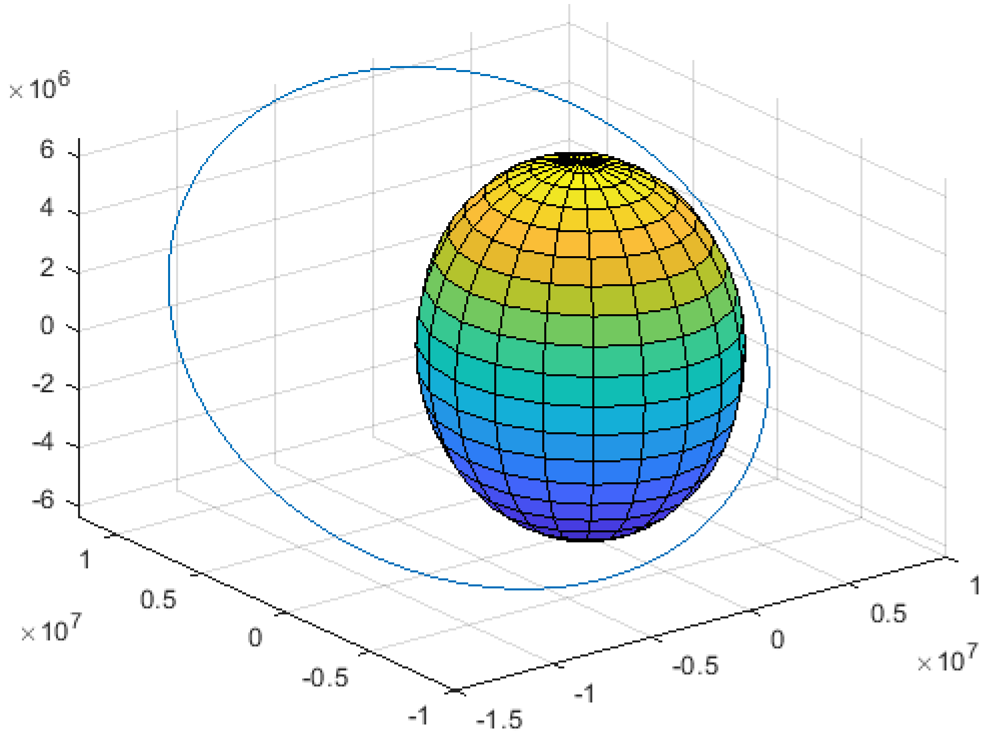
Table of Contents

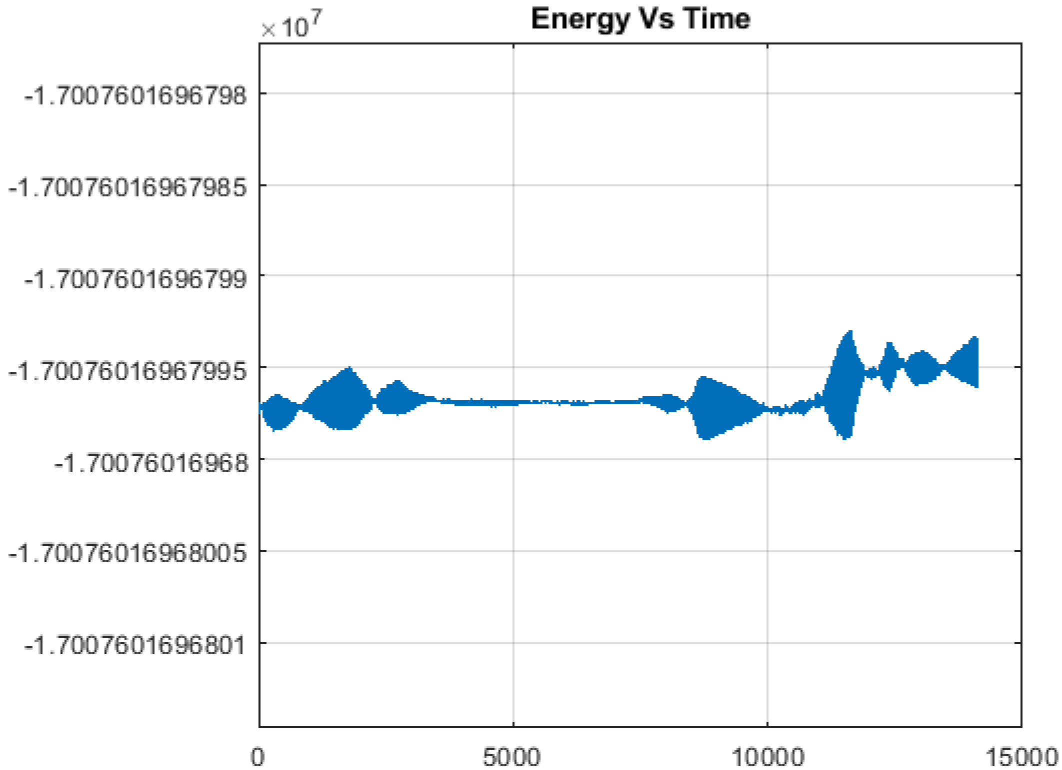

```
%Orbits HW 1

%1 Use ODE45 to simulate an orbit with the given initial conditions

clear all
close all
clc

%Planet Stuff

M = 5.97219e24;
G = 6.67408e-11;
mu = G*M;
R = 6371*10^3;

%Problem 1 Stuff

r = [7642 170 2186]*10^3; %m
semiMajor = norm(r);
r_dot = [0.32 6.91 4.29]*10^3; %m/s



v = norm(r_dot);
dist = norm(r);

%Intial Conditions for ODE45
S0 = [r r_dot];

period = 2*pi*sqrt(semiMajor^(3)/mu)*2;

Orbits = 1;

tspan = [0 period];

opts  = odeset('reltol', 1e-200, 'abstol', 1e-200);

[t,s] = ode45(@Satellite,tspan,S0,opts);

[X,Y,Z] = sphere;
X = X*R;
