The Design and Analysis of Geosynchronous Orbits

This document describes several MATLAB scripts that are useful for performing mission design and analysis for Earth satellites in geosynchronous orbits. Geosynchronous satellites are usually in equatorial or low-inclination orbits with an orbital period very close to the time required for the Earth to complete one inertial rotation.

The algorithms implemented in these MATLAB scripts are based on the numerical techniques described in the following two articles:

"East-west Stationkeeping Requirements of Nearly Synchronous Satellites Due to Earth's Triaxiality and Luni-Solar Effects", A. Kamel, D. Ekman and R. Tibbitts, *Celestial Mechanics* **8** (1973) 129-148.

"On the Orbital Eccentricity Control of Synchronous Satellites", A. Kamel and C. Wagner, *The Journal of the Astronautical Sciences*, Vol. XXX, No. 1, pp. 61-73, January-March, 1982.

geosync1.m – equilibrium longitudes and radii of geosynchronous satellites

This MATLAB script calculates the *equilibrium* longitudes and radii of geosynchronous satellites based on the EGM96 gravity model of degree (zonals) and order (tesserals) 3. These are the four Earth-relative locations where the proper combination of geocentric radius and east longitude will minimize the longitudinal drift of satellites located at these points.

The following is the output created by this script.

```
                          program geosync1

          < equilibrium longitudes, radii and acceleration >


 location      east longitude         radius          drift acceleration
                 (degrees)          (kilometers)          (degrees/day^2)

    1              75.0602          42166.2409        -2.2287387749e-012

    2             162.0816          42166.2847        7.5875859013e-012

    3             255.0880          42166.2411        2.0642563986e-012

    4             348.5962          42166.2811        4.3719224598e-012


 drift cycle period at longitude   75.0602 degrees =  2339.2602 days

 drift cycle period at longitude  255.0880 degrees =  2883.5951 days
```

The very small values of drift acceleration confirm that these east longitude locations are indeed equilibrium points relative to a triaxial Earth.

The fundamental astrodynamic constants of the EGM96 gravity model are:

r_{eq} = Earth equatorial radius $= 6378.1363$ km

μ = Earth gravitational constant $= 398600.4415$ km^3/sec^2

ω = Earth inertial rotation rate $= 7.292115 \bullet 10^{-5}$ radians/sec

The spherical harmonic and longitude coefficients of the EGM96 gravity model are determined from the following expressions:

$$J_{nm} = -\sqrt{C_{nm}^2 + S_{nm}^2}$$

$$\lambda_{nm} = \frac{1}{m}\tan^{-1}\left(\frac{S_{nm}}{C_{nm}}\right)$$

where *n* is the degree and *m* is the order of these terms.

The first few unnormalized gravity coefficients are given by

$$\begin{aligned} C_{22} &= 1.57446037456 \bullet 10^{-6} \\ S_{22} &= -9.03803806639 \bullet 10^{-7} \\ &\vdots \\ C_{31} &= 2.19263852917 \bullet 10^{-6} \\ S_{31} &= 2.68424890297 \bullet 10^{-7} \end{aligned}$$

The geocentric radius of a synchronous satellite is given by

$$a_s = a_{sk} + \delta a_s$$

where a_{sk} is the Keplerian or unperturbed geosynchronous semimajor axis which is given by

$$a_{sk} = \left(\frac{\mu}{\omega_e^2}\right)^{1/3}$$

In this equation μ is the gravitational constant of the Earth and ω_e is the inertial rotation rate of the Earth. The "correction" of the Keplerian semimajor axis due to the triaxial shape of the Earth and a gravity model of degree and order 3 for a satellite located at an east longitude denoted by λ_s is given by the expression

$$\begin{aligned} \delta a_s = {} & 2J_{20}a_{sk}\left(\frac{r_{eq}}{a_{sk}}\right)^2 + 12J_{22}a_{sk}\left(\frac{r_{eq}}{a_{sk}}\right)^2 \cos 2\left(\lambda_s - \lambda_{22}\right) \\ & -8J_{31}a_{sk}\left(\frac{r_{eq}}{a_{sk}}\right)^3 \cos 3\left(\lambda_s - \lambda_{31}\right) + 80J_{33}a_{sk}\left(\frac{r_{eq}}{a_{sk}}\right)^3 \cos 3\left(\lambda_s - \lambda_{33}\right) \end{aligned}$$

