Problem 1:

1. Using the vis-viva equation, the magnitude of the first impulse is given as:

Which is equivalent to the speed of the elliptic transfer orbit minus the speed of the original circular orbit. Since the transfer orbit has periapsis at the original circular orbit and apoapsis at the second circular orbit,

Substituting the expression for a:

The magnitude of the second impulse is found using the same method:

We also know that,

We can then nondimensionalize the expressions for the two impulses. Starting with rewriting in terms of R:

Now we can divide each expression by :

1. Finding the three impulses for the bi-elliptic transfer:

can be calculated as the semi-major axis of the first transfer orbit:

can then be written as:

Finding :

Where,

Substituting and :

Finding :

can then be written as:

The quantities can be defined:

The expressions can then be written as:

Normalizing the expressions:

1. The bi-parabolic transfer is a case of the bi-elliptic transfer where and consequently . Starting with rewriting the normalized expressions from the bi-elliptic transfer:

Taking the limit as :

The expression for the magnitudes can be found by multiplying by :



clc;clear;close all;

R1=1:0.01:20;

R2=10:0.01:16;

S=[2,5,10,11,12,15];

dvH=@(R) (sqrt((2.\*R)./(1+R))-1)+(sqrt(1./R).\*(1-sqrt(2./(1+R))));

dvE=@(R,S) (sqrt((2.\*R.\*S)./(1+R.\*S))-1)+(sqrt(1./(R.\*S)).\*(sqrt(2./(1+S))-sqrt(2./(1+R.\*S))))+(sqrt(2.\*S./(R+R.\*S))-sqrt(1./R));

dvP=@(R) (sqrt(2)-1)+(sqrt(1./R).\*(sqrt(2)-1));

figure(1)

hold on

p1H=dvH(R1);

p1P=dvP(R1);

p1E=zeros(length(R1),length(S));

for i=1:length(S)

p1E(:,i)=dvE(R1,S(i))';

plot(R1,p1E(:,i))

end

legend('','','','','','')

plot(R1,p1H,'b','DisplayName','Hohmann')

plot(R1,p1P,'r','DisplayName','Parabolic')

xlabel('R')

ylabel('\DeltaV/Vc1')

legend

figure(2)

hold on

p1H=dvH(R2);

p1P=dvP(R2);

p1E=zeros(length(R2),length(S));

marker=['r','y','m','c','k','g'];

for i=1:length(S)

p1E(:,i)=dvE(R2,S(i))';

plot(R2,p1E(:,i))

end

legend('','','','','','')

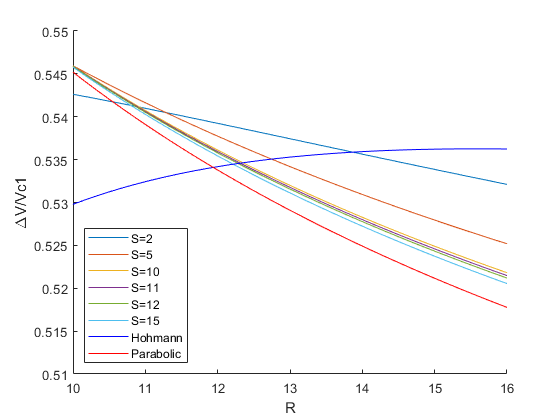
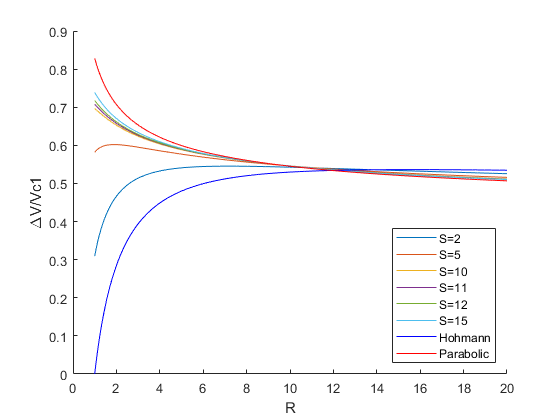
plot(R2,p1H,'b','DisplayName','Hohmann')

plot(R2,p1P,'r','DisplayName','Parabolic')

xlabel('R')

ylabel('\DeltaV/Vc1')

axis([-inf inf 0.51 0.55])