```

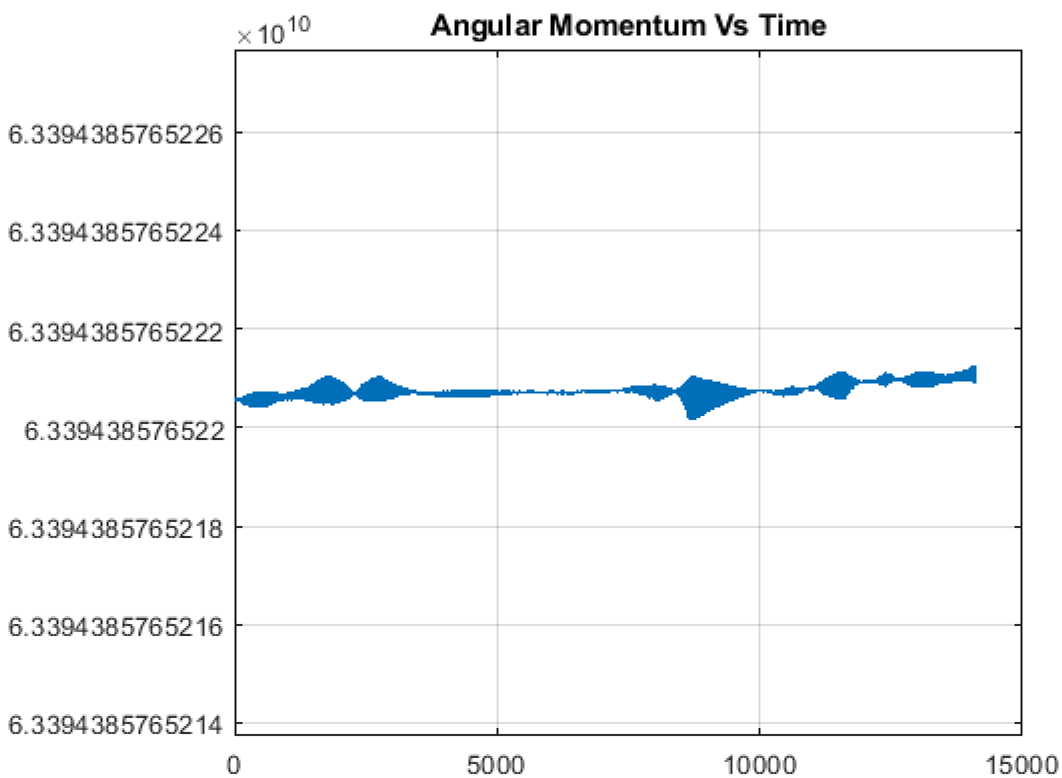
```
Y = Y*R;
Z = Z*R;

figure(1)
surf(X,Y,Z);
hold on
plot3(s(:,1),s(:,2),s(:,3))
grid on

%Getting regular values i guess idk im tired
for i = 1:length(s)
 VelVec(i) = norm(s(i,4:6));
 RadVec(i) = norm(s(i,1:3));
end
%Mass-specific orbit energy
Energy = 1/2.*(VelVec).^2 - mu./RadVec;

%Angular Momentum
for i = 1:length(s)
h = cross(s(i,1:3),s(i,4:6));
hVec(i,:) = h;
AngularMom(i) = norm(hVec(i,:));
end

%Eccentricity
for i = 1:length(s)
ee =   cross(s(i,4:6),hVec(i,:));
ecc(i,:) = (1/mu).*((ee)-mu.*(s(i,1:3)./RadVec(i)));
eccentricity(i) = norm(ecc(i,:));
end

semiMajor = min(RadVec)/(1-mean(eccentricity));
```