The equilibrium longitudes are the points at which the ratio g_1/g_2 is equal to zero. The "g" factors g_1 and g_2 are determined from the following two equations:

$$g_1 = 6J_{22}\left(\frac{r_{eq}}{a_{sk}}\right)^2 \sin 2\left(\lambda_s - \lambda_{22}\right) - \frac{3}{2}J_{31}\left(\frac{r_{eq}}{a_{sk}}\right)^3 \sin\left(\lambda_s - \lambda_{31}\right)$$
$$+45J_{33}\left(\frac{r_{eq}}{a_{sk}}\right)^3 \sin 3\left(\lambda_s - \lambda_{33}\right)$$

$$g_2 = \frac{\partial g_1}{\partial \lambda_s} = 12J_{22}\left(\frac{r_{eq}}{a_{sk}}\right)^2 \cos 2\left(\lambda_s - \lambda_{22}\right) - \frac{3}{2}J_{31}\left(\frac{r_{eq}}{a_{sk}}\right)^3 \cos\left(\lambda_s - \lambda_{31}\right)$$
$$+135J_{33}\left(\frac{r_{eq}}{a_{sk}}\right)^3 \cos 3\left(\lambda_s - \lambda_{33}\right)$$

The *normalized* longitudinal acceleration can be determined from the following expression:

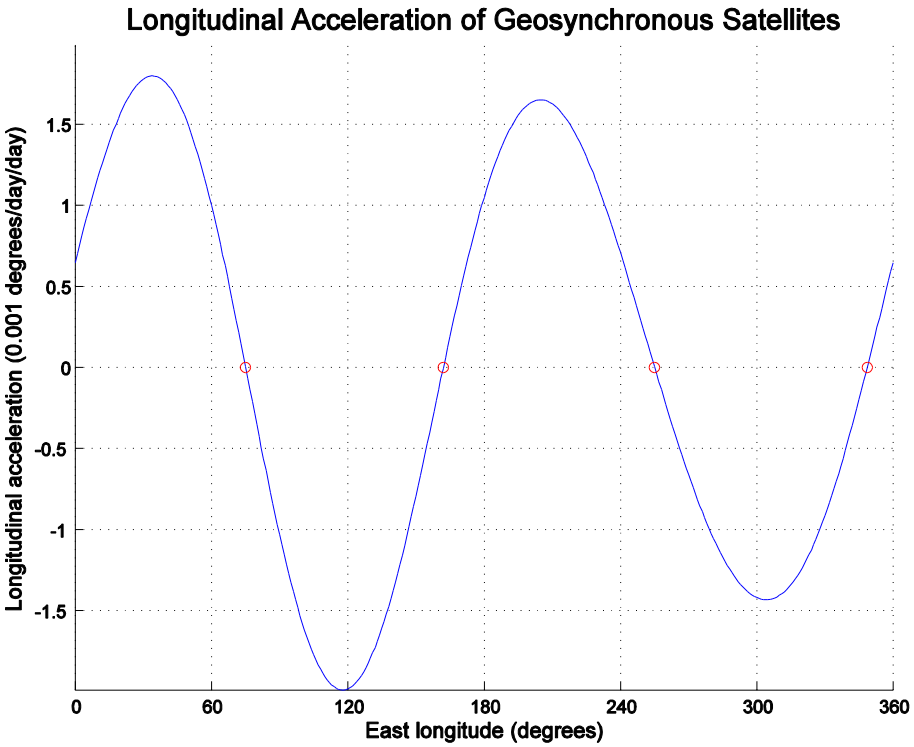
$$\ddot{\bar{\lambda}} = 18\left\{c_{22}\sin 2\left(\lambda_s - \lambda_{22}\right) - 0.25c_{31}\sin\left(\lambda_s - \lambda_{31}\right) + 7.5c_{33}\sin 3\left(\lambda_s - \lambda_{33}\right)\right\}$$

In this equation $c_{nm} = J_{nm}\left(r_{eq}/a_s\right)^n$. The dimensional longitude acceleration is $\ddot{\bar{\lambda}}\omega_e^2$. The equilibrium longitudes are determined using Brent's root-finding algorithm.

Equilibrium longitudes with negative g_1 values are stable while those with positive g_1 values are unstable. The equilibrium longitudes located at 75.0602° and 255.088° are stable points with drift cycle periods given by

$$\tau = \frac{\pi}{\sqrt{-3g_2}}$$

After this MATLAB script has calculated the characteristics of the equilibrium points, it will ask if you would like to create and display a plot of the longitudinal acceleration as a function of east longitude of the satellite. This information can be used to graphically verify the calculations described in this section. The following is the graphics display for this program option. The location of each equilibrium point computed by this script is marked by a small red circle.



geosync2.m – required osculating semimajor axis

This MATLAB script can be used to estimate the initial osculating semimajor axis required for an Earth satellite in geosynchronous orbit. The calculations include perturbations due to the point-mass gravity of the Sun and Moon and non-spherical Earth gravity. The user can also elect to include the effect of solar radiation pressure in the calculations. The script also allows the user to specify the degree and order of the Earth gravity model to use during the solution.