Problem 2:

clc;clear;

HminusP = @(R) (sqrt((2.\*R)./(1+R))-1)+(sqrt(1./R).\*(1-sqrt(2./(1+R))))-((sqrt(2)-1)+(sqrt(1./R).\*(sqrt(2)-1)));

transR=fsolve(HminusP,10);

S=[2,5,10,11,12,15];

Rvals=zeros(1,length(S));

for i=1:length(S)

s=S(i);

HminusE = @(R) (sqrt((2.\*R)./(1+R))-1)+(sqrt(1./R).\*(1-sqrt(2./(1+R))))-((sqrt((2.\*R.\*s)./(1+R.\*s))-1)+(sqrt(1./(R.\*s)).\*(sqrt(2./(1+s))-sqrt(2./(1+R.\*s))))+(sqrt(2.\*s./(R+R.\*s))-sqrt(1./R)));

Rvals(i)=fsolve(HminusE,12);

end

clc;

scatter(S,Rvals,'filled')

xlabel('S')

ylabel('R')

fprintf('The value of R where the total impulse of the Hohmann transfer is the same as the total impulse of the bi-parabolic transfer is: %g\n',transR)

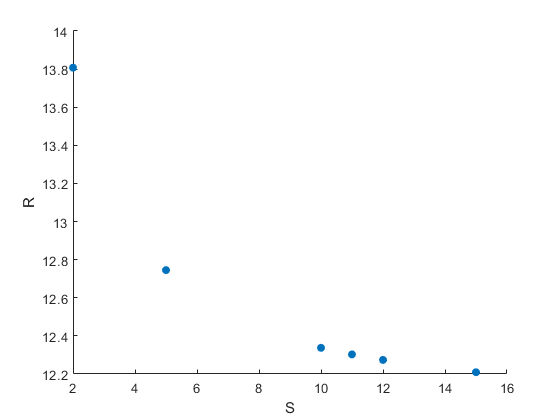
fprintf('The values of R where the total impulse of the Hohmann transfer is the same as the total impulse of the bi-elliptic transfer at S=2,5,10,11,12,15 is \n')

fprintf('%g, ',Rvals)

The value of R where the total impulse of the Hohmann transfer is the same as the total impulse of the bi-parabolic transfer is: 11.938

The values of R where the total impulse of the Hohmann transfer is the same as the total impulse of the bi-elliptic transfer at S=2,5,10,11,12,15 is

13.8076, 12.7452, 12.3373, 12.3029, 12.2739, 12.2093



Problem 3:

clc;clear;

v1=1;

mu=1;

v2=0.5;

r1=mu/v1^2;

r2=mu/v2^2;

R=r2/r1;

if R<11.938

fprintf('The Hohmann transfer is the most efficient\n')

end

%Finds properties of the transfer orbit

eT=(r2-r1)/(r2+r1);

aT=(r1+r2)/2;

pT=aT\*(1-eT^2);

nuT=0:0.01:pi;

radiusT=pT./(1+eT\*cos(nuT));

[xT,yT]=pol2cart(nuT,radiusT);

%Finds delta V

dv1=sqrt(mu/r1)\*(sqrt((2\*R)/(1+R))-1);

dv2=sqrt(mu/r1)\*sqrt(1/R)\*(1-sqrt((2)/(1+R)));

%Finds coords of circular orbits

nu=0:0.01:2\*pi;

radius1=zeros(1,length(nu));

radius1(:)=r1;

radius2=zeros(1,length(nu));

radius2(:)=r2;