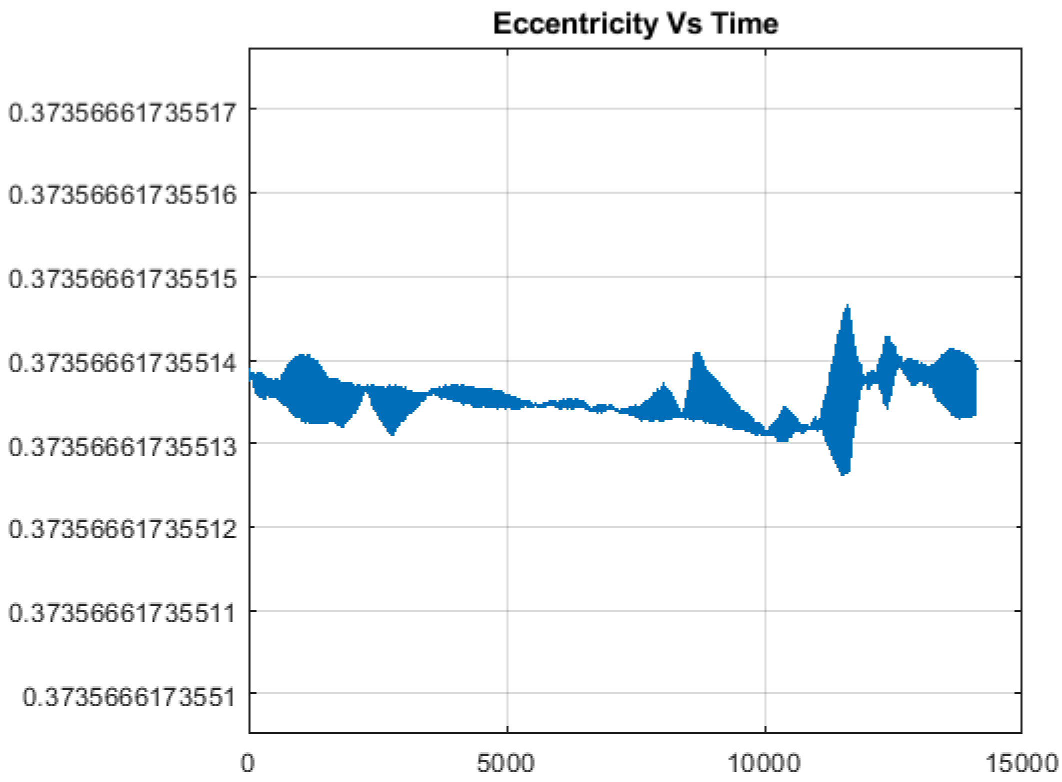


Plotting

```
figure(2)
plot(t,Energy)
grid on
title('Energy Vs Time')
figure(3)
plot(t,AngularMom)
grid on
title('Angular Momentum Vs Time')
figure(4)
plot(t,eccentricity)
grid on
title('Eccentricity Vs Time')
```


Energy Vs Time
×10^7
-1.7007601696798
-1.70076016967985
-1.7007601696799
-1.70076016967995
-1.70076016968
-1.70076016968005
-1.7007601696801
0
5000
10000
15000


Angular Momentum Vs Time
×10^10
6.3394385765226
6.3394385765224
6.3394385765222
6.339438576522
6.3394385765218
6.3394385765216
6.3394385765214
0
5000
10000
15000