The numerical method implemented in this script uses a combination of one-dimensional root-finding and numerical integration of the orbital equations of motion to determine the initial osculating semimajor axis that will result in a satellite orbit with a geosynchronous period subject to the perturbations mentioned above.

The one-dimensional objective function used during the root-finding calculations is

$$f(t) = \lambda_m - \lambda_{m_0}$$

where λ_m is the mean east longitude of the satellite's subpoint at any time *t* and λ_{m_0} is mean east longitude at the initial time.

The true east longitude of the satellite subpoint at any time is given by

$$\lambda_t(t) = \omega + \Omega + \theta - \alpha_g$$

and the mean east longitude is

$$\lambda_m(t) = \omega + \Omega + M - \alpha_g$$

In these two equations, ω is the argument of perigee, Ω is the right ascension of the ascending node, θ is the satellite's true anomaly, *M* is the mean anomaly and α_g is the Greenwich sidereal time at any simulation time *t*.

The relationship between mean and true anomaly is given by Kepler's equation for elliptic orbits,

$$M = E - e\sin E$$

via the following intermediate calculations for eccentric anomaly:

$$\sin E = \sin\theta\sqrt{1-e^2}$$
$$\cos E = e + \cos\theta$$
$$E = \tan^{-1}(\sin E, \cos E)$$

The inverse tangent calculation in the last equation is a four quadrant operation.

The orbital period of a geosynchronous satellite in seconds is given by

$$\tau = \frac{2\pi}{\omega_e}$$

where ω_e is the inertial rotation rate of the Earth in radians per second. The algorithm numerically integrates the equations of motion for one or more orbital periods, evaluates the mean east longitude objective function and uses Brent's root-finding method to drive the difference to a small tolerance. The user can specify how many orbital periods to use during the solution process. More orbital periods will provide a better semimajor axis solution at the expense of longer computation times. A value between 1 and 5 orbits is recommended.

During the solution the algorithm uses a bracketing interval for the required osculating semimajor axis *a* given by

$$a_0 - 100 \le a \le a_0 + 100$$

where a_0 is the initial semimajor axis guess (in kilometers) provided by the user.

The `geosync2.m` script will begin by prompting the user for the initial calendar date and universal time. It will then ask for the degree and order of the gravity model and which perturbations to include in the simulation. Finally, the script will prompt the user for the initial orbital elements of the satellite. If the initial orbit is not equatorial (orbital inclination > 0), the software assumes that the true east longitude input by the user is at the ascending node.

The following is a typical interaction with this script.

```
                  program geosync2

   < geosynchronous osculating semimajor axis >


initial calendar date and time

please input the calendar date
(1 <= month <= 12, 1 <= day <= 31, year = all digits!)
? 1,1,1998

please input the universal time
(0 <= hours <= 24, 0 <= minutes <= 60, 0 <= seconds <= 60)
? 0,0,0


gravity model inputs


please input the degree of the gravity model (zonals)
(0 <= zonals <= 18)
? 4

please input the order of the gravity model (tesserals)
(0 <= tesserals <= 18)
? 4


orbital perturbations


would you like to include solar perturbations (y = yes, n = no)
? y

would you like to include lunar perturbations (y = yes, n = no)
? y

would you like to include srp perturbations (y = yes, n = no)
? y


solar radiation pressure inputs


please input the reflectivity constant (non-dimensional)
? 1.85

please input the cross-sectional area (square meters)
? 10

please input the spacecraft mass (kilograms)
? 2000


please input the numerical integration error tolerance
(a value between 1.0e-8 and 1.0e-10 is recommended)
? 1e-8
```

```
orbital elements menu

   <1> user input

   <2> data file

? 1


initial orbital elements


please input the semimajor axis (kilometers)
(semimajor axis > 0)
? 42160

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

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

please input the satellite's east longitude (degrees)
(0 <= east longitude <= 360)
? 45


please input the number of orbits to model
(integer >= 1)
? 5
```

The software will display the initial and final east longitude and geodetic latitude. This information indicates how well the orbit "closed" at the solution. The following is the program output created for this example.

