Table of Contents


```
% MANE 4100 Computational Assignment
% By: LESLIE ALEMAN
```

Clear Workspace

```
clear all
close all
clc
```

Variable descriptions

```
%time - 1,000 x 1 vector of equally spaced ephemeris time going from
 t0 to tf

%stateR - true  position vector of Cassini relative to Saturn

%r12 - position vector of Cassini relative to Saturn at t0

%r - propogated position vector of Cassini relative to Saturn using
 two-body

%rB1 - propogated position vector of Cassini using n-body (CASSINI
+SATURN+TITAN)

%rB2 - propogated position vector of Cassini using n-body (CASSINI
+SATURN+SUN)

%rB3 - propogated position vector of Cassini using n-body (CASSINI
+SATURN+TITAN+SUN)

%e_2body - difference between two-body results and the true trajectory

%e_CST - difference between n-body results(CASSINI+SATURN+TITAN) and
 the true trajectory
```

```
%e_CSS - difference between n-body results(CASSINI+SATURN+SUN) and the
 true trajectory

%e_CSTS - difference between n-body results(CASSINI+SATURN+TITAN+SUN)
 and the true trajectory
```

Load SPICE Kernels

```
% Load leap second kernel (lsk)
cspice_furnsh('C:\Users\lesli\Documents\MATLAB\Space Comps
\Assignment1\naif0011.tls');

% Load ephemeris file (spk)
cspice_furnsh('C:\Users\lesli\Documents\MATLAB\Space Comps
\Assignment1\071004AP_SCPSE_07272_07328.bsp');
```

Part a: Construct time vector

```
% Startting and ending date strings
t0 = 'Oct 5, 2007 12:00:00.0';
tf = 'Nov 5, 2007 10:00:00.0';

% Convert date string to ephemeris time (et)
% Note that ephemeris time is in units of seconds
et0 = cspice_str2et(t0);
etf = cspice_str2et(tf);

time = linspace(et0,etf,1000);
```

Part b: Compute the location of Cassini

Use spkezr function Syntax: cspice_spkezr('TARGET', et, 'FRAME', 'LT', 'ORIGIN') Use 'J2000' as the frame

```
% Get position and velocity of Cassini relative to Saturn
[state12, lt12] = cspice_spkezr('CASSINI',
 time , 'J2000', 'LT', 'SATURN');

stateR = state12(1:3,:);
r12 = state12(1:3,1);%radius from saturn to cassini
v12 = state12(4:6,1);
```

Part c: Propagate position and velocity of Cassini using the Two-Body equation

```
%At t0:
options = odeset('RelTol',1e-12,'AbsTol',1e-12);
[t,x] = ode45(@TwoBody, time,[r12;v12],options);
r = x(:,1:3);
```