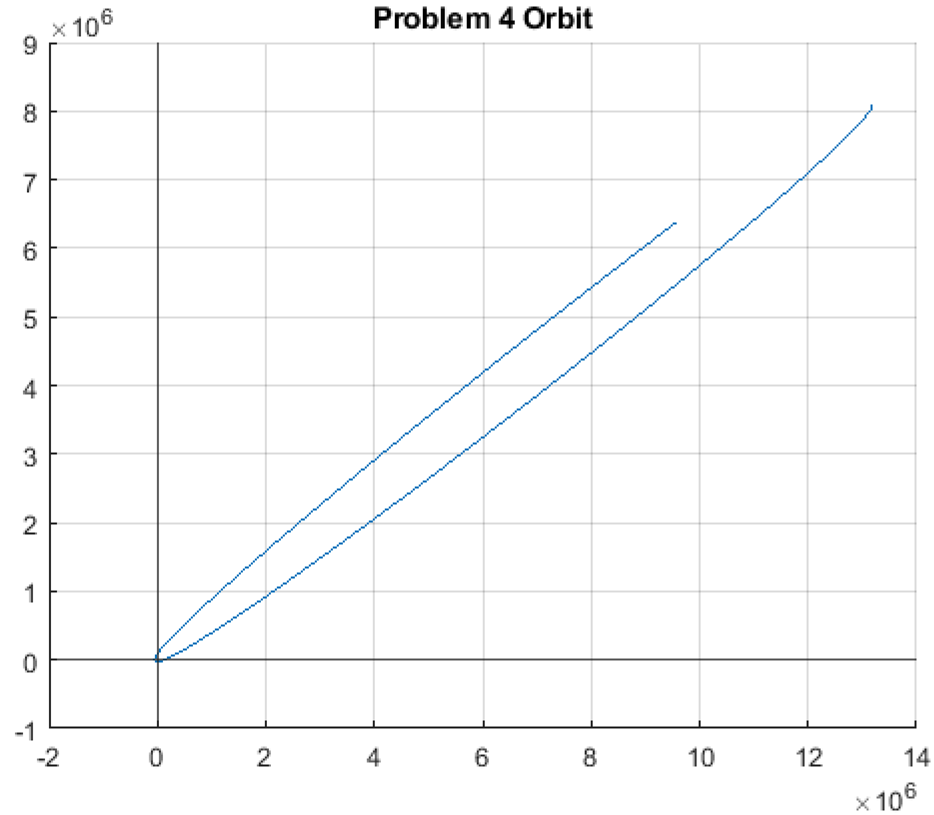


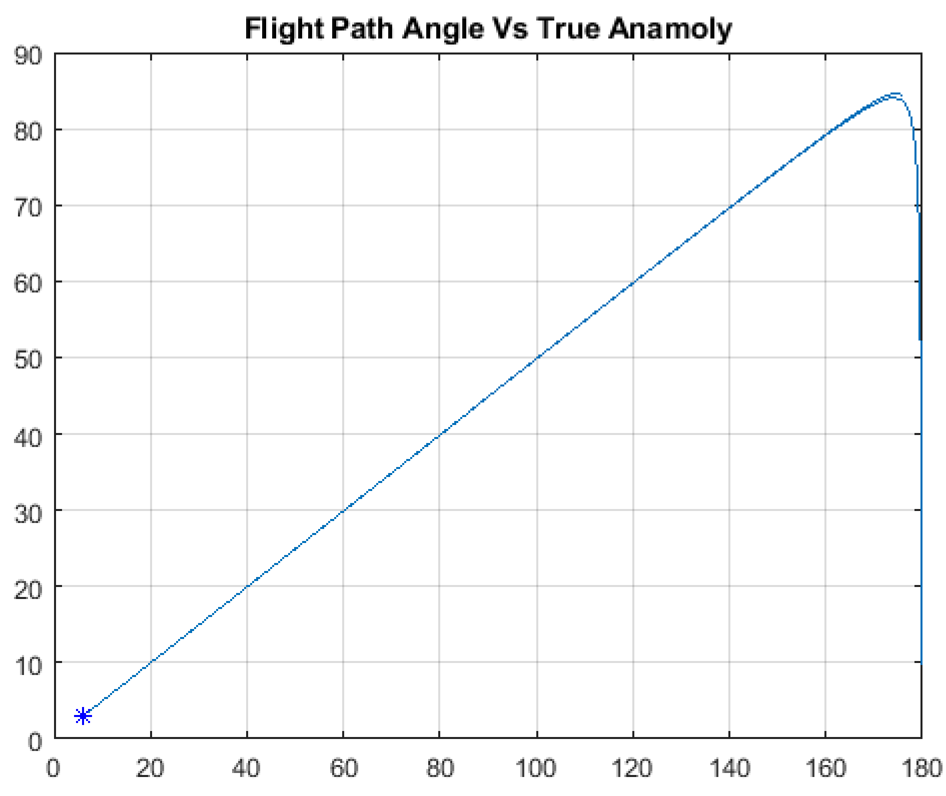
4

```
%Planet Stuff

M = 5.97219e24;
G = 6.67408e-11;
mu = G*M;
R = 6371*10^3;

%Problem 1 Stuff

% r = [7642 170 2186]*10^3; %m
% semiMajor = norm(r);
% r_dot = [0.32 6.91 4.29]*10^3 %m/s

r = [1.5 1 0.8]*R; %m
semiMajor = norm(r);

Tu = 806.8;


pew = R^3/(Tu)^2;

r_dot = [-0.5 -0.3 -0.2]*sqrt(pew/R); %m/s
```

```
v = norm(r_dot);
dist = norm(r);

%Intial Conditions for ODE45
S0 = [r r_dot];

period = 2*pi*Tu;
Orbits = 1;
tspan = [0 period];

opts  = odeset('reltol', 1e-3, 'abstol', 1e-3);
[t,s] = ode45(@Satellite,tspan,S0,opts);

figure(5)
hold on
plot3(s(:,1),s(:,2),s(:,3))
xline(0)
yline(0)
title('Problem 4 Orbit')
grid on

%Getting regular values i guess idk im tired
for i = 1:length(s)
 VelVec(i) = norm(s(i,4:6));
 RadVec(i) = norm(s(i,1:3));
end
%Mass-specific orbit energy
Energy = 1/2.*(VelVec).^2 - mu./RadVec;

%Angular Momentum
for i = 1:length(s)
h = cross(s(i,1:3),s(i,4:6));
hVec(i,:) = h;
AngularMom(i) = norm(hVec(i,:));
end

%Eccentricity
for i = 1:length(s)
ee =   cross(s(i,4:6),hVec(i,:));
ecc(i,:) = (1/mu).*((ee)-mu.*(s(i,1:3)./RadVec(i)));
eccentricity(i) = norm(ecc(i,:));
end

semiMajor = min(RadVec)/(1-mean(eccentricity));