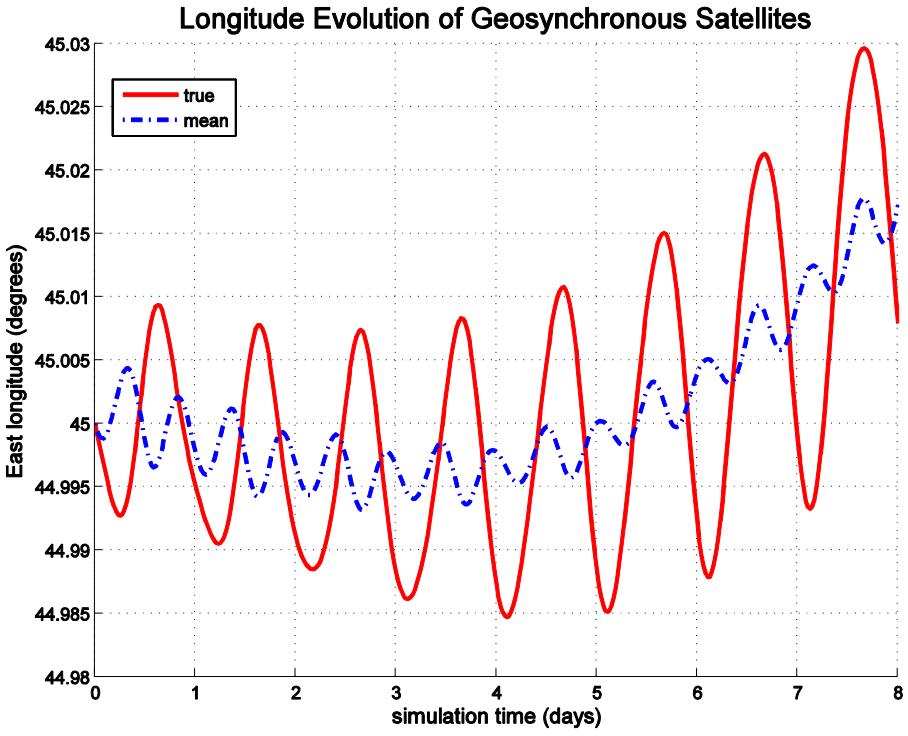
```
                  program geosync2

    < geosynchronous osculating semimajor axis >

initial calendar date          01-Jan-1998
initial universal time         00:00:00

initial mean east longitude     45.000000  degrees
final mean east longitude       45.000000  degrees

initial geodetic latitude        0.000000  degrees
final geodetic latitude          0.272835  degrees

reflectivity constant            1.850000

cross-sectional area            10.000000  sqr meters

spacecraft mass               2000.000000  kilograms

degree of gravity model          4.0
order of gravity model           4.0
```

```
number of orbits modeled          5.0

initial orbital elements

      sma (km)          eccentricity      inclination (deg)     argper (deg)
 4.2166908088e+004  0.0000000000e+000  0.0000000000e+000  0.0000000000e+000

     raan (deg)      true anomaly (deg)    arglat (deg)        period (min)
 0.0000000000e+000  1.4544413882e+002  1.4544413882e+002  1.4362080817e+003
```

After the script solves the problem it will ask the user if he or she would like to create a graphics display of the evolution of the satellite's east longitude. A typical user interaction with this program option is as follows:

```
would you like to create and display graphics (y = yes, n = no)
? y

please input the simulation period (days)
? 8

please input the graphics step size (minutes)
? 30
```

The following is a plot of the long-term evolution of both the true and mean east longitude for this example. It illustrates that the secular variation of the east longitude is small and only starts to "drift" after about 5 days which is the prediction period used in this example.



geosync3.m – repositioning a geosynchronous satellite

This MATLAB script determines the impulsive maneuvers required to reposition a geosynchronous satellite. Repositioning is the process of moving a geosynchronous satellite in longitude to the east or west relative to some initial location. This orbit modification is performed by creating an elliptical *drift* orbit with one impulsive maneuver and then applying a second impulse that re-establishes a circular geosynchronous orbit at the "target" longitude.

The semimajor axis of the elliptical drift orbit is given by

$$a_d = a_s\left(1+\frac{D}{360^\circ}\right)^{2/3}$$

where a_s is the semimajor axis of the initial geosynchronous circular orbit.

The drift rate *D* is given by

$$D = \Delta\lambda / n_d$$

In this equation $\Delta\lambda$ is the longitude change (+ west, $-$ east) relative to the initial location, and n_d is the integer number of drift orbits that the spacecraft travels during the change.