[x1,y1]=pol2cart(nu,radius1);

[x2,y2]=pol2cart(nu,radius2);

hold on

plot(x1,y1,x2,y2,xT,yT)

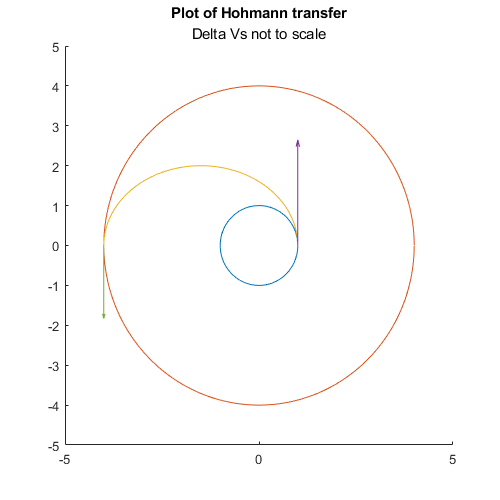
quiver(xT(1),yT(1),0,dv1)

quiver(xT(length(nuT)),yT(length(nuT)),0,-dv2)

axis([-5 5 -5 5])

set(gcf,'position',[300,300,500,500])

The Hohmann transfer is the most efficient



Problem 4:

clc;clear;

%Defines givens

mu=398600;

r1=300+6378.145;

i1=57;

i1=deg2rad(i1);

e1=0;

Omega1=60;

Omega1=deg2rad(Omega1);

omega=0;

%Calculates position of first orbit

nu=0:0.001:2\*pi;

position1=zeros(length(nu),3);

velocity1=zeros(length(nu),3);

for i=1:length(nu)

oe1=[r1,e1,Omega1,i1,omega,nu(i)];

[r,v]=oe2rv\_Hackbardt\_Chris(oe1,mu);

position1(i,:)=r';

velocity1(i,:)=v';

end

%Calculates pos of second orbit

i2=0;

period=23.934\*60\*60;

a2=(mu\*(period/(2\*pi))^2)^(1/3);

r2=a2;

e2=0;

Omega2=0;

position2=zeros(length(nu),3);

velocity2=zeros(length(nu),3);

for i=1:length(nu)

oe2=[a2,e2,Omega2,i2,omega,nu(i)];

[r,v]=oe2rv\_Hackbardt\_Chris(oe2,mu);

position2(i,:)=r';

velocity2(i,:)=v';

end

%calculates the first impulse

v1=sqrt(mu/r1);

v2=sqrt(mu/r2);

hvec1=cross(position1(1,:),velocity1(1,:));

hvec2=cross(position2(1,:),velocity2(1,:));

lvec=cross(hvec1,hvec2);

lvec=lvec/norm(lvec);

position1T=r1\*lvec;

aT=(r1+r2)/2;

eT=(r2-r1)/(r2+r1);

deltaV1=sqrt(((2\*mu)/r1)-(mu/aT))-v1;

u1vec=cross(hvec1,lvec)/norm(hvec1);

V1b4=v1\*u1vec;

deltaV1vec=deltaV1\*u1vec;

V1aft=V1b4+deltaV1vec;

%calculates pos of transfer orbit

nuT=0:0.001:pi;

oeT=rv2oe\_Hackbardt\_Chris(position1T,V1aft,mu);

positionT=zeros(length(nuT),3);

velocityT=zeros(length(nuT),3);

for i=1:length(nuT)

oeT(6)=nu(i);

[r,v]=oe2rv\_Hackbardt\_Chris(oeT,mu);

positionT(i,:)=r';

velocityT(i,:)=v';

end

%calculates second impulse

v2bf=sqrt(((2\*mu)/r2)-(mu/aT));

V2b4=-v2bf\*u1vec;

u2vec=-cross(hvec2,lvec)/norm(hvec2);

