# Table of Contents

# Pegasus dV analysis

```
clear all
close all
```

# Constants

```
alt = 600; %km
i = pi/2; %rad, polar orbit
recession_i = 65*pi/180; %rad
Re = 6378; %km
mu = 3.986e5; %km3/s2
J2 = 0.00108263; %J2 of Earth
a_main = Re+alt; %km
n_main = sqrt(mu/(a_main*a_main*a_main));
P_main = 2*pi*sqrt(a_main*a_main*a_main/mu); %s
```

# Pick sma that we'll use as a transfer orbit

```
x = 5;

min_orbs = 5;
max_orbs = 10;
a_xfer_array = zeros(1,max_orbs-min_orbs);
dv_array = zeros(1,max_orbs-min_orbs);
v_main = sqrt(mu/a_main);
counter = 0;
for x = min_orbs:max_orbs %number of revs on xfer orbit until satellites on the sa
    counter = counter + 1;
P_xfer = P_main*(1-1/(2*x));
a_xfer = ((P_xfer/2/pi)^2*mu)^(1/3);
a_xfer_array(counter) = a_xfer;
v_xfer_apo = sqrt(2*mu/(a_main) - mu/a_xfer);
dV_hoh = abs(v_main - v_xfer_apo);
dv_array(counter) = dV_hoh*1e3;
end

[ax,p1,p2] = plotyy(min_orbs:max_orbs,a_xfer_array,min_orbs:max_orbs,dv_array);
hold on
% plot([1,10],[Re,Re],'--b')
ylabel('Transfer orbit SMA (km)')
ylabel(ax(2),'Apogee dV (m/s)')
xlabel('num orbits')
```
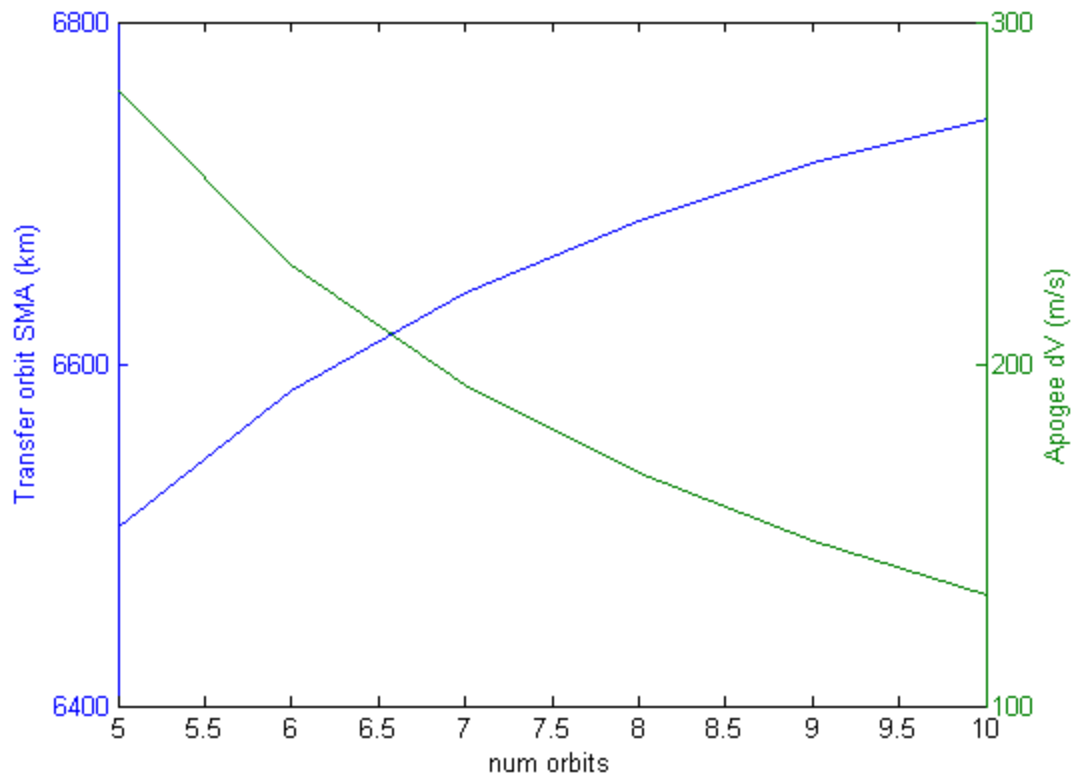
# Nodal regression

```matlab
e =  a_main/a_xfer - 1; %a_main is apogee
apo_alts = 600:200:1600;
calc_mp = @(m_payload, dV, Ve, fs) m_payload*(exp(dV/Ve)-1)/...
    (1-fs*(exp(dV/Ve)-1));

figure
for apo = apo_alts
    a_xfer = (apo+2*Re + alt)/2;
    e = 1-a_main/a_xfer;
    n_xfer = sqrt(mu/(a_xfer*a_xfer*a_xfer));
    Vp_xfer = sqrt(2*mu/a_main - mu/a_xfer);
