ORBITAL PERIODS OF A SATELLITE

This section describes several MATLAB scripts and a function that can be used to calculate the orbital periods of a satellite. In orbital mechanics we are concerned with the following types of orbital periods:

- *Keplerian* – the unperturbed or two-body period
- *Nodal* – the time interval from one ascending (or descending) node to the next
- *Anomalistic* – the time interval from one perigee to the next
- *Sidereal* – the time interval from one value of argument of latitude to the next identical value

Several of the applications in this section propagate satellite orbits by numerical integration during the solution process. Each of these methods uses a J_2 perturbed form of the orbital equations of motion. These routines can easily be modified to include higher fidelity gravity models and other perturbations such as aerodynamic drag and solar radiation pressure.

period1.m – orbital periods, Kozai J2 method

This MATLAB script can be used to estimate the Keplerian, nodal, anomalistic and sidereal periods of an Earth satellite. These calculations include the effect of the main oblateness gravity term J_2 on the orbital period.

The anomalistic period of a satellite based on *osculating orbital elements at perigee* is given by

$$\tau_a = 2\pi\sqrt{\frac{a^3}{\mu}}\left\{1-\frac{3J_2 r_{eq}^2}{2a^2\left(1-e^2\right)^3}\left(1-3\sin^2 i\sin^2\omega\right)\right\}$$

The nodal period of a satellite based on *osculating orbital elements at the ascending node* is

$$\tau_n = 2\pi\sqrt{\frac{a^3}{\mu}}\left\{1-\frac{3J_2\left(4-5\sin^2 i\right)}{4\left(a/r_{eq}\right)^2\sqrt{1-e^2}\left(1+e\cos\omega\right)^2}-\frac{3J_2\left(1+e\cos\omega\right)^3}{2\left(a/r_{eq}\right)^2\left(1-e^2\right)^3}\right\}$$

The sidereal period of a satellite based on *osculating orbital elements at perigee* is given by this next expression.

$$\tau_s = 2\pi\sqrt{\frac{a^3}{\mu}}\left\{1-\frac{3}{2}J_2\left(\frac{ar_{eq}^2}{r^3}\right)\left(1-3\sin^2 i\sin^2 u\right)-\frac{3}{4}J_2\left(\frac{r_{eq}}{a}\right)^2\frac{1}{\sqrt{1-e^2}}\left[\frac{4-5\sin^2 i}{\left(1+e\cos\theta\right)^2}\right]\right\}$$

In these equations

$$a = \text{semimajor axis}$$
$$e = \text{orbital eccentricity}$$
$$i = \text{orbital inclination}$$
$$\omega = \text{argument of perigee}$$
$$\Omega = \text{right ascension of the ascending node}$$
$$\theta = \text{true anomaly}$$
$$u = \text{argument of latitude} = \omega + \theta$$
$$r = \text{geocentric radius} = a\left(1 - e^2\right) / 1 + e\cos\theta$$
$$J_2 = \text{Earth oblateness gravity coefficient}$$
$$\mu = \text{Earth gravitational constant}$$
$$r_{eq} = \text{Earth equatorial radius}$$

The following is a typical user interaction with this MATLAB script.

```
               program period1

    < orbital periods - j2 analytic solution >


   please input the semimajor axis (kilometers)
   (semimajor axis > 0)
   ? 8000

   please input the orbital eccentricity (non-dimensional)
   (0 <= eccentricity < 1)
   ? 0.015

   please input the orbital inclination (degrees)
   (0 <= inclination <= 180)
   ? 28.5

   please input the argument of perigee (degrees)
   (0 <= argument of perigee <= 360)
   ? 270

   please input the true anomaly (degrees)
   (0 <= true anomaly <= 360)
   ? 30
```

The following is a typical output created with this application.

```
       < orbital periods - j2 solution >

   semimajor axis         8000.000000  kilometers

   eccentricity (nd)         0.015000

   inclination              28.500000  degrees
```

```
argument of perigee     270.000000  degrees

true anomaly             30.000000  degrees


Keplerian period        118.684684  minutes

nodal period            118.386783  minutes

anomalistic period      118.644052  minutes

sidereal period         118.451692  minutes
```

period2.m – nodal period, numerically integrated motion

This MATLAB script can be used to estimate the nodal period of an Earth satellite. The satellite orbit is numerically integrated during the solution process while searching for *time* roots of the following nonlinear scalar function:

$$f\left(t\right)=\frac{r_z}{r_m}=0$$

In this expression r_z is the *z*-component of the satellite's position vector and r_m is the scalar geocentric distance of the satellite at any simulation time *t*. During the solution process the value of the *z* component of the satellite's velocity is also checked. Whenever this value is positive, the satellite is ascending in its orbit and the root of the *objective function* defined above ensures an ascending node crossing.