The longitude drift during one drift orbit is given by

$$\Delta\lambda = \omega_e\tau_d - \omega_s\tau_d$$

In this equation ω_e is the inertial rotation rate of the Earth, ω_s is the orbital rate of the drift orbit and τ_d is the orbital period of the drift orbit. The longitude drift is caused by the difference between the Earth rotation and orbital rates during the drift period.

For a longitude change to the <u>west</u>, the orbital eccentricity of the drift orbit is determined from

$$e_d = 1-\frac{a_s}{a_d}$$

The perigee velocity of the drift orbit is given by

$$V_p = \sqrt{\frac{2\mu}{a_s}-\frac{\mu}{a_d}}$$

The magnitude of the delta-v that creates the elliptical drift orbit is

$$\Delta V = V_p - V_{lc}$$

where V_{lc} is the local circular velocity of the initial geosynchronous orbit, $V_{lc} = \sqrt{\mu/a_s}$. This delta-v is applied in the direction of orbital motion and creates an elliptical orbit that has a longer period than the

original circular orbit. After the satellite has drifted for n_d orbits, a delta-v of equal magnitude but opposite direction is applied. This delta-v re-circularizes the elliptical drift orbit and creates a geosynchronous orbit at the new longitude.

For a longitude change to the east, the orbital eccentricity of the drift orbit is determined from

$$e_d = \frac{a_s}{a_d} - 1$$

The apogee velocity of the drift orbit is given by

$$V_a = \sqrt{\frac{2\mu}{a_s} - \frac{\mu}{a_d}}$$

The magnitude of the delta-v that creates the elliptical drift orbit is

$$\Delta V = V_{lc} - V_a$$

This delta-v is applied opposite to the direction of orbital motion and creates an elliptical orbit that has a shorter period than the original circular orbit. As before a delta-v of equal magnitude but opposite direction is applied to create a geosynchronous orbit at the new user-defined east longitude.

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

```
                program geosync3

 < repositioning maneuvers for gso satellites >


 please input the semimajor axis (kilometers)
 ? 42165

 please input the longitude change
 (+ west, - east; degrees)
 ? -30

 please input the number of drift orbits
 ? 10

                program geosync3

    < repositioning maneuvers for gso satellites >


 semimajor axis           42165.000000  kilometers

 delta-longitude            -30.000000  degrees

 number of orbits           10.0

 drift period               237.357151  hours
```

```
drift orbit characteristics

semimajor axis           41930.423442  kilometers

eccentricity (nd)            0.005594

perigee altitude         35317.709884  kilometers

apogee altitude          35786.863000  kilometers

keplerian period          1424.142906  minutes

total delta-v               17.224908  meters/second
```

geosync4.m – east-west stationkeeping of geosynchronous satellites

This MATLAB script can be used to determine the impulsive delta-v and drift cycle period required for east-west stationkeeping of geosynchronous satellites in equatorial orbits. The east-west stationkeeping requirement is specified by a longitude "deadband" centered about the nominal east longitude of the satellite.

A "drift" orbit is established by biasing the initial semimajor axis such that the satellite moves from this initial condition to the edge of the deadband. After reaching the edge of the deadband the satellite then drifts toward the other side. When the satellite reaches the other end of the deadband, a single impulsive maneuver is performed to create an elliptical orbit with an apogee equal to the original semimajor axis of the drift orbit. Once the satellite reaches this new apogee, another single maneuver is performed to circularize the satellite's orbit at this radius.

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

```
                    program geosync4

< east-west stationkeeping of geosynchronous satellites >


initial east longitude and deadband

please input the satellite's east longitude (degrees)
(0 <= east longitude <= 360)
? 45


please input the total deadband constraint (degrees)
? 1
```

The following is the program output for this example.

