Celestial Body Position Tracking

By Nate Krinsky, EE '19

Introduction

Our project attempts to provide an alternative navigation system in the event of the failure of the existing GPS system to locate the user. This requires reading signals from non-GPS satellites called LEO (low earth orbit) satellites. In order to track and read signals from these LEO satellites, it is required to calculate their position in space. The following is a description of the different mathematical and computerized tools that we used to estimate the position of LEO satellites in our project.

Orbital Mechanics

Orbital Propagation Calculations

The path that a small celestial body takes around a large massive body such as a star or planet is called an orbit. The shape of the orbit can be described as a conic region, which is a round shape resulting from the intersection of a plane and a cone. Figure 1 below shows the conic regions of circle, ellipse, parabola, and hyperbola that result in planes intersection a pair of cones at different angles. A satellite's orbit is either circular or, more commonly, elliptical.

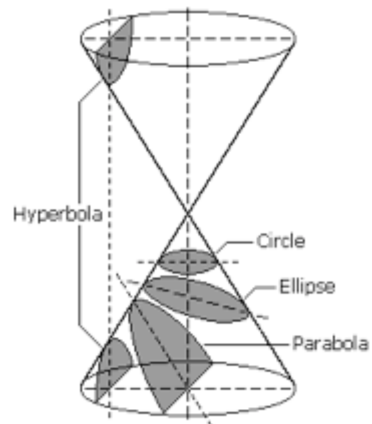

Figure 1. Different conic regions resulting in intersections between a plane and a pair of cones at different angles [1].

An object's orbit can be uniquely described by a set of six parameters known as orbital elements. Those six elements are the orbit's semi-major axis, its eccentricity, the inclination, the argument of periapsis, the time of periapsis passage, and the longitude of ascending node. Knowledge of these six elements, which can be concluded by as few as three measurements, can allow the calculation of future positions as a function of time [1].

Eccentricity (denoted by the variable e) is essentially a measure of how circular the orbit path is. It is calculated by dividing the distance between the foci of the elliptical orbit divided by the semi-major axis. If the orbit is perfectly circular, $e = 0$ since the foci overlap and thus the distance between them is 0.

The semi-major axis (denoted by the variable *a*) describes the shape of the orbit. The major axis is the longest line that can be drawn through the center of an ellipse, while the minor axis is the smallest line that can be drawn through the center. The semi-major axis is simply half the length of the major axis, or the distance from the furthest point from the center to the center. In the case when the orbit is perfectly circular, i.e. eccentricity is 0, the minor and major axes are equal and the semi-major axis is equal to the radius.

Inclination (denoted by the variable *i*) is the angular distance between the orbital plane and the plane of the primary's equator. An inclination of 0 degrees indicates that the satellite's orbital plane is on the same plane as the primary's equatorial plane and that the satellite rotates in the same direction as the primary. Contrastingly, an inclination of 180 degrees also indicates that the equatorial and orbital planes are coplanar but the satellite rotates in the opposite direction of the primary. An inclination of 90 degrees refers to an orbital plane that is perpendicular to the primary's equator. Figure 2 shows a cross-sectional view of a primary, a satellite, and the orbital and equatorial planes which highlights the angle of inclination.

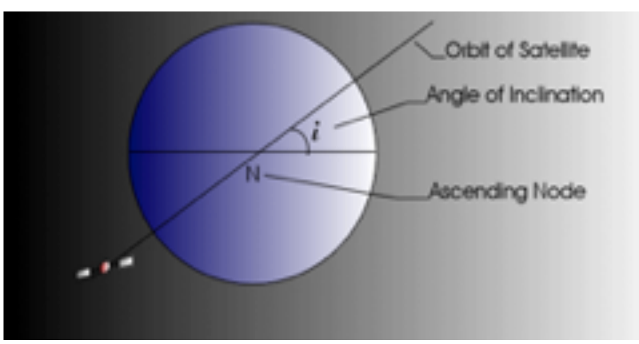

Figure 2. Visualization of ascending node and angle of inclination on a celestial body [6].

