## Table of Contents



# ORBITAL MECHANICS - HOMEWORK 4

```
clear all
close all
```

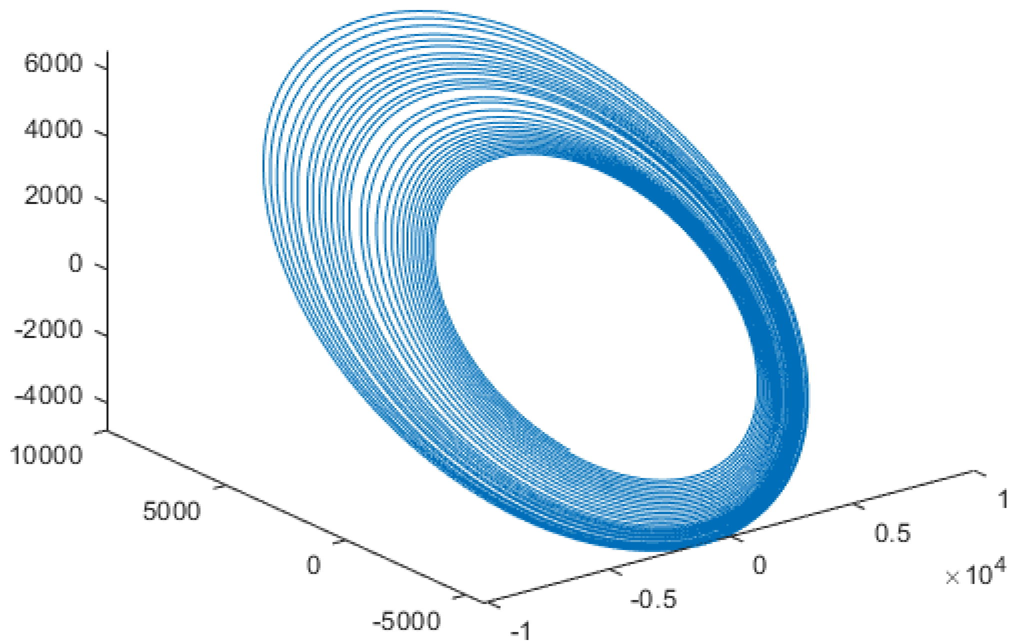
# Ex. 1

```
rvec=[7500;0;0];          %position vector [km]
vvec=[1;6;5];             %velocity vector [km/s]
mu=398600;                %gravitational parameter [km^3/s^2]
r=norm(rvec);
v=norm(vvec);

% a) Compute energy and period

E=(v^2/2)-(mu/r);         %orbital energy [km^2/s^2] or [J]
a=-mu/2/E;                %semi mayor axis [km]
n=sqrt(mu/a^3);           %orbit rate [rad/s]
T=2*pi/n;                 %orbital period [s]

fprintf('The orbit energy is: %.1f J\n',E)
fprintf('The orbital period is: %.1f s\n',T)

%This is an elliptical orbit, cause energy is less than zero and the
 semi
%mayor axis is a finite positive number (so it is not either a
 parabola or
%a hyperbola. It is also not a circular orbit because the semi mayor
 axis
%and the radius do not coincide

% b) Integrate the two body eqn

tspan=[0:5:20*T];
y0=[rvec;vvec];
[t,y1]=ode45(@orbitfun,tspan,y0);
figure(1)
plot3(y1(:,1),y1(:,2),y1(:,3))
title('Two body problem integration')
```