V2aft=v2\*u2vec;

deltaV2=v2-v2bf;

deltaV2vec=V2aft-V2b4;

%Calculates time and mass for transfer

tauT=2\*pi\*sqrt(aT^3/mu);

timeForTrans=tauT/2;

g0=9.80665;

Isp=320;

massRatio1=exp(deltaV1/(g0\*Isp));

massRatio2=exp(deltaV2/(g0\*Isp));

%plots

hold on

scale=10;

plot3(position1T(1),position1T(2),position1T(3),'r\*')

plot3(positionT(end,1),positionT(end,2),positionT(end,3),'r\*')

plot3(position1(:,1),position1(:,2),position1(:,3),position2(:,1),position2(:,2),position2(:,3))

plot3(positionT(:,1),positionT(:,2),positionT(:,3))

quiver3(position1T(1),position1T(2),position1T(3),deltaV1vec(1),deltaV1vec(2),deltaV1vec(3),5000)

quiver3(positionT(end,1),positionT(end,2),positionT(end,3),deltaV2vec(1),deltaV2vec(2),deltaV2vec(3),5000)

axis([-50000 50000 -50000 50000 -50000 50000])

title('View of the transfer between two circular orbits')

subtitle('Delta Vs not to scale')

view(-259.6770,16.8)

%Prints results

DeltaLet=char(916);

subScr1=char(8321);

subScr2=char(8322);

fprintf(['The magnitude of ' DeltaLet 'V' subScr1 ' = %g km/s\n'],deltaV1)

fprintf(['The magnitude of ' DeltaLet 'V' subScr2 ' = %g km/s\n'],deltaV2)

fprintf(['The total ' DeltaLet 'V required is %g km/s\n'],deltaV1+deltaV2)

fprintf('The transfer takes %g seconds to complete\n',timeForTrans)

fprintf('The mass ratio for impulse 1 is %g\n',massRatio1)

fprintf('The mass ratio for impulse 2 is %g\n',massRatio2)

fprintf('Changing the longitude of the ascending node does change where the orbit transfer begins\n')

fprintf('since that would rotate the orbit plane about the z-axis.\n')

The magnitude of ΔV₁ = 2.42572 km/s

The magnitude of ΔV₂ = 1.46682 km/s

The total ΔV required is 3.89254 km/s

The transfer takes 18989.9 seconds to complete

The mass ratio for impulse 1 is 1.00077

The mass ratio for impulse 2 is 1.00047

Changing the longitude of the ascending node does change where the orbit transfer begins

since that would rotate the orbit plane about the z-axis.

Chart

Description automatically generated

Problem 5:

clc;clear;

%Defines givens

mu=398600;

r1=350+6378.145;

i1=28;

i1=deg2rad(i1);

e1=0;

Omega1=0;

Omega1=deg2rad(Omega1);

omega=0;

%Calculates position of first orbit

nu=0:0.001:2\*pi;

position1=zeros(length(nu),3);

velocity1=zeros(length(nu),3);

for i=1:length(nu)

oe1=[r1,e1,Omega1,i1,omega,nu(i)];

[r,v]=oe2rv\_Hackbardt\_Chris(oe1,mu);

position1(i,:)=r';

velocity1(i,:)=v';

end

%Calculates pos of second orbit

i2=55;

i2=deg2rad(i2);

a2=26558;

r2=a2;

e2=0;

Omega2=0;

position2=zeros(length(nu),3);

velocity2=zeros(length(nu),3);

for i=1:length(nu)

oe2=[a2,e2,Omega2,i2,omega,nu(i)];