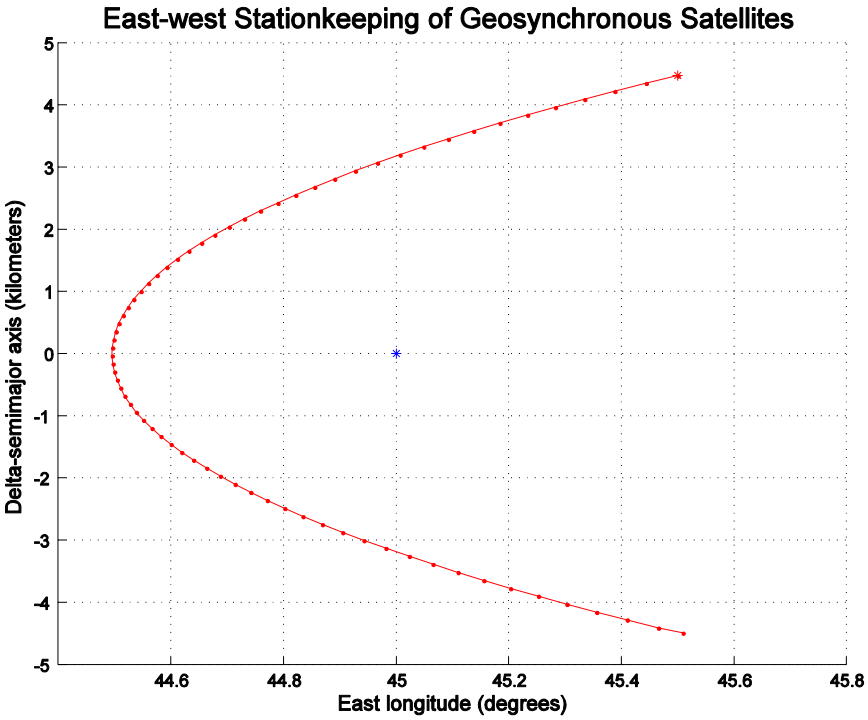
```
                    program geosync4

< east-west stationkeeping of geosynchronous satellites >

satellite east longitude            45.0000 degrees

total longitude deadband             1.0000 degrees
```

```
initial drift rate                   0.0575 degrees/day

single maneuver delta-v              0.3262 meters/second

total annual delta-v                 1.7113 meters/second/year

drift cycle period                  69.6248 days


geosynchronous semimajor axis    42166.2534 kilometers
drift cycle semimajor axis       42170.7272 kilometers
delta semimajor axis                 4.4738 kilometers
```

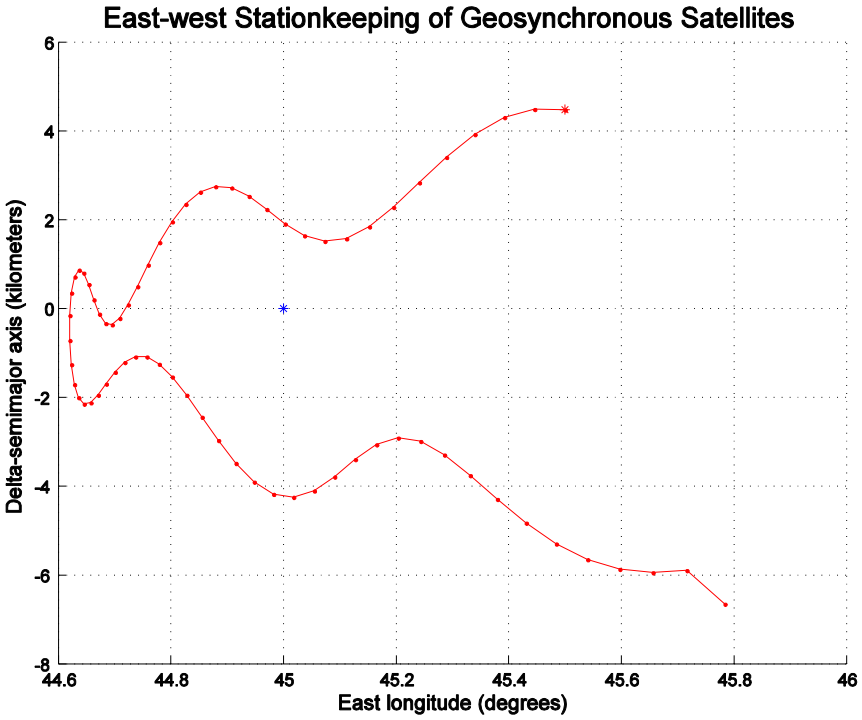
After the script has solved the problem it will ask the user if he or she would like to create a graphics display of the satellite's orbital motion during the stationkeeping. The following is a typical user response to this option:

```
would you like to create and display graphics (y = yes, n = no)
? y

initial calendar date and time

please input the calendar date
(1 <= month <= 12, 1 <= day <= 31, year = all digits!)
? 1,1,1998

please input the universal time
(0 <= hours <= 24, 0 <= minutes <= 60, 0 <= seconds <= 60)
? 0,0,0

please input the graphics step size (days)
? 1

orbital perturbations

would you like to include solar perturbations (y = yes, n = no)
? n

would you like to include lunar perturbations (y = yes, n = no)
? n

please input the algorithm error tolerance
(a value of 1.0e-8 is recommended)
? 1e-8
```

The following is a plot of delta-semimajor axis versus east longitude of the satellite. The delta-semimajor axis of this plot is the difference between the instantaneous semimajor axis and the synchronous semimajor axis a_s at the nominal east longitude.