Part d: Propagate postion and Velocity of Cassini using the N-Body

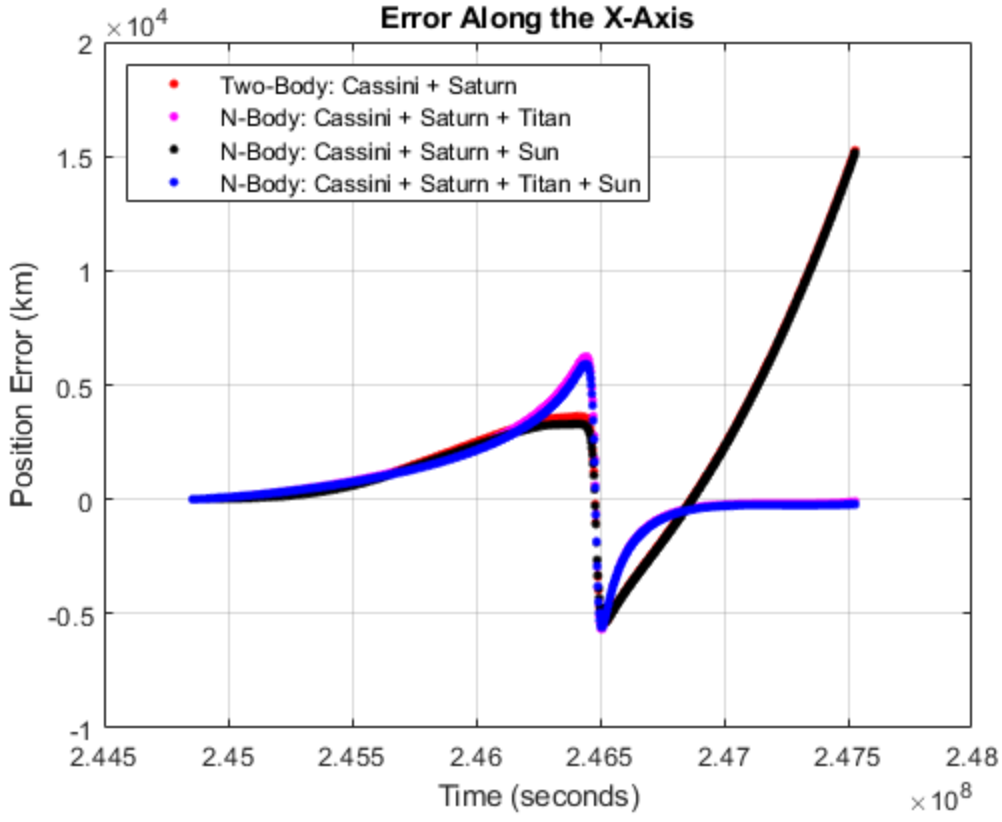
```
options = odeset('RelTol',1e-12,'AbsTol',1e-12);
[tB1,xB1] = ode45(@nBody, time,[r12;v12],options); %CASSINI+SATURN
+TITAN
rB1 = xB1(:,1:3);
%---------------------------
options = odeset('RelTol',1e-12,'AbsTol',1e-12);
[tB2,xB2] = ode45(@nBody2, time,[r12;v12],options); %CASSINI+SATURN
+SUN
rB2 = xB2(:,1:3);
%-------------------------------------------------------
options = odeset('RelTol',1e-12,'AbsTol',1e-12);
[tB3,xB3] = ode45(@nBody3, time,[r12;v12],options); %CASSINI+SATURN
+TITAN+SUN
rB3 = xB3(:,1:3);

%{
%% Plot orbits
%figure(1)
plot3(stateR(1,:),stateR(2,:),stateR(3,:),'k'), axis equal, grid on
[xs,ys,zs] = sphere(50);
hold on
saturn_r = 60270;
surf(saturn_r*xs,saturn_r*ys,saturn_r*zs)
title('CSPICE Solution')
hold off

%-----------------------------------------
%figure(2)
plot3(r(:,1),r(:,2),r(:,3),'k'), axis equal, grid on
[xs,ys,zs] = sphere(50);
hold on
saturn_r = 60270;
surf(saturn_r*xs,saturn_r*ys,saturn_r*zs)
title('Two-Body System')
hold off
%-------------------------
%figure(3)
plot3(rB1(:,1),rB1(:,2),rB1(:,3),'k'), axis equal, grid on
[xs,ys,zs] = sphere(50);
hold on
saturn_r = 60270;
surf(saturn_r*xs,saturn_r*ys,saturn_r*zs)
title('Cassini+Saturn+Titan System')
hold off
%-----------------------------------------
%figure(4)
plot3(rB2(:,1),rB2(:,2),rB2(:,3),'k'), axis equal, grid on
[xs,ys,zs] = sphere(50);
hold on
```