%Now for the real part 4 stuff
rhat = r/norm(r);
%Looking for

%f
for i = 1:length(s)
```

```
f(i) = 2*atand(sqrt((1+eccentricity(i))/(1-
eccentricity(i)))*tan(Energy(i)/2));
ft(i) = acosd(dot(s(i,1:3),ecc(i,:))/(RadVec(i)*eccentricity(i)));
end
%i k&h
for i = 1:length(s)
 incline(i) = acosd(hVec(i,3)/AngularMom(i));
end
%w n & e
for i = 1:length(s)
n(i,:) = cross([0 0 1],hVec(i,:));
N(i) = norm(n);
w = acosd(dot(n(i,:),ecc(i,:))/(N(i)*eccentricity(i)));
end
%Omega ihat & n
for i = 1:length(s)
Omega(i) = acosd(n(i,1)/N(i));
end

%Flight Path
for i = 1:length(s)
gamma(i) = atand((eccentricity(i)*sind(ft(i)))/
(1+eccentricity(i)*cosd(ft(i))));
end

%Find position of minumum flight path angle
[MinAng,index] = min(gamma);
Dist = RadVec(index);

%Now the perifocal frame
i = mean(incline);
Omega = mean(Omega);
w = mean(w);

M1 = [1 0 0; 0 cosd(i) sind(i); 0 -sind(i) cosd(i)];

M3 = [cosd(Omega) sind(Omega) 0; -sind(Omega) cosd(Omega) 0; 0 0 1];
M32 = [cosd(w) sind(w) 0; -sind(w) cosd(w) 0; 0 0 1];

NP = M32*M1*M3;

Pr = s(index,1:3)*NP;
Pv = s(index,4:6)*NP;

%plotting flight path angle
figure(6)
plot(ft,gamma)
hold on
plot(ft(index),gamma(index),'b*')
title("Flight Path Angle Vs True Anamoly")
grid on
```