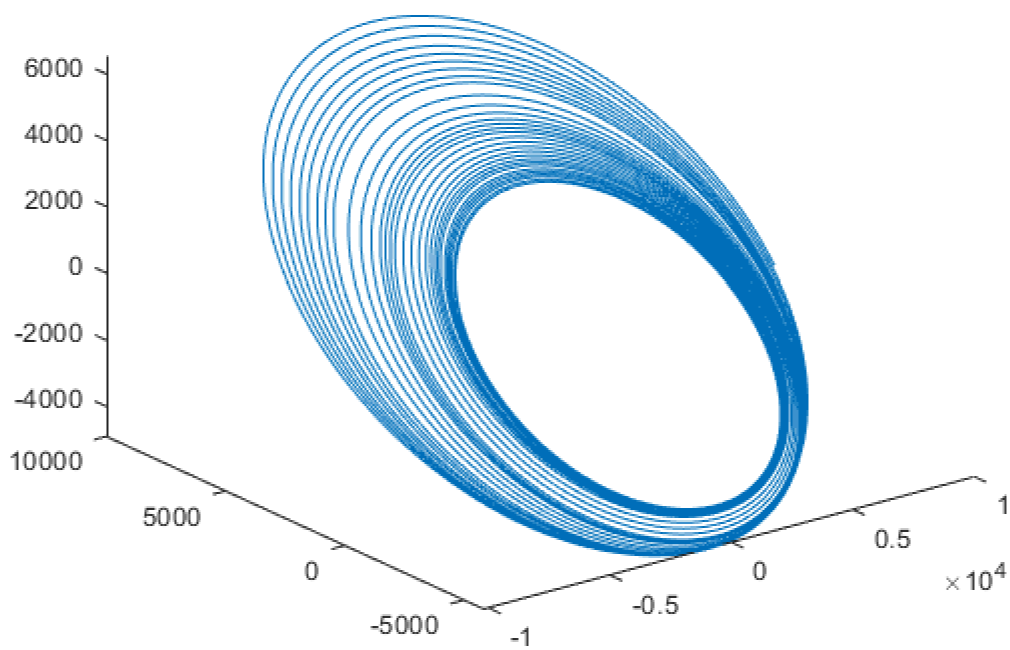
```
% c) Integrate the perturbed eqn of motion

[t,y2]=ode45(@orbitfunperturbed,tspan,y0);
figure(2)
plot3(y2(:,1),y2(:,2),y2(:,3))
title('Perturbed integration')

% d) Comment the difference
% The second orbit (perturbed) has a less constant decreasing of the
% "amplitude" and it looks like the perturbed plot ends at a semi
 mayor
% axis which is less than the one for the TBP. In other words the
 TBP's
% solution seems to be a larger orbit in the end.

% e) difference between the two radius
deltar(:,1)=y1(:,1)-y2(:,1);
deltar(:,2)=y1(:,2)-y2(:,2);
deltar(:,3)=y1(:,3)-y2(:,3);
for i=1:length(deltar)
        dr(i)=norm(deltar(i,:));
end
figure(3)
plot(tspan,dr,'-r','LineWidth',1.2)
xlabel('Time (sec)')
ylabel('\Deltar')
title('Difference in radius for perturbed and unperturbed
 integration')
```

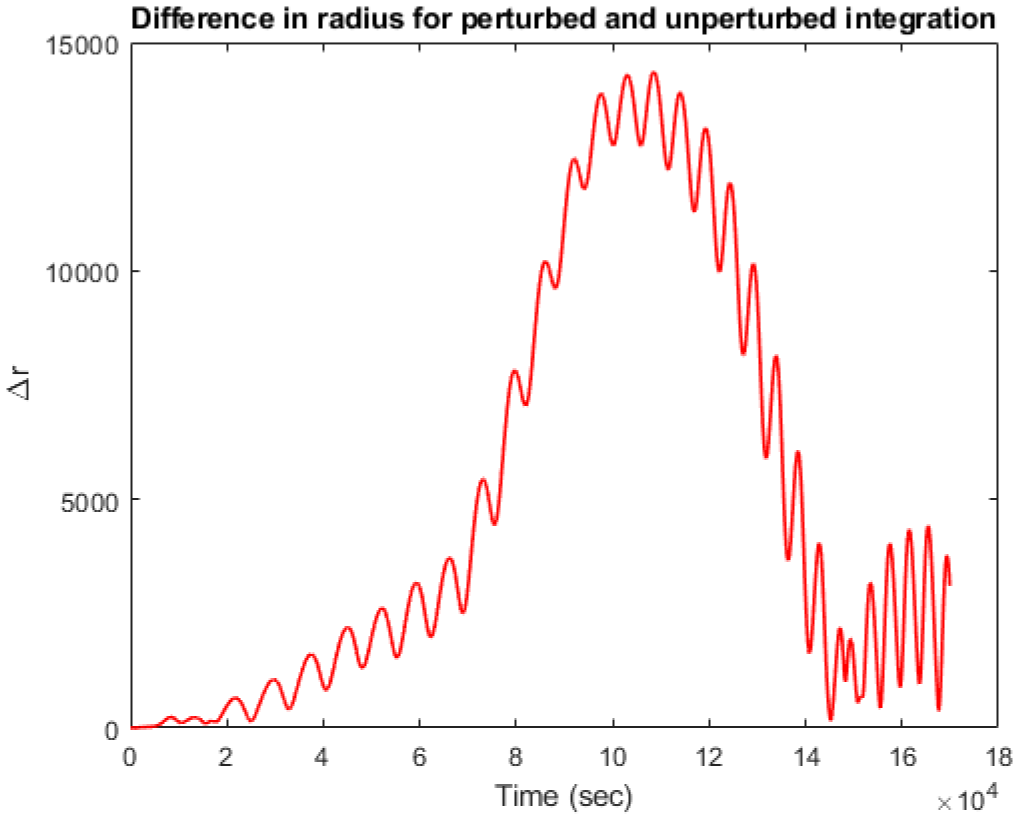
*The orbit energy is: -22.1 J*
*The orbital period is: 8495.9 s*