incs = (65:85)*pi/180;
plane_change_time = zeros(length(incs));
inc_dvs = zeros(length(incs));
prop_mass = zeros(length(incs));

counter = 0;
for recession_i = incs
    counter = counter+1;
nodal_regression_rate = ...
    -3*n_xfer*J2*Re*Re*cos(recession_i)/(2*a_xfer*a_xfer*(1-e^2)^2);
plane_change_time(counter) = abs(30/(nodal_regression_rate*180/pi*3600*24)); %days
% Assuming we start from the first circular orbit...
```
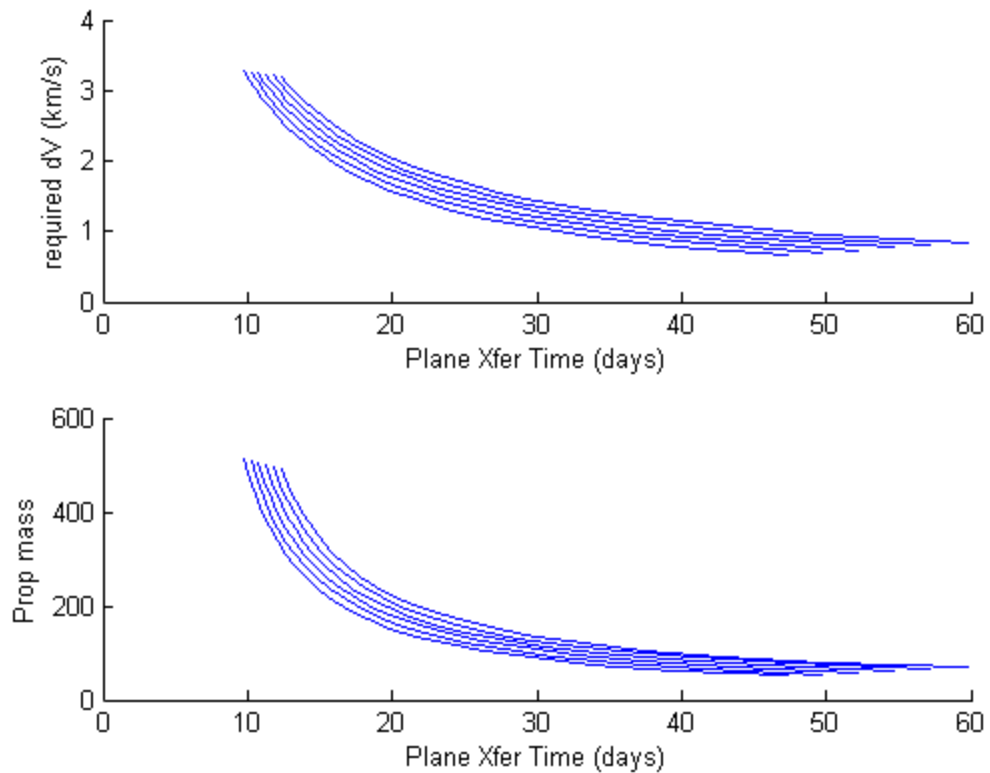
```matlab
    v_main = sqrt(mu/a_main);
    v_xfer_apo = sqrt(2*mu/(apo+Re) - mu/a_xfer);
    dV_hoh = abs(v_main - v_xfer_apo);
    di = i - recession_i;
    dV_xfer = abs(Vp_xfer - v_main);
    dV_inc = 2*v_xfer_apo*sin(di/2);
    inc_dvs(counter) = dV_inc + dV_xfer;
    prop_mass(counter) = calc_mp(200,(dV_inc + dV_xfer)*1e3,300*9.81,0.1);
end
% figure
% [ax,p1,p2] = plotyy(plane_change_time,inc_dvs,plane_change_time,prop_mass);
subplot(2,1,1)
hold on
plot(plane_change_time, inc_dvs)
xlabel('Plane Xfer Time (days)')
ylabel('required dV (km/s)')
subplot(2,1,2)
hold on
plot(plane_change_time, prop_mass)
xlabel('Plane Xfer Time (days)')
ylabel('Prop mass')

end
```

# To Pegasus Orbit from LV Primary Payload Orbit

```
v_781 = sqrt(mu/(Re+781));
v_xfer_ap = sqrt(2*mu/(Re+781) - mu/(Re + (781+600)/2));
v_xfer_pe = sqrt(2*mu/(Re+600) - mu/(Re + (781+600)/2));
dV_pri_to_mission = abs(v_781-v_xfer_ap) + abs(v_xfer_pe-v_main)
```

*dV_pri_to_mission =*

*0.0962*

*Published with MATLAB® R2013b*