```
function x = propagateState(oe0,t,t_0,mu,J2,Re)
%DESCRIPTION: Computes the propagated position and velocity in km, km/
s
%accounting for approximate J2 perturbations
%
%INPUTS:
% oe0       Orbit elements [a,e,i,Om,om,f] at time t0 (km,s,rad)
% t         Current time (s)
% t0        Time at the initial epoch (s)
% MU        Central body's gravitational constant (km^3/s^2)
% J2        Central body's J2 parameter (dimensionless)
% Re        Radius of central body (km)
%
%OUTPUTS:
% x         Position and velocity vectors of the form [r; rdot] (6x1)
 at
%             time t


%make sure that function has outputs
x = NaN(6,1);

%1) Compute the mean orbit elements oe(t) at time t due to J2
 perturbations
    a = oe0(1);
    e = oe0(2);
    i = oe0(3);
    Om = oe0(4);
    om = oe0(5);
    f = oe0(6);
    Om_dot = -((3/2)*((sqrt(mu)*J2*Re^2)/(2*(1-e^2)^2*a^(7/2)))) *
 cos(i);
    om_dot = Om_dot * ((5/2)*sin(i)^2 - 2) / cos(i);

    % Effects of peterbations
    Om = Om + Om_dot * (t - t_0);
    om = om + om_dot * (t - t_0);

    % Normalize to domain
    Om = Om - 2*pi*(floor(Om/(2*pi)));
    om = om - 2*pi*(floor(om/(2*pi)));

%2) Solve the time-of-flight problem to compute the true anomaly at
 tiem t
    h = sqrt(mu*a*(1-e^2));
    n = sqrt(mu/a^3);
    P = 2*pi/n;
    M = @(t) 2*pi/P*t;
    M_ = M(t-t_0);
    tol = 1e-9;

    [E_f,f_f] = keptof(2*pi*i/length(t),e,M_,tol);
```

```
%3) Compute r(t), rdot(t) in the perifocal frame
    r = a*(1-e^2) / (1 + e*cos(f));

    %Position
    r_f = [r*cos(f_f) r*sin(f_f) 0]';

    % Velocity
    r_dot_f = (mu/h)*[-sin(f_f) e+cos(f_f) 0]';

%4) Compute r(t), rdot(t) in the ECI frame, save into x
    PN = angle2dcm(Om,i,om,'ZXZ');

    r_ECI = PN' * r_f;
    r_dot_ECI = PN * r_dot_f;

    x = [r_ECI;r_dot_ECI];

function [E,f] = keptof(E_0,e,M,tol)
f_calc = @(E) 2*atan(sqrt((1+e)/(1-e)) * tan(E/2));
ratio = 1;
E_ = E_0;
%f_ = f_calc(E_0);

%i = 2;
    while ratio > tol
        ratio = (E_ - e*sin(E_) - M)/(1 - e*cos(E_));
        E_ = E_ - ratio;
        %f_(i) = f_calc(E_(i));
        %i = i+1;
    end
%iteration = 1:i-1;
E = E_;
f = f_calc(E_);
end
end
```

```
Not enough input arguments.

Error in propagateState (line 22)
    a = oe0(1);
```

Published with MATLAB® R2020a