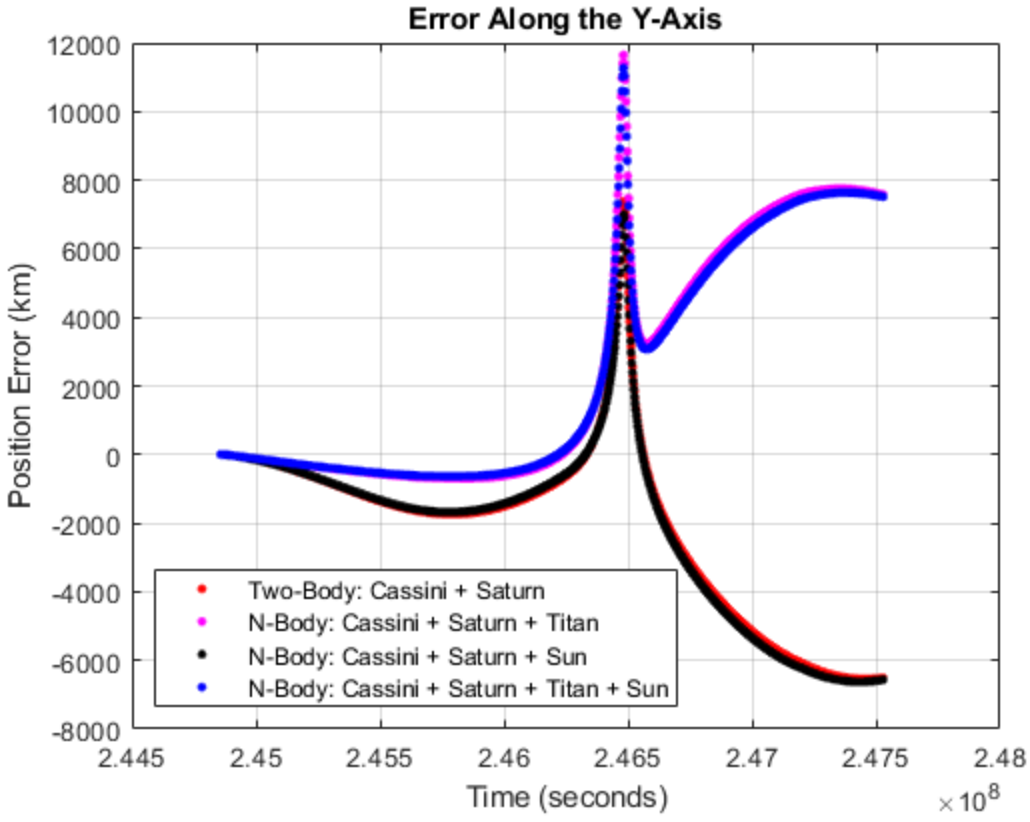
```
saturn_r = 60270;
surf(saturn_r*xs,saturn_r*ys,saturn_r*zs)
title('Cassini+Saturn+Sun System')
hold off
%-----------------------------------------------
%figure(5)
plot3(rB3(:,1),rB3(:,2),rB3(:,3),'k'), axis equal, grid on
[xs,ys,zs] = sphere(50);
hold on
saturn_r = 60270;
surf(saturn_r*xs,saturn_r*ys,saturn_r*zs)
title('Cassini+Saturn+Titan+Sun System')
hold off
%}
```