Figure 2 also shows the ascending node. Nodes are the points where the orbital plane intersect the primary's equatorial plane. The ascending plane is the node where the satellite passes from the southern hemisphere of the primary to the northern hemisphere [6]. The other node, the descending node, is the point where the satellite passes from the northern hemisphere to the southern hemisphere. The longitude of ascending node (Ω) is one of the elements used for propagation calculations.

The periapsis is the point on the orbital when the satellite comes closest to its primary. In our solar system, different planets have special names for their periapsides: Earth's is called the perigee, the sun's is called the perihelion, and Jupiter's is called the perijove, to name a few examples [1]. The two orbital elements involving the periapsis point are the argument of periapsis (ω) and the time of periapsis passage (*T*). As its name suggests, the time of periapsis passage is the time at which the satellite passes through the periapsis point. The argument of periapsis is the angular distance from the ascending node to the periapsis point [6].

Another relevant parameter, while not considered an orbital element, is true anomaly, which refers to the angular distance between a point in orbit past the point of periapsis. True anomaly is represented by the greek letter nu (ν). For a circular orbit, this is equal to mean anomaly M, the fraction of the orbital period that has elapsed since periapsis. In general, v and M are related by the approximation $v = M + 2e\ sin(v) + 1.25e^2 sin(2M)$ [1].

These six orbital elements, along with some basic knowledge of mechanics, allow for the prediction of a satellite's position at some arbitrary time in the future. While the derivation of these formulas can be found in Braeunig's primer on orbital mechanics [1], the relevant quantities of instantaneous spacecraft velocity and position vector can be described in the following equations:

Distance from primary: $r = \frac{a(1-e^2)}{1 + e\ cos(v)}$, where *a* is the semi-major axis, *e* is the eccentricity, and *v* is the true anomaly.

Position angle on orbital plane: $\phi = tan^{-1}\left(\frac{e \sin(v)}{1 + e \cos(v)}\right)$, where *e* is the eccentricity and *v* is the true anomaly.

Spacecraft velocity: $v = \sqrt{GM(\frac{2}{r} - \frac{1}{a})}$, where *G* refers to the universal gravitational constant 6.67259x10^{-11} N-m^2/kg^2, *M* refers to the mass of the earth, *r* refers to the magnitude of the position vector, and *a* is the semi-major axis.

Thus the orbital elements, and values calculated from them, allow for the orbital prediction of an object as a function of time.

Space Perturbations

Although the above calculations do a decent job estimating the time-varying position of an object in an orbit, the picture is incomplete. As Muira [5] analogizes in his master's thesis on perturbation models, propagation calculations are akin to estimating a future position of a car through knowledge of its velocity and position at a point in time. However, the car's path may have hills, the car may encounter pot holes, and different areas of the road may have different speed limits. The analogue in orbital mechanics are factors such as drag, radiation, spherical harmonics, and effects from massive bodies other than Earth. In order to account for these perturbations, more advanced models are required. Models that do not take these factors into account, such as an early model developed by the National Space Surveillance Center in 1959, fail to accurately predict the position of a satellite further than a week from the observed data [5].

While the main force acting on a satellite in Earth's orbit is Earth's gravitational force, the gravitational forces from the sun and moon cannot be neglected. Perturbations from these bodies are called third-body perturbations, and they cause linear changes in a satellite's ascending node longitude, argument of periapsis, and mean anomaly, though the mean anomaly is affected much less than the other two parameters. Approximations for the effects on ascending node longitude and argument of periapsis as caused by gravitational effects of the sun and moon are easily calculated and can be located in the appendix.

Standard propagation calculations also make the incorrect assumption that Earth is perfectly spherical and homogenous, which it of course is not. The perturbations caused by this fact are called zonal harmonics. The effect on ascending node longitude and argument of periapsis due to Earth's non-spherical nature are written in the appendix.