[r,v]=oe2rv\_Hackbardt\_Chris(oe2,mu);

position2(i,:)=r';

velocity2(i,:)=v';

end

%calculates the first impulse

v1=sqrt(mu/r1);

v2=sqrt(mu/r2);

hvec1=cross(position1(1,:),velocity1(1,:));

hvec2=cross(position2(1,:),velocity2(1,:));

lvec=cross(hvec1,hvec2);

lvec=lvec/norm(lvec);

position1T=r1\*lvec;

aT=(r1+r2)/2;

eT=(r2-r1)/(r2+r1);

deltaV1=sqrt(((2\*mu)/r1)-(mu/aT))-v1;

u1vec=cross(hvec1,lvec)/norm(hvec1);

V1b4=v1\*u1vec;

deltaV1vec=deltaV1\*u1vec;

V1aft=V1b4+deltaV1vec;

%calculates pos of transfer orbit

nuT=0:0.001:pi;

oeT=rv2oe\_Hackbardt\_Chris(position1T,V1aft,mu);

positionT=zeros(length(nuT),3);

velocityT=zeros(length(nuT),3);

for i=1:length(nuT)

oeT(6)=nu(i);

[r,v]=oe2rv\_Hackbardt\_Chris(oeT,mu);

positionT(i,:)=r';

velocityT(i,:)=v';

end

%calculates second impulse

v2bf=sqrt(((2\*mu)/r2)-(mu/aT));

V2b4=-v2bf\*u1vec;

u2vec=-cross(hvec2,lvec)/norm(hvec2);

V2aft=v2\*u2vec;

deltaV2=v2-v2bf;

deltaV2vec=V2aft-V2b4;

%Calculates time and mass for transfer

tauT=2\*pi\*sqrt(aT^3/mu);

timeForTrans=tauT/2;

%Calculates theta

theta=acos((dot(hvec1,hvec2)/(norm(hvec1)\*norm(hvec2))));

theta=rad2deg(theta);

%plots

hold on

scale=10;

plot3(position1T(1),position1T(2),position1T(3),'r\*')

plot3(positionT(end,1),positionT(end,2),positionT(end,3),'r\*')

plot3(position1(:,1),position1(:,2),position1(:,3),position2(:,1),position2(:,2),position2(:,3))

plot3(positionT(:,1),positionT(:,2),positionT(:,3))

quiver3(position1T(1),position1T(2),position1T(3),deltaV1vec(1),deltaV1vec(2),deltaV1vec(3),5000)

quiver3(positionT(end,1),positionT(end,2),positionT(end,3),deltaV2vec(1),deltaV2vec(2),deltaV2vec(3),5000)

axis([-30000 30000 -30000 30000 -30000 30000])

title('View of the transfer between two circular orbits')

subtitle('Delta Vs not to scale')

view(62.1,16.85)

%Prints results

DeltaLet=char(916);

subScr1=char(8321);

subScr2=char(8322);

fprintf('The line of intersection is along %g %g %g\n',lvec(1),lvec(2),lvec(3))

fprintf(['The magnitude of ' DeltaLet 'V' subScr1 ' = %g km/s\n'],deltaV1)

fprintf(['The magnitude of ' DeltaLet 'V' subScr2 ' = %g km/s\n'],deltaV2)

fprintf(['The total ' DeltaLet 'V required is %g km/s\n'],deltaV1+deltaV2)

fprintf('The transfer takes %g hours to complete\n',timeForTrans/3600)

fprintf('The location of impulse one is: %g km %g km %g km\n',position1T(1),position1T(2),position1T(3))

fprintf('The location of impulse two is: %g km %g km %g km\n',positionT(end,1),positionT(end,2),positionT(end,3))

fprintf('The eccentricity of the transfer orbit is %g\n',eT)

fprintf('Theta = %g deg',theta)