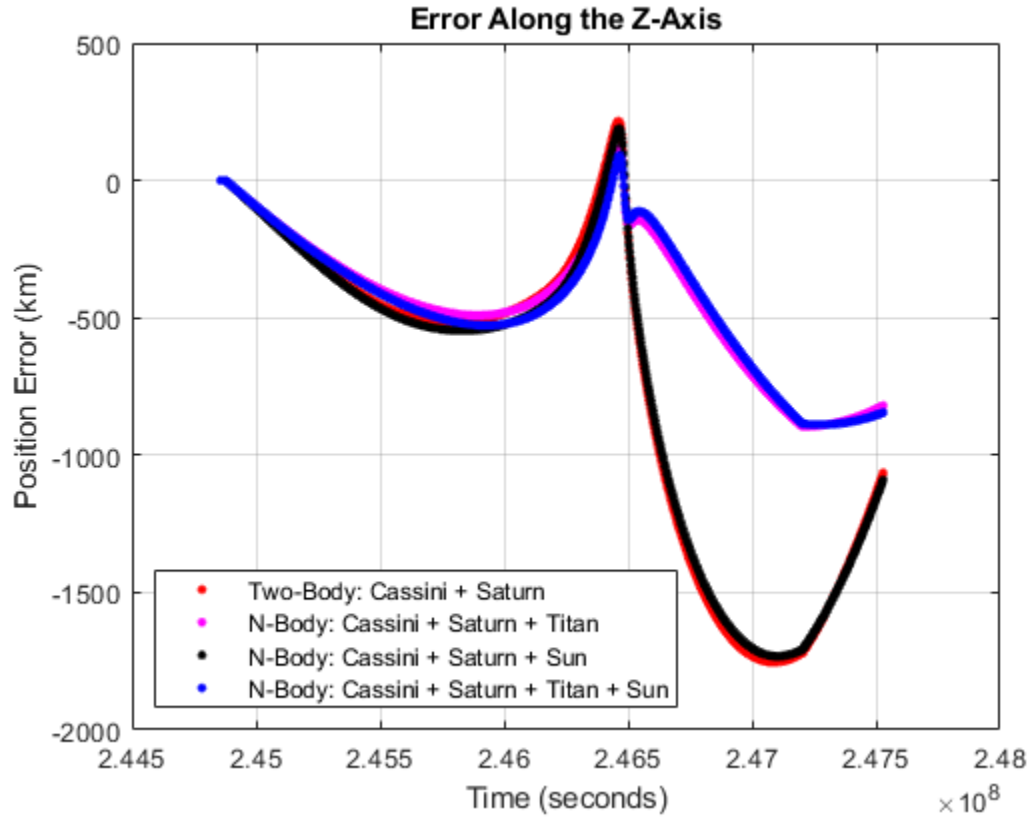
Part e: Calculate error along each axis

```
r_cassini = stateR';
e_2body = r - r_cassini;
e_CST = rB1 - r_cassini;
e_CSS = rB2 - r_cassini;
e_CSTS = rB3 - r_cassini;
%----------i direction
figure(6)
plot(time,-e_2body(:,1),'.r','MarkerSize',10), grid on
hold on
plot(time,-e_CST(:,1),'.m','MarkerSize',10), grid on
hold on
plot(time,-e_CSS(:,1),'.k','MarkerSize',10), grid on
hold on
plot(time,-e_CSTS(:,1),'.b','MarkerSize',10), grid on
hold off
title('Error Along the X-Axis')
l1 = 'Two-Body: Cassini + Saturn';
l2 = 'N-Body: Cassini + Saturn + Titan';
l3 = 'N-Body: Cassini + Saturn + Sun';
l4 = 'N-Body: Cassini + Saturn + Titan + Sun';
xlabel('Time (seconds)')
ylabel('Position Error (km)')
legend(l1,l2,l3,l4,'Location','northwest')
%------------------- j direction
figure(7)
plot(time,-e_2body(:,2),'.r','MarkerSize',10), grid on
hold on
plot(time,-e_CST(:,2),'.m','MarkerSize',10), grid on
hold on
plot(time,-e_CSS(:,2),'.k','MarkerSize',10), grid on
hold on
plot(time,-e_CSTS(:,2),'.b','MarkerSize',10), grid on
hold off
title('Error Along the Y-Axis')
l1 = 'Two-Body: Cassini + Saturn';
l2 = 'N-Body: Cassini + Saturn + Titan';
```

```
l3 = 'N-Body: Cassini + Saturn + Sun';
l4 = 'N-Body: Cassini + Saturn + Titan + Sun';
xlabel('Time (seconds)')
ylabel('Position Error (km)')
legend(l1,l2,l3,l4,'Location','southwest')
%------------------------------ k direction
figure(8)
plot(time,e_2body(:,3),'.r','MarkerSize',10), grid on
hold on
plot(time,e_CST(:,3),'.m','MarkerSize',10), grid on
hold on
plot(time,e_CSS(:,3),'.k','MarkerSize',10), grid on
hold on
plot(time,e_CSTS(:,3),'.b','MarkerSize',10), grid on
hold off
title('Error Along the Z-Axis')
l1 = 'Two-Body: Cassini + Saturn';
l2 = 'N-Body: Cassini + Saturn + Titan';
l3 = 'N-Body: Cassini + Saturn + Sun';
l4 = 'N-Body: Cassini + Saturn + Titan + Sun';
xlabel('Time (seconds)')
ylabel('Position Error (km)')
legend(l1,l2,l3,l4,'Location','southwest')
```