**Two body problem integration**




**Perturbed integration**






# Ex. 2

```
hp=350;                   %altitude at perigee [km]
ha=2020;                  %altitude at apogee [km]
re=6378;                  %mean earth radius [km]
rp=re+hp;                 %position at perigee [km]
ra=re+ha;                 %position at apogee [km]
i2=34*pi/180;             %inclination [rad]
a2=(rp+ra)/2;             %semi mayor axis [km]
e2=(ra-rp)/(rp+ra);       %eccentricity
n2=sqrt(mu/a2^3);         %mean orbit rate [rad/s]
J2_2=.00108263;           %earth oblateness
p2=a2*(1-e2^2);

nodal=((-3*J2_2*n2*re^2*cos(i2))/(2*p2^2))*180*86400/pi;
apsidal=((3/4)*J2_2*n2*(re/p2)^2*(5*sin(i2)^2-1))*180*86400/pi;
fprintf('The nodal regression rate is: %.1f deg/day\n',nodal)
fprintf('The apsidal rate of change is: %.1f deg/day\n',apsidal)
```

*The nodal regression rate is: -4.7 deg/day*
*The apsidal rate of change is: 1.6 deg/day*

# Ex.4

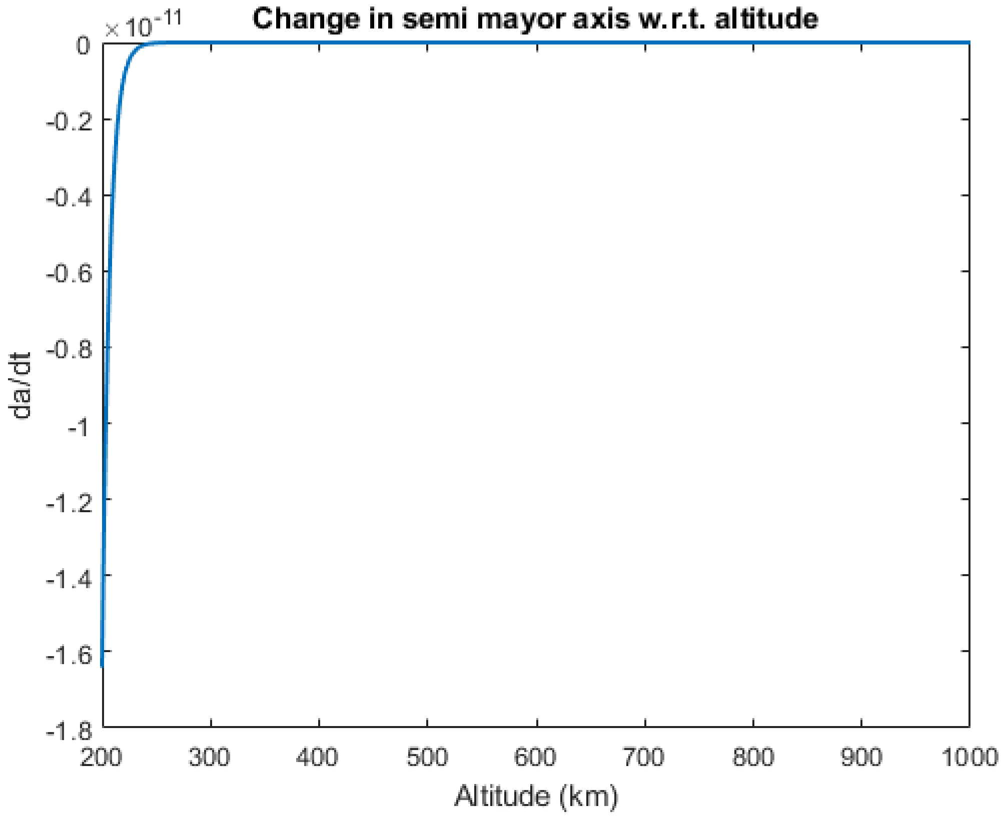
```
hp4=300;
```

```
ha4=500;
rp4=re+hp4;
ra4=re+ha4;
J2_4=.00108263;         %earth oblateness
a4=(rp4+ra4)/2;         %semi mayor axis [km]
e4=(ra4-rp4)/(rp4+ra4); %eccentricity
n4=sqrt(mu/a4^3);       %orbit rate [rad/s]
p4=a4*(1-e4^2);         %parameter [km]

%RAAN rate of change for sun-synchronous orbits [rad/s]
raan_rate4=(360*pi/180)/(365*86400);
%inclination (cosine)
i4=acos((-2*p4^2*raan_rate4)/(3*J2_4*n4*re^2));
%Rate of change of the argument of periapsis
dw_dt4=(3/4)*J2_4*n4*(re/p4)^2*(5*sin(i4)^2-1);
fprintf('The orbit inclination is: %.1f°\n',i4*180/pi)
fprintf('The average change in the argument of perigee is: %.1f deg/
day\n',dw_dt4*180*86400/pi)
```