The following is a typical user interaction with this script.

```
              program period2

   < nodal period - integrated solution >


please input the semimajor axis (kilometers)
(semimajor axis > 0)
? 8000

please input the orbital eccentricity (non-dimensional)
(0 <= eccentricity < 1)
? .015

please input the orbital inclination (degrees)
(0 <= inclination <= 180)
? 28.5

please input the argument of perigee (degrees)
(0 <= argument of perigee <= 360)
? 270
```

The following is the program output created by this example.

```
                 program period2

      < nodal period - integrated solution >

   semimajor axis          8000.000000  kilometers

   eccentricity (nd)          0.015000

   inclination               28.500000  degrees

   argument of perigee      270.000000  degrees

   raan                       0.000000  degrees

   Keplerian period         118.684684  minutes

   nodal period             118.386635  minutes
```

period3.m – anomalistic period, numerically integrated motion

This MATLAB script can be used to estimate the anomalistic period of an Earth satellite. The satellite orbit is numerically integrated during the solution process while searching for *time* roots of the following nonlinear scalar function:

$$f(t) = \theta = 0$$

In this expression θ is the true anomaly of the satellite at any simulation time t.

The following is a typical user interaction with this script.

```
                 program period3

      < anomalistic period - integrated solution >


   please input the semimajor axis (kilometers)
   (semimajor axis > 0)
   ? 8000

   please input the orbital eccentricity (non-dimensional)
   (0 <= eccentricity < 1)
   ? 0.015

   please input the orbital inclination (degrees)
   (0 <= inclination <= 180)
   ? 28.5

   please input the argument of perigee (degrees)
   (0 <= argument of perigee <= 360)
   ? 270
```

```
please input the right ascension of the ascending node (degrees)
(0 <= raan <= 360)
? 0
```

The following is the program output created by this example.

```
              program period3

    < anomalistic period - integrated solution >

semimajor axis        8000.000000  kilometers

eccentricity (nd)        0.015000

inclination             28.500000  degrees

argument of perigee    270.000000  degrees

raan                     0.000000  degrees


Keplerian period       118.684684  minutes

anomalistic period     118.642220  minutes
```

period4.m – sidereal period, numerically integrated motion

This script can be used to estimate the sidereal period of an Earth satellite. The satellite orbit is numerically integrated during the solution process while searching for *time* roots of the following nonlinear scalar objective function:

$$f\left(t\right) = u = \omega + \theta = 0$$

In this expression u is the argument of latitude, θ is the true anomaly and ω is the argument of perigee of the satellite at any simulation time t.

The following is a typical user interaction with this script.

```
                  program period4

    < sidereal period - integrated solution >


please input the semimajor axis (kilometers)
(semimajor axis > 0)
? 8000

please input the orbital eccentricity (non-dimensional)
(0 <= eccentricity < 1)
? 0.015

please input the orbital inclination (degrees)
```

```
(0 <= inclination <= 180)
? 28.5

please input the argument of perigee (degrees)
(0 <= argument of perigee <= 360)
? 270

please input the right ascension of the ascending node (degrees)
(0 <= raan <= 360)
? 0

please input the true anomaly (degrees)
(0 <= true anomaly <= 360)
? 30
```

The following is the program output created by this example.

```
              program period4

   < sidereal period - integrated solution >

semimajor axis        8000.000000  kilometers

eccentricity (nd)        0.015000

inclination             28.500000  degrees

argument of perigee    270.000000  degrees

raan                     0.000000  degrees

true anomaly            30.000000  degrees


Keplerian period       118.684684  minutes

sidereal period        118.451675  minutes
```

tperiod.m – nodal and anomalistic period function

This MATLAB function computes the nodal and anomalistic periods of a satellite subject to the gravity perturbation due to J_2.

The syntax of this MATLAB script is:

```
function [tnodal, tanomal] = tperiod (sma, ecc, inc, argper)

% orbital periods

% input

%  sma    = semimajor axis
%  ecc    = orbital eccentricity
%  inc    = orbital inclination (radians)
```

```
%  argper = argument of perigee (radians)

% output

%  tnodal  = nodal period (seconds)
%  tanomal = anomalistic period (seconds)
```