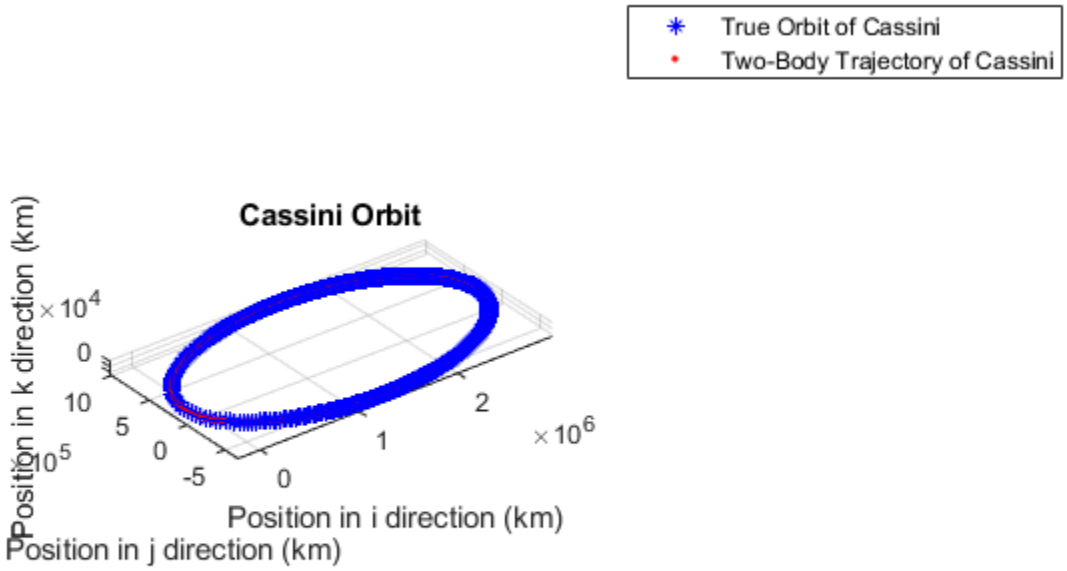



Error Along the Y-Axis
Position Error (km)
Time (seconds)
×10^8
Two-Body: Cassini + Saturn
N-Body: Cassini + Saturn + Titan
N-Body: Cassini + Saturn + Sun
N-Body: Cassini + Saturn + Titan + Sun


Error Along the Z-Axis
Position Error (km)
Time (seconds)
×10^8
Two-Body: Cassini + Saturn
N-Body: Cassini + Saturn + Titan
N-Body: Cassini + Saturn + Sun
N-Body: Cassini + Saturn + Titan + Sun

Part f: Create a 3D plot of the true orbit of Cassini and the two-body trajectory.

```
figure(9)
plot3(stateR(1,:),stateR(2,:),stateR(3,:),'b*'), axis equal, grid on
hold on
v1 = 'True Orbit of Cassini';
plot3(r(:,1),r(:,2),r(:,3),'r.'), axis equal, grid on
[xs,ys,zs] = sphere(50);
v2 = 'Two-Body Trajectory of Cassini';
title('Cassini Orbit')
legend(v1,v2)
xlabel('Position in i direction (km)')
ylabel('Position in j direction (km)')
zlabel('Position in k direction (km)')
hold off
```



Published with MATLAB® R2020a

```
function xdot = TwoBody(t,x)
%Cassini + Saturn
m2 =2150; %mass of cassini orbiter in kg
m1 = 568.5e24; %mass of Saturn in kg
G = 6.67408e-11; %gravitational constant
mu = (G*(m1+m2)) / (1000)^3; %km^3/s^2

r = x(1:3); %position vector of cassini relative to saturn
v = x(4:6); %velocity vector of cassini relative to saturn

rho = norm(r);

a = (-mu/rho^3)*r; %acceleration of cassini relative to saturn
xdot = [v;a];
```

```
Not enough input arguments.

Error in TwoBody (line 8)
r = x(1:3); %position vector of cassini relative to saturn
```

Published with MATLAB® R2020a

```
function xdot = nBody(t,x)
% SATURN + CASSINI  + TITAN

%%Part d: Propagate position and velocity of Cassini forward in time
 using n-body equation
m1 =568.5e24; %mass of Saturn in kg
m2 =2150; %mass of cassini orbiter in kg
m3 = 1.345e23;% mass of Titan in kg
G = 6.67408e-11; %gravitational constant
mu = (G*(m1+m2)) / (1000)^3; %mu Cassini/Saturn in km^3/s^2
mu3 = (G*m3) / (1000)^3;%mu for Titan
%-----------------------------------------
%r32: radius from titan to cassini
%r31: radius from titan to saturn
%r12: radius from saturn to cassini -- x(1:3)
[state31, lt31] = cspice_spkezr('SATURN', t , 'J2000', 'LT', 'TITAN');
r31 = state31(1:3); %radius from titan to saturn

r32 = r31 + x(1:3); %radius from titan to cassini

%-----------------------------------------------

r = x(1:3); %radius from saturn to cassini
v = x(4:6);%velocity  of cassini
rho = norm(r);

rho32 = norm(r32);
rho31 = norm(r31);
r_hat32 = r32 / rho32;
r_hat31 = r31 / rho31;

termP1 = mu3 *( (r_hat32/(rho32)^2) - (r_hat31/
(rho31)^2) ); %Permutation term


a = ( (-mu/rho^3)*r ) - termP1; %acceleration of cassini relative to
 saturn
xdot = [v;a];
```

```
Not enough input arguments.

Error in nBody (line 15)
[state31, lt31] = cspice_spkezr('SATURN', t , 'J2000', 'LT',
 'TITAN');
```