*The orbit inclination is: 97.0°*
*The average change in the argument of perigee is: 15.8 deg/day*

# Ex. 5

```
h=[200:1:1000];         %range of altitude [km]
hf=(h+6378)*10^3;
mu5=398600*10^9;        %gravitational parameter [km^3/s^2]

% Let's assume a simple air model (following data come from AirBus
 A380
% specs)

A5=843;                 %wing area [m^2]
m5=386000;              %max landing mass [kg]
cd=0.0265;              %drag coefficient
rho0=1.225;             %density at sea level [kg/m^3]
h0=7;
v5=279.4;               %velocity [m/s]
for i=1:length(hf)
    rho5(i)=rho0*exp(-h(i)/h0);
    E5(i)=(v5^2/2)-(mu5/hf(i));
    a5(i)=-mu5/2/E5(i);
    da_dt(i)=-(A5/m5)*cd*rho5(i)*(v5^3)/(mu5/a5(i)^2);
end
figure(5)
plot(h,da_dt,'LineWidth',1.5)
xlabel('Altitude (km)')
ylabel('{da}/{dt}')
title('Change in semi mayor axis w.r.t. altitude')
```




# Functions

```
function dydt=orbitfun(t,y)
%{
  y      - column vector containing the position and velocity vectors
           of the system at time t
  r      - position vector
  v      - velocity vector
  mu     - gravitational parameter for eartg
  rn     - magnitude of the relative position vector
  a      - acceleration vectors of m1 & m2
  dydt   - column vector containing the velocity and acceleration
           vectors of the system at time t

%}
r=[y(1);y(2);y(3)];
v=[y(4);y(5);y(6)];

mu=398600;
rn=norm(r);

a=[(-mu/rn^3)*r(1);(-mu/rn^3)*r(2);(-mu/rn^3)*r(3)];

dydt=[v;a];
end
```

```
function dydt=orbitfunperturbed(t,y)
%{
  y      - column vector containing the position and velocity vectors
           of the system at time t
  r      - position vector
  v      - velocity vector
  mu     - gravitational parameter for eartg
  rn     - magnitude of the relative position vector
  a      - acceleration vectors of m1 & m2
  dydt   - column vector containing the velocity and acceleration
           vectors of the system at time t

%}
r=[y(1);y(2);y(3)];
v=[y(4);y(5);y(6)];

mu=398600;
rn=norm(r);
J2=.00108263;
re=6378;

ad_constant=(-3*J2*re^2*mu)/(2*rn^4);

advec=ad_constant*[1;0;0];
a=[(-mu/rn^3)*r(1)+advec(1);(-mu/rn^3)*r(2)+advec(2);(-mu/
rn^3)*r(3)+advec(3)];


dydt=[v;a];
end
```

*Published with MATLAB® R2019b*