The line of intersection is along 1 0 0

The magnitude of ΔV₁ = 2.02605 km/s

The magnitude of ΔV₂ = 1.41089 km/s

The total ΔV required is 3.43693 km/s

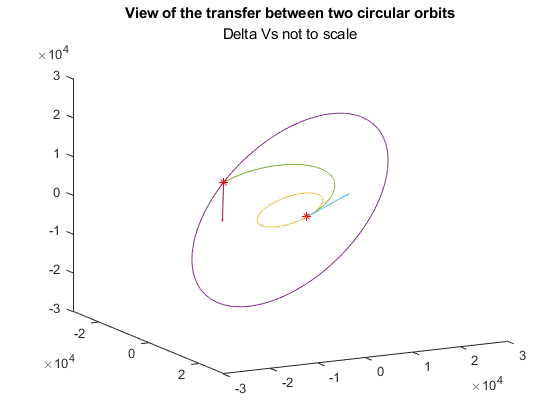
The transfer takes 2.96776 hours to complete

The location of impulse one is: 6728.15 km 0 km 0 km

The location of impulse two is: -26558 km 13.8973 km 7.38934 km

The eccentricity of the transfer orbit is 0.595739

Theta = 27 deg



Problem 6:

clc;clear;close all;

%Defines givens

mu=398600;

r1=300+6378.145;

i1=28.5;

i1=deg2rad(i1);

e1=0;

Omega1=0;

omega=0;

%Calculates position of first orbit

nu=0:0.001:2\*pi;

position1=zeros(length(nu),3);

velocity1=zeros(length(nu),3);

for i=1:length(nu)

oe1=[r1,e1,Omega1,i1,omega,nu(i)];

[r,v]=oe2rv\_Hackbardt\_Chris(oe1,mu);

position1(i,:)=r';

velocity1(i,:)=v';

end

%Calculates pos of second orbit

i2=0;

period=23.934\*60\*60;

a2=(mu\*(period/(2\*pi))^2)^(1/3);

r2=a2;

e2=0;

Omega2=0;

position2=zeros(length(nu),3);

velocity2=zeros(length(nu),3);

for i=1:length(nu)

oe2=[a2,e2,Omega2,i2,omega,nu(i)];

[r,v]=oe2rv\_Hackbardt\_Chris(oe2,mu);

position2(i,:)=r';

velocity2(i,:)=v';

end

%calculates the first impulse (not changing i)

v1=sqrt(mu/r1);

v2=sqrt(mu/r2);

hvec1=cross(position1(1,:),velocity1(1,:));

hvec2=cross(position2(1,:),velocity2(1,:));

lvec=cross(hvec1,hvec2);

lvec=lvec/norm(lvec);

position1T=r1\*lvec;

aT=(r1+r2)/2;

eT=(r2-r1)/(r2+r1);

deltaV1=sqrt(((2\*mu)/r1)-(mu/aT))-v1;

u1vec=cross(hvec1,lvec)/norm(hvec1);

V1b4=v1\*u1vec;

deltaV1vec=deltaV1\*u1vec;

V1aft=V1b4+deltaV1vec;

%calculates pos of transfer orbit (not chnging i)

nuT=0:0.001:pi;

oeT=rv2oe\_Hackbardt\_Chris(position1T,V1aft,mu);

positionT=zeros(length(nuT),3);

velocityT=zeros(length(nuT),3);

for i=1:length(nuT)

oeT(6)=nuT(i);

[r,v]=oe2rv\_Hackbardt\_Chris(oeT,mu);

positionT(i,:)=r';

velocityT(i,:)=v';

end

%calculates second impulse (part of first transfer orbit)

v2bf=sqrt(((2\*mu)/r2)-(mu/aT));

V2b4=-v2bf\*u1vec;

u2vec=-cross(hvec2,lvec)/norm(hvec2);

V2aft=v2\*u2vec;

deltaV2vec=V2aft-V2b4;

deltaV2=norm(deltaV2vec);

%Calculates first impulse (changes i)

oeT2=oeT;

oeT2(4)=i2;

oeT2(6)=0;

[r,v]=oe2rv\_Hackbardt\_Chris(oeT2,mu);

position1T2=r';

velocity1T2=v';