Problem 4 Orbit
×10^6
9
8
7
6
5
4
3
2
1
0
-1
-2
0
2
4
6
8
10
12
14
×10^6


Flight Path Angle Vs True Anamoly
90
80
70
60
50
40
30
20
10
0
0
20
40
60
80
100
120
140
160
180

5

```
Omega = 200;
i = 90;
w = 170;

M1 = [1 0 0; 0 cosd(i) sind(i); 0 -sind(i) cosd(i)];

M3 = [cosd(Omega) sind(Omega) 0; -sind(Omega) cosd(Omega) 0; 0 0 1];
M32 = [cosd(w) sind(w) 0; -sind(w) cosd(w) 0; 0 0 1];

NP = M32*M1*M3;
PN = NP';

r_dot = [2.73861;8.19407;0]';

rr = r_dot*PN;
```

2

```
a = 423;
e = 0.007092;
Va = sqrt((1-e)/(1+e))*sqrt(mu/a);
Vp = sqrt((1+e)/(1-e))*sqrt(mu/a);
rEarth = 6378.1*10^3;


E = mu/(2*a);
```

3

```
a = 6377*10^3;
e = 0.5;
Va = sqrt((1-e)/(1+e))*sqrt(mu/a);
Vp = sqrt((1+e)/(1-e))*sqrt(mu/a);
ra = a*(1+e);
rp = a*(1-e);

ha = Va*ra;
hp = Vp*rp;
```

Orbiting Body function

```
function dstatedt = Satellite(t,state)
%   S0 = [r r_dot];

% x = state(1);
% y = state(2);
% z = state(3);
% xdot = state(4);
% ydot = state(5);
```

```
% zdot = state(6);

M = 5.97219e24;
G = 6.67408e-11;
mu = G*M;


r = state(1:3);
rnorm = norm(r);
v = state(4:6);

vel = v;
accel = -mu/(rnorm^3).*r;


dstatedt = [vel;accel];



end
```

```
Warning: RelTol has been increased to 2.22045e-14.
```

Published with MATLAB® R2021a