
Orbital elements to radius/velocity

By: Noe Lepez Da Silva Duarte Created: 08 Feb. 2022

```
function [r, v] = oe2eci(a,e,inc,RAAN,omega,nu,mu)
% This function computes the radius vector (ECI) for a given set of orbital
% elements
% - a = Semi-major axis [m]
% - e = orbit of interest eccentricity.
% - inc = inclination [deg]
% - RAAN = Right Ascension of Ascending Node [deg]
% - omega = Argument of Perigee [deg]
% - nu = True anomaly [deg].
% - mu = gravitational parameter [km^3/s^2]
%
% The function output 'r' in km and 'v' in km/s
%

nu = deg2rad(nu);

% Calculate r in the perifocal coordinate frame
r = (a*(1-e^2))/(1+e*cos(nu));
r_vec = [r*cos(nu);r*sin(nu);0];

% Calculate v in the perifocal coordinate frame
p = a * (1 - e^2);
term_1 = sqrt(mu/p);
v_vec = [-term_1*sin(nu);term_1*(e+cos(nu));0];

% Make orbital elements into radians for MATLAB
omega_rad = deg2rad(omega);
i_rad = deg2rad(inc);
RAAN_rad = deg2rad(RAAN);

% Create a array of rotation angles to rotate the r vector obtained in the
% perifocal from into the ECI frame
angle_vec = [-omega_rad -i_rad -RAAN_rad];
total_rot_r = r_vec;
total_rot_v = v_vec;

% Loop through the 3-1-3 rotations
i=1;
for j=[3 1 3]
    if j == 1
        inter_rot = Rx(angle_vec(i));
    elseif j == 3
        inter_rot = Rz(angle_vec(i));
    end
    total_rot_r = inter_rot*total_rot_r;
    total_rot_v = inter_rot*total_rot_v;
    i = i+1;
end
```

```
% Create outputs
r = total_rot_r;
v = total_rot_v;

% Rotation matrices as function as to be easily reused
function R1=Rx(ph)
    R1 = [1 0 0;
          0 cos(ph) sin(ph);
          0 -sin(ph) cos(ph)];
end

function R3=Rz(ps)
    R3 = [cos(ps) sin(ps) 0;
          -sin(ps) cos(ps) 0;
          0 0 1];
end

end
```

Published with MATLAB® R2021b