The following is a plot of this example that includes the point-mass gravity perturbation of the sun and moon.



In both of these plots the "nominal" east longitude of the satellite is marked with an asterisk.

After the graphics are complete the software will display the following information which is based on the numerical integration of the satellite's motion.

```
                    program geosync4

< east-west stationkeeping of geosynchronous satellites >

integrated solution

initial calendar date       01-Jan-1998
initial universal time      00:00:00

final calendar date         11-Mar-1998
final universal time        14:59:40

final semimajor axis      42161.7620 kilometers

final eccentricity          0.000041

final east longitude       45.508808 degrees

first delta-v               0.163431 meters/second

second delta-v              0.163422 meters/second

total delta-v               0.326853 meters/second
```

This information can be used to verify the calculations described in the following section.

Technical Discussion

If g_1 is positive, the initial east longitude of the satellite is given by

$$\lambda_0 = \lambda_s + \frac{1}{2}\Delta\lambda$$

If g_1 is negative, the initial east longitude of the satellite is given by

$$\lambda_0 = \lambda_s - \frac{1}{2}\Delta\lambda$$

where $\Delta\lambda$ is the *total* deadband longitude constraint.

The drift cycle period is determined with the following expression:

$$P_D \le \frac{2}{\omega_e}\sqrt{\left|\frac{2\Delta\lambda}{\ddot{\bar{\lambda}}}\right|}$$

In this equation ω_e is the inertial rotation rate of the Earth and $\ddot{\bar{\lambda}}$ is the normalized longitudinal acceleration given by

$$\ddot{\bar{\lambda}} = 18\left\{c_{22}\sin 2\left(\lambda_s - \lambda_{22}\right) - 0.25c_{31}\sin\left(\lambda_s - \lambda_{31}\right) + 7.5c_{33}\sin 3\left(\lambda_s - \lambda_{33}\right)\right\}$$

In this equation $c_{nm} = J_{nm}\left(r_{eq}/a_s\right)^n$. The dimensional longitude acceleration is $\ddot{\bar{\lambda}}\omega_e^2$.

The spherical harmonic and longitude coefficients of the EGM96 gravity model are determined from the following expressions:

$$J_{nm} = -\sqrt{C_{nm}^2 + S_{nm}^2}$$

$$\lambda_{nm} = \frac{1}{m}\tan^{-1}\left(\frac{S_{nm}}{C_{nm}}\right)$$

where *n* is the degree (zonals) and *m* is the order (tesserals) of these terms.

The delta-v required for each two-impulse Hohmann orbit transfer is given by

$$\Delta V_D = \frac{1}{3}\omega_e V_{syn} P_D \ddot{\bar{\lambda}}$$

The total annual ΔV required for east-west stationkeeping is given by

$$\Delta V_D = \frac{1}{3}\omega_e V_{syn} P_D \ddot{\bar{\lambda}}\left(\frac{365.25}{P_D}\right)$$

The initial semimajor axis required for the drifting motion can be determined from

$$a_0 = a_s\left(1+\eta\right)$$

where $\eta = \pm\frac{4}{3}\sqrt{3g_1 sign\left(g_1\right)\Delta\lambda}$ and a_s is the "reference" synchronous radius at the satellite's nominal east longitude λ_s. The equations used to determine this radius are described in the technical discussion for the `geosync1.m` script. The positive sign in the expression for η should be used whenever g_1 is positive.

The initial drift rate of the satellite is determined from

$$\Delta\dot{\lambda}_0 = \mp 2\sqrt{3g_1 sign\left(g_1\right)\Delta\lambda_0}$$

where the negative sign of this equation is used whenever $g_1 > 0$ and the positive sign is used whenever $g_1 < 0$.

geosync5.m – north-south stationkeeping of geosynchronous satellites

This MATLAB script can be used to graphically display the long-term orbital inclination behavior of geosynchronous satellites. It can also be used to calculate the impulsive delta-v required to correct these inclination changes. The inclination perturbation of geosynchronous satellites is caused mainly by the gravitational attraction of the sun and moon.

This script numerically integrates the Cartesian form of the equations of orbital motion. These equations include the point-mass gravity of the sun and moon and the J_2 gravity effect of the Earth. The orbital conditions at the time of a north-south correction maneuver are computed using a one-dimensional root-finder based on Brent's method.