Satellites also experience atmospheric drag during their orbits which acts as a force in the direction opposite of their velocity. The effect of this drag is to continuously change satellite's semi-major axis, orbital period, and velocity, which can be approximated by describing uniform changes per revolution. The appendix contains the relevant equations and a description of the symbols introduced to perform those calculations.

General Perturbation Models

Owing to the necessity of being able to accurately track all objects orbiting the Earth, models that incorporate perturbation have been developed to accurately predict the motion of orbiting bodies. In 1960, a model taking into account the perturbations described above called the Simplified General Perturbation (SGP) model was released [3]. While successful, the number of satellites that needed to be cataloged quickly grew too large to be practically accounted for by the relatively complex model, necessitating a simplified version of the model called SGP4 being released in 1970 [3]. SGP4 itself failed to properly track bodies at an altitude above 255 kilometers. Satellites at this height or higher use the Simplified Deep Space Perturbation (SDP/SDP4) model. Although all four models are still supported, SGP4 is the most commonly used.

The SGP4 model begins with first initializing the values of the atmospheric drag effects, earth zonal harmonics, and lunar and solar gravitational effects based on the specifics of the satellite and its motion. The initialization of these parameters will not be discussed here but can be found in Appendix A of Hoots' A History of Analytical Orbit Modeling in the United States Space Surveillance System [3]. SGP4 first updates the orbital elements by taking into account the effects of zonal harmonics and atmospheric drag effects. Mean anomaly, argument of periapsis, and ascending node longitude, from which position and velocity can be determined, are updated according to the equations found in the appendix to this paper. From these updates, position estimation calculations are performed.

Two-line element (TLE) data

Two-line element data, or TLE data, is a standard data system used by satellites to transmit but of the orbital elements discussed in detail above. It is a standardized way to convey the data necessary to calculate satellite position, and it integrates both the space perturbation models and the PyEphem library. TLE data sets from many satellites are updated daily and can be found on open source websites like CelesTrak.com [4].

A set of TLE data contains, as the name would suggest, two lines of data, the first with nine data elements and the second with eight. All TLE sets follow the same format. Figure 3 shows the standard input of a TLE set with the different elements labeled.

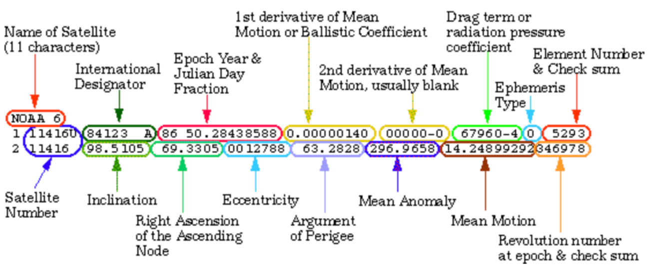

Figure 3. Standard form of Two-Line Element sets [2].

As can be seen in the figure, a TLE set explicitly contains some orbital elements like inclination, eccentricity, and argument of periapsis (labeled argument of perigee) as elements of the data set. It also contains information necessary for the SGP4, such as mean anomaly, ballistic coefficient, and drag coefficient. Other data, such as the date of the data set, obviously necessary to compute the time varying position function, as well as the satellite's name and number so anyone receiving the signal knows where it is coming from [2].

A receiver that has received a TLE set is able to use the SGP4 model to accurately predict future positions of the satellite that transmitted the set.

PyEphem

While it is technically possible to calculate a satellite's position by hand using the equations written in this paper, this is entirely impractical and never done in practice. A more realistic situation is to automate the process of downloading TLE data sets to use them as inputs to applications that perform SGP4 calculations. As such, intimate knowledge of the propagation and perturbation models is not strictly necessary, but helpful in verifying what is happening under the hood. One tool that uses the SGP4 model is PyEphem.

PyEphem is a simple and easy to use open source Python library. Given a time and specific point on Earth's surface, it can calculate the relative positions of planets, moons, and satellites seen from that position using the SGP4 model. Alternatively, given the identity of a planet or satellite, it can calculate the geographic location directly beneath that object at a particular time. Ephem is short for ephemeris, the traditional term for a table that tells the position of a celestial body by date and time [7].