Published with MATLAB® R2020a

```
function xdot = nBody2(t,x)
% CASSINI + SATURN + SUN
%Part d: Propagate position and velocity of Cassini forward in time
 using n-body equation
m1 =568.5e24; %mass of Saturn in kg;
m2 =2150; %mass of cassini orbiter in kg
m3= 1.989e30; %mass of the Sun in kg
G = 6.67408e-11; %gravitational constant
mu = (G*(m1+m2)) / (1000)^3; %mu Cassini/Saturn in km^3/s^2
mu3 = (G*m3) / (1000)^3; %mu for Sun

%-----------------------------------------
%r32: radius from sun to cassini
%r31: radius from sun  to saturn

[state31, lt31] = cspice_spkezr('SATURN',t , 'J2000', 'LT', 'SUN');
r31 = state31(1:3); %radius from sun  to saturn
r32 = r31 + x(1:3);%radius from sun to cassini

%-----------------------------------------------

r = x(1:3); %radius from cassini to saturn
v = x(4:6); %velocity of cassini relative to saturn
rho = norm(r);

rho32 = norm(r32);
rho31 = norm(r31);
r_hat32 = r32 / rho32;
r_hat31 = r31 / rho31;

termP1 = mu3*( (r_hat32/(rho32)^2) - (r_hat31/
(rho31)^2) ); %Permutation term

a = ((-mu/rho^3)*r) - termP1; %acceleration of cassini relative to
 saturn
xdot = [v;a];
```

```
Not enough input arguments.

Error in nBody2 (line 15)
[state31, lt31] = cspice_spkezr('SATURN',t , 'J2000', 'LT', 'SUN');
```

Published with MATLAB® R2020a

```
function xdot = nBody3(t,x)
%CASSINI + SATURN + TITAN + SUN
%Part d: Propagate position and velocity of Cassini forward in time
 using n-body equation
m1 =568.5e24; %mass of Saturn in kg;
m2 =2150; %mass of cassini orbiter in kg
m3 = 1.345e23;% mass of Titan in kg
m4 = 1.989e30; %mass of the Sun in kg
G = 6.67408e-11; %gravitational constant
mu = (G*(m1+m2)) / (1000)^3; %mu Cassini/Saturn in km^3/s^2
mu3 = (G*m3) / (1000)^3;%mu for Titan
mu4 = (G*m4) / (1000)^3; %mu for Sun

%----------------------------------------
%r32: radius from titan to cassini
%r31: radius from titan to saturn
%r42: radius from sun to cassini
%r41: radius from sun  to saturn

[state31, lt31] = cspice_spkezr('SATURN',
 t , 'J2000', 'LT', 'TITAN'); %GOOD
r31 = state31(1:3); %radius from titan to saturn

r32 = r31 + x(1:3); %radius from titan to cassini

[state41, lt41] =
 cspice_spkezr('SATURN',t , 'J2000', 'LT', 'SUN'); %GOOD
r41 = state41(1:3); %radius from sun  to saturn

r42 = r41 + x(1:3); %radius from sun to cassini

%-----------------------------------------------

r = x(1:3);%radius from cassini to saturn
v = x(4:6); %acceleration of cassini relative to saturn
rho = norm(r);

rho32 = norm(r32);
rho31 = norm(r31);
r_hat32 = r32 / rho32;
r_hat31 = r31 / rho31;
termP1 = mu3 *( (r_hat32/(rho32)^2) - (r_hat31/(rho31)^2) );%first
 permutation term - titan

rho42 = norm(r42);
rho41= norm(r41);
r_hat42 = r42 / rho42;
r_hat41 = r41 / rho41;
termP2= mu4 *( (r_hat42/(rho42)^2) - (r_hat41/(rho41)^2) ); %second
 permutation term - sun
```

```
a = ((-mu/rho^3)*r) - (termP1 + termP2); %acceleration of cassini
 relative to saturn
xdot = [v;a];

Not enough input arguments.

Error in nBody3 (line 19)
[state31, lt31] = cspice_spkezr('SATURN', t , 'J2000', 'LT', 'TITAN');
 %GOOD
```

Published with MATLAB® R2020a