V1T2aft=velocity1T2;

deltaV1T2vec=V1T2aft-V1b4;

deltaV1T2=norm(deltaV1T2vec);

positionT2=zeros(length(nuT),3);

velocityT2=zeros(length(nuT),3);

nuT2=0:0.001:pi;

for j=1:length(nuT2)

oeT2(6)=nuT2(j);

[r,v]=oe2rv\_Hackbardt\_Chris(oeT2,mu);

positionT2(j,:)=r';

velocityT2(j,:)=v';

end

%Calculates second impules raises orbit

V2T2b4=velocityT2(end,:);

V2T2aft=v2\*u2vec;

deltaV2T2vec=V2T2aft-V2T2b4;

deltaV2T2=norm(deltaV2T2vec);

percentInc=0:0.01:1;

deltaInc=(i2-i1);

deltaVs=zeros(1,length(percentInc));

for i=1:length(percentInc)

%Calculates first impulse (changes i)

oeT3=oeT;

oeT3(4)=i1+(percentInc(i)\*deltaInc);

oeT3(6)=0;

[r,v]=oe2rv\_Hackbardt\_Chris(oeT3,mu);

position1T3=r';

velocity1T3=v';

V1T3aft=velocity1T3;

deltaV1T3vec=V1T3aft-V1b4;

deltaV1T3=norm(deltaV1T3vec);

oeT3(5)=pi;

oeT3(6)=pi;

%Calculates second impules raises orbit and finishes inclination change

[r,v]=oe2rv\_Hackbardt\_Chris(oeT3,mu);

positionendT3=r';

velocityendT3=v';

V2T3b4=velocityendT3;

V2T3aft=v2\*u2vec;

deltaV2T3vec=V2T3aft-V2T3b4;

deltaV2T3=norm(deltaV2T3vec);

deltaVs(i)=deltaV1T3+deltaV2T3;

end

deltaVs=deltaVs/v1;

minDV=min(deltaVs);

mostEff=deltaVs==minDV;

mostEff=percentInc(mostEff);

figure(1)

plot(percentInc,deltaVs)

xlabel('Percent of inclination change completed at periapsis of transfer orbit')

ylabel('\DeltaV / vc1')

%Prints results

DeltaLet=char(916);

subScr1=char(8321);

subScr2=char(8322);

fprintf(2,'For a transfer with inclination change occuring at apoapsis of transfer orbit\n')

fprintf(['The magnitude of ' DeltaLet 'V' subScr1 ' = %g km/s\n'],deltaV1)

fprintf(['The magnitude of ' DeltaLet 'V' subScr2 ' = %g km/s\n'],deltaV2)

fprintf(['The total ' DeltaLet 'V required is %g km/s\n'],deltaV1+deltaV2)

fprintf(2,'For a transfer with inclination change occuring at periapsis of transfer orbit\n')

fprintf(['The magnitude of ' DeltaLet 'V' subScr1 ' = %g km/s\n'],deltaV1T2)

fprintf(['The magnitude of ' DeltaLet 'V' subScr2 ' = %g km/s\n'],deltaV2T2)

fprintf(['The total ' DeltaLet 'V required is %g km/s\n'],deltaV1T2+deltaV2T2)

fprintf('\n')

fprintf(['The smallest total impulse occurs when f = %g with a value of ' DeltaLet 'V/vc1 = %g\n'],mostEff,minDV)

For a transfer with inclination change occuring at apoapsis of transfer orbit

The magnitude of ΔV₁ = 2.42572 km/s

The magnitude of ΔV₂ = 1.83023 km/s

The total ΔV required is 4.25595 km/s

For a transfer with inclination change occuring at periapsis of transfer orbit

The magnitude of ΔV₁ = 4.98922 km/s

The magnitude of ΔV₂ = 1.46683 km/s

The total ΔV required is 6.45605 km/s

The smallest total impulse occurs when f = 0.08 with a value of ΔV/vc1 = 0.547692