PyEphem contains two relevant classes whose objects have orbital mechanic methods: Body and Observer. A Body object corresponds to some celestial body, be it a planet, moon, star, or satellite. Ephem contains specialty Body classes for commonly used bodies, for example the planet Mars: `m = ephem.Mars()`. Body objects for satellites can be initialized directly from TLE data, for example:

```
iss = ephem.readtle("ISS (ZARYA)", "1
25544U 98067A   03097.78853147
.00021906  00000-0  28403-3 0  8652",
"2 25544  51.6361  13.7980 0004256
35.6671  59.2566 15.58778559250029")
```

Body objects contain many attributes that describe the body and its motion, including distance from the Earth (or whatever its primary is), instantaneous velocity, height above sea level, and the latitude and longitude directly underneath. When a Body object is created, all these attributes are calculated at the time and date of the run time. To calculate these attributes at a different time, pass the time and date as the parameter to the `calculate()` method:

```
iss.compute('2003/3/23')
print('%s %s' % (iss.sublong,
iss.sublat))
>> -76:24:18.3 13:05:31.1
```

Observer objects represent a particular location on Earth's surface. They are initialized with a latitude, longitude, and altitude coordinate, as well as a particular date. For example:

```
lowell = ephem.Observer()
lowell.lon = '-111:32.1'
lowell.lat = '35:05.8'
lowell.elevation = 2198
lowell.date = '1986/3/13'
```

An Observer object can then be passed as the argument of a Body's `calculate()` function. Doing so will calculate the attributes of the body at the time set in the Observer's date attribute. These calculations also inform the location of the Body in relation to the Observer. For example the time that a particular Body appears to rise from the horizon in relation to the Observer, depending on the body's orbital path, with the `next_rising()` method.

With these functions, the PyEphem library provides a user-friendly way to perform astronomical calculations without expertise with orbital mechanics and perturbation models. The usability of this library provides opportunities for creative applications, for example calculating the relative velocities of satellites at different positions to notice the shift in their transmitted frequencies. The PyEphem has more functionality that is described in detail on its website [7].

Conclusion

Position tracking of celestial bodies is a complex process that requires significant mathematical calculations. Luckily, through decades of clever engineering and hard work, the process has been greatly simplified for engineers like us who wish to determine position with minimal effort. The SGP4 perturbation model coupled with the TLE sets that make use of this model provide easily accessible and regularly updated datasets to be used in calculations. Our project made use of this model through the PyEphem open source library, which used these calculations under the hood. Though minimal knowledge of these calculations is needed to use our product, having a basic level of understanding is useful know how to interpret our results.

Appendix

Third body perturbations from the sun and moon:

$$\Omega_{moon} = -\frac{0.00338\cos(i)}{n}$$
$$\Omega_{sun} = -\frac{0.00154\cos(i)}{n}$$
$$\omega_{moon} = \frac{0.0169\left(4 - sin^2(i)\right)}{n}$$
$$\omega_{sun} = -0.00077(4 - sin^2(i))/n$$

In the above equations, *i* is inclination and *n* is the number of orbit revolutions per day [1].

Earth zonal perturbations:

$$\Omega = -2.0647 \times 10^{14} a^{-\frac{7}{2}} \cos(i)(1 - e^2)^{-2}$$
$$\omega = 1.03237 \times 10^{14} a^{-\frac{7}{2}} \left(4 - 5sin^2(i)\right)(1 - e^2)^{-2}$$

In the above equations, a is the semi-major axis, i is the orbital inclination, and e is the orbital eccentricity [1].

Atmospheric drag perturbations:

$$\Delta a = -\frac{2\pi C_D A \rho a^2}{m}$$
$$\Delta P = -\frac{6\pi^2 C_D A \rho a^2}{mv}$$
$$\Delta v = \pi C_D A \rho a v/m$$

In the above equations, C_D is the drag coefficient, ρ is the air density, v is the velocity, and A is the area of the sun perpendicular to the satellite [1]. The expression $m/C_D A$, which appears in each equation, is referred to as the ballistic coefficient.