The delta-v required to correct the inclination change is applied at either the ascending or descending node. The scalar magnitude of this impulsive maneuver is given by

$$\Delta V = 2V \sin\left(\frac{\Delta i}{2}\right)$$

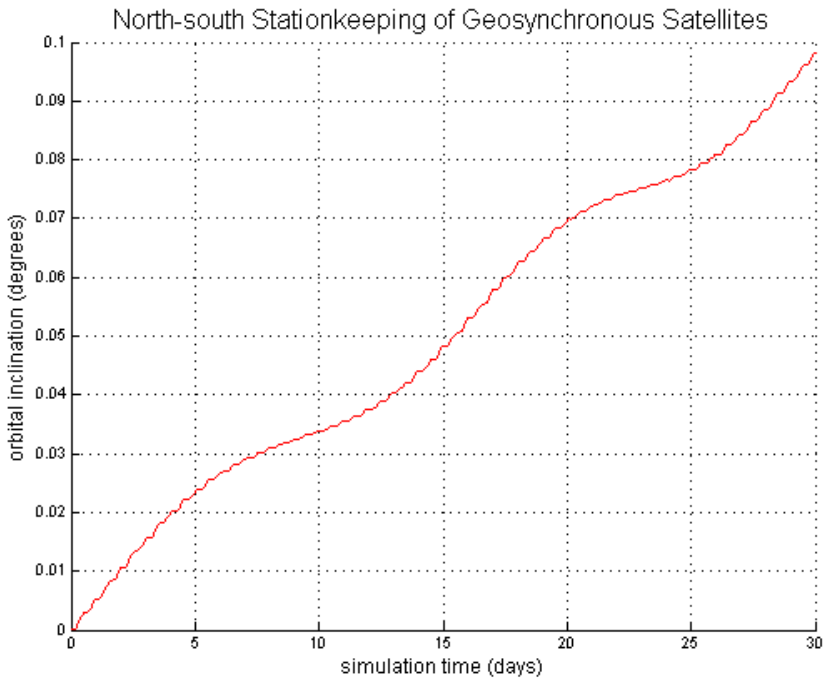
where V is the speed of the spacecraft prior to the maneuver and Δi is the inclination change.

The following is a typical user interaction with the graphics option of this script.

```
                    program geosync5

< north-south stationkeeping of geosynchronous satellites >


(1) create and display graphics

(2) predict time and delta-v

? 1


please input the integration error tolerance
(a value of 1.0e-8 is recommended)
? 1e-8


initial calendar date and time

please input the calendar date
(1 <= month <= 12, 1 <= day <= 31, year = all digits!)
? 1,1,2003

please input the universal time
(0 <= hours <= 24, 0 <= minutes <= 60, 0 <= seconds <= 60)
? 0,0,0


initial orbital elements
```

```
please input the semimajor axis (kilometers)
(semimajor axis > 0)
? 42164

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

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

please input the satellite's east longitude (degrees)
(0 <= east longitude <= 360)
? 45


please input the simulation period (days)
? 30

please input the graphics step size (minutes)
? 120
```

The following is the graphics display for this example.



The following is a typical user interaction with the "`predict time and delta-v`" option of this script.

```
please input the integration error tolerance
(a value of 1.0e-8 is recommended)
? 1e-8
```

```
please input the root-finding error tolerance
(a value between 1.0e-4 and 1.0e-6 is recommended)
? 1e-4


initial calendar date and time

please input the calendar date
(1 <= month <= 12, 1 <= day <= 31, year = all digits!)
? 1,1,2003

please input the universal time
(0 <= hours <= 24, 0 <= minutes <= 60, 0 <= seconds <= 60)
? 0,0,0


initial orbital elements


please input the semimajor axis (kilometers)
(semimajor axis > 0)
? 42164

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

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

please input the satellite's east longitude (degrees)
(0 <= east longitude <= 360)
? 45

please input the final orbital inclination (degrees)
(0 <= inclination <= +180)
? .05
```

The following is the script output for this example.

```
final calendar date       16-Jan-2003

final universal time        09:56:22

mission elapsed time          15.4141 (days)

delta-v                        2.6834 (mps)

      sma (km)        eccentricity     inclination (deg)    argper (deg)
 4.2165306913e+004  1.4971447098e-004  5.0000000000e-002  1.6714095246e+002

     raan (deg)     true anomaly (deg)   arglat (deg)       period (min)
 1.0309105956e+002  3.9520052673e+001  2.0666100513e+002  1.4361262783e+003
```