcposvel(VGA1Ephem.Venus.a,VGA1Ephem.Venus.e,VGA1Ephem.Venus.i,VGA1Ephem.Venus.
[R_EGA1,Vp_EGA1] =
 calcposvel(EGA1Ephem.Earth.a,EGA1Ephem.Earth.e,EGA1Ephem.Earth.i,EGA1Ephem.Earth.
% V_inf_in = vf - v0V, V_inf_out = v0_V_FB - v0V;
[v_sc_VGA1,v_sc_EGA1] = solvelambert(R_VGA1,R_EGA1,(Event.EGA1 -
 Event.VGA1)*86400,0);

V_inf_in_EGA1 = v_sc_EGA1 - Vp_EGA1;

%Calculate hyperbolic excess velocity of S/C up departure at EGA1
EGA2Ephem = Meeus(Event.EGA2);
JOIEphem = Meeus(Event.JOI);

[R_EGA2,Vp_EGA2] =
 calcposvel(EGA2Ephem.Earth.a,EGA2Ephem.Earth.e,EGA2Ephem.Earth.i,EGA2Ephem.Earth.
[R_JOI,Vp_JOI] =
 calcposvel(JOIEphem.Jupiter.a,JOIEphem.Jupiter.e,JOIEphem.Jupiter.i,JOIEphem.Jupi

[v_sc_EGA2,v_sc_JOI] = solvelambert(R_EGA2,R_JOI,(Event.JOI -
 Event.EGA2)*86400,0);

V_inf_out_EGA2 = v_sc_EGA2 - Vp_EGA2;

%Construct periapse radius of each gravity assist as func. of phi
%first flyby
P_Earth = 3*365.242189*86400;
a = ((P_Earth/(2*pi))^2*mu_S)^(1/3);
V_sc_post_EGA1 = sqrt(mu_S*((2/norm(R_EGA1)) - (1/a)));
V_inf = (norm(V_inf_out_EGA2) + norm(V_inf_in_EGA1))/2;
V_sc_sun = sqrt((2*mu_S)/norm(R_EGA1) - (mu_S/a));

num = -norm(V_sc_sun)^2 + V_inf^2 + norm(Vp_EGA1)^2;
denom = 2*V_inf*norm(Vp_EGA1);
```

```
theta = acos((num/denom));
%construct VNC frame transformation
V_hat = Vp_EGA1/norm(Vp_EGA1);
N_hat = (cross(R_EGA1,Vp_EGA1))/norm(cross(R_EGA1,Vp_EGA1));
C_hat = cross(V_hat,N_hat);
T = [V_hat N_hat C_hat];

phi = [0:0.01:2*pi];

for i = 1:numel(phi)
    V_inf_out_EGA1 = V_inf*[cos(pi - theta);sin(pi -
 theta)*cos(phi(i));-sin(pi - theta)*sin(phi(i))];
    V_inf_out_EGA1(:,i) = T*V_inf_out_EGA1;
    V_inf_in_EGA2(:,i) = V_inf_out_EGA1(:,i) + Vp_EGA1 - Vp_EGA2;
    Psi_EGA1(i) = acos((dot(V_inf_in_EGA1,V_inf_out_EGA1(:,i)))/
(norm(V_inf_in_EGA1)*norm(V_inf_out_EGA1(:,i))));
    Psi_EGA2(i) = acos((dot(V_inf_in_EGA2(:,i),V_inf_out_EGA2))/
(norm(V_inf_in_EGA2(:,i))*norm(V_inf_out_EGA2)));
    rp_EGA1(i) = (mu_E/V_inf^2)*((1/cos((pi - Psi_EGA1(i))/2) - 1));
    rp_EGA2(i) = (mu_E/V_inf^2)*((1/cos((pi - Psi_EGA2(i))/2) - 1));
end
phiEGA1 = find(rp_EGA1 > (Re+300));
phiEGA2 = find(rp_EGA2 > (Re+300));
acceptable_phi = intersect(phiEGA1,phiEGA2);
plot_acc_phix = [rad2deg(phi(min(acceptable_phi)))
 rad2deg(phi(min(acceptable_phi))) rad2deg(phi(max(acceptable_phi)))
 rad2deg(phi(max(acceptable_phi))) ];
plot_acc_phiy = [0 14000 14000 0];

rplim = (Re+300)*ones(numel(phi),1);

figure
hold on
plot(phi*(180/pi),rp_EGA1)
plot(phi*(180/pi),rp_EGA2)
plot(phi*(180/pi),rplim,'--k')
patch(plot_acc_phix,plot_acc_phiy,'gree','FaceAlpha',0.2)
xlim([0 360])
ylim([0 14000])
grid on
grid minor
xlabel('\phi (Degrees)')
ylabel('r_p (km)')
title('Periapse Radius r_p VS \phi')
legend('EGA1 r_p','EGA2 r_p','Minimum r_p','Acceptable \phi region')
```

Published with MATLAB® R2020b