Perturbations in the SGP4 model:

$$\Omega = \Omega_0 + \frac{d\Omega}{dt}(t-t_0) - \frac{21}{2}\frac{5.41\times10^{-4} n_0\, a_E^2 \cos(i_0)}{a_0^2\,(1-e_0^2)} C_1 (t-t_0)^2$$

$$\omega = \omega_0 + \frac{d\omega}{dt}(t-t_0) - \delta\omega - \delta M$$

$$M = M_0 + n_0(t-t_0) + \frac{dM}{dt}(t-t_0) + \delta\omega + \delta M$$

where

$$\delta\omega = BC_3 \cos(\omega_0)(t-t_0)$$

$$\delta M = -\frac{2}{3}(q_0-s)^4 B \frac{1}{(a_0-s)^3}\frac{a_E}{e_0^2 a_0}[(1+\frac{a_0 e_0}{a_0-s}\cos[M_0 + n_0(t-t_0) + \frac{dM}{dt}(t-t_0)])^3 - (1+\frac{a_0 e_0}{a_0-s}\cos(M_0)^3]$$

A subscript of 0 refers to a quantity at the reference time. B is the atmospheric drag coefficient, q_0 is a constant equal to 120 km plus the radius of the Earth, and a_E refers to the radius of the Earth. C_1, C_2, and C_3 are variables used to simplify the expressions and are defined as:

$$C_2 = B(q_0-s)^4 \frac{1}{(a_0-s)^4} n_0 (1-\eta^2)^{7/2}[a_0(1 + \frac{3}{2}\eta^2 + 4e_0\eta + e_0\eta^3) + \frac{3}{2}\frac{5.41\times10^{-4}}{(a_0-s)(1-\eta^2)}(-\frac{1}{2} + \frac{3}{2}cs^2(i_0))(8+24\eta^2+3\eta^4)$$

Additionally, $\eta = \frac{a_0 e_0}{a_0 - s}$.

The perturbations based on lunar and solar gravitational forces are accounted for simply by incorporating the ascending node longitude and argument of periapsis effects described above:

$$\Omega = \boldsymbol{\Omega} + \frac{\boldsymbol{d}}{\boldsymbol{dt}}(-0.00338\cos(i)/n - 0.00154\cos(i)/n)(t-t_0)$$

$$\omega = \omega + \frac{d}{dt}(0.0169(4-\sin^2(i))/n - 0.00077(4-sn^2(i))/n)(t-t_0)$$

Additional perturbations and their effects on orbital elements are further explored in Hoots' paper but will not be repeated here.

References

1. Braeunig, R. (2013). Orbital Mechanics. Retrieved from http://www.braeunig.us/space/orbmech.htm

2. Dismukes, K. (2011, September 23). Definition of Two-line Element Set Coordinate System. Retrieved from https://spaceflight.nasa.gov/realdata/sightings/SSapplications/Post/JavaSSOP/SSOP_Help/tle_def.html

3. Hoots, F. R. (2004). A History of Analytical Orbit Modeling in the United States Space Surveillance System. Retrieved from http://aero.tamu.edu/sites/default/files/faculty/alfriend/S6.1 Hoots.pdf

4. Kelso, T.S. (2018, September 17). CelesTrak. Retrieved from http://celestrak.com/

5. Miura, N.Z. (2009). Comparison and design of simplified general perturbation models (SGP4) and code for NASA Johnson Space Center, Orbital Debris Program office (Master's thesis). Retrieved from https://digitalcommons.calpoly.edu/cgi/viewcontent.cgi?referer=https://en.wikipedia.org/&httpsredir=1&article=1094&context=theses

6. Nelson, E. (n.d.). Keplerian Elements. Retrieved from https://marine.rutgers.edu/cool/education/class/paul/orbits.html

7. Rhodes, B. C. (2008). PyEphem Quick Reference. Retrieved from https://rhodesmill.org/pyephem/quick.html