Escola d'Enginyeria de Telecomunicació i Aeroespacial de Castelldefels
UNIVERSITAT POLITÈCNICA DE CATALUNYA BARCELONATECH

Navegació Aèria, Cartografia i Cosmografia (NACC)

PRÀCTICA 1

GROUND TRACK OF GPS SATELLITES

Joan Olmos

EETAC 2021-22

V2.0

Contents

1 Objective

The objective of this lab. work is to be able to plot the satellite ground track for any given satellite or constellation of satellites. Initially we will work with the GPS constellation because the real-time ephemeris are available in the Internet in a clear text format, but the programs can be applied to any satellite for which the Keplerian elements are known. **The results to be delivered are marked in red characters.**

2 Understanding GPS ephemeris

The GPS almanac is the set of satellite ephemeris that every GPS satellite transmits. It includes information about the status (health) of the entire GPS satellite constellation and the Keplerian elements of each satellite orbit. These files can be obtained from the source (1) or from http://celestrak.com/GPS/almanac/Yuma/2020/

The almanac contains the following data and keplerians about each satellite (2):

1) **ID**: PRN (Pseudo Random Noise) id. of the SVN (Space Vehicle NAVSTAR)
2) **Health:** Indicates the operational state of the SV (Space Vehicle). The value "000" means that the SV is usable.
3) **Eccentricity:** This shows the amount of the orbit deviation from a circular orbit. It is the distance between the foci divided by the length of the semi-major axis (the NAVSTAR orbits are almost circular).
4) **Time of Applicability (ToA):** The time instant (seconds within the GPS week) for which the almanac has been computed (ephemeris epoch).
5) **Orbital Inclination:** The angle of the SV orbit plane with respect to the equator (GPS is at approx. 55 degrees) in radians. The SV ground track will not rise above approx. 55 degrees of latitude.
6) **Rate of Right Ascension:** Rate of change of the angle of right ascension in rad/s.
7) **SQRT(A) Square Root of Semi-Major Axis:** The semi-major axis is the distance from the center of the ellipse to either the point of apogee or the point of perigee.
8) **Right Ascension at GPS week epoch:** Geographic longitude (in rads. with respect to Greenwich meridian) of the ascending node of the orbit at the GPS week epoch (Saturday to Sunday midnight).
9) **Argument of Perigee at ToA:** An angular measurement (rads.) along the orbital path measured from the ascending node to the point of perigee. It is measured in the direction of the SV's motion.
10) **Mean Anomaly at ToA:** Angle (rads.) traveled past the perigee at ToA. When the SV has passed perigee and heading towards apogee, the mean anomaly is positive. After the point of apogee, the mean anomaly value will be negative to the point of perigee.
11) **Af(0):** SV clock bias in seconds
12) **Af(1):** SV clock Drift in seconds per seconds

13) **Week:** GPS week number (0000-1023). Counter increased (module 1024) every week since 0h UTC of Jan 6th, 1980 (the GPS epoch = 2444244.5 JD). First roll-out happened at 23:59:47 (UTC) on Aug 21, 1999 (roll epoch = 2451412.499850 JD).

The provided function "almanac.m" will download from http://celestrak.com the current GPS almanac and store it in an ASCII file ready for use in the next steps of the lab. work. The return of subroutine "almanac.m" is a matrix called *Eph(i, j)* with 31 rows and 13 columns, where row *i* holds the ephemeris for satellite *i* stored in the same order as they appear in the web page of Celestrak.

3 Understanding the "Two Line Elements" (TLE) format

The TLE is the standard way of expressing the keplerian elements (orbit characterization parameters). In the Celestrak web page you may find the TLE for most civilian satellites. They are just two lines of numbers (ascii code) that hold all the information to compute the satellite position at a given moment. The keplerian elements are given in a slightly different format than the already seen Celestrak GPS almanac. The detailed description of the TLE format can be found at: http://celestrak.com/columns/v04n03/. We summarize it here:

From the first line we are only interested in the fourth field (we consider that "fields" are separated by white space). For example:

1 25544U 98067A **22061.21033787** .00008312 00000+0 15594-3 0 9992

The fourth field is 22061.21033787, where the first 2 digits (22) are the last digits of the current year (2022) and the remaining digits (061.21033787) give the ToA of the keplerian elements expressed in days from 00:00 (UTC) of January 1, 2022. The matlab function "time2toa.m" (available in Atenea) accepts the ToA in that format and returns (inside the "esec" variable) the number of seconds elapsed since (or remaining to) the ToA. If "esec < 0" then the supplied ToA is in the future.

The second line includes the most relevant parameters separated by white space. For example:

2 25544 **51.6434 146.3647 0005536 203.7607 179.2077 15.49533599328593**

We explain the fields in the order that they appear (starting at field 3):

3) 51.6434 **Orbit inclination** [degrees]
4) 146.3647 **Right ascension of the ascending node at ToA** [degrees]
5) 0005536 **Orbit eccentricity** times 10^7 (e = 0.0005536)
6) 203.7607 **Argument of the perigee at ToA** [degrees]
7) 179.2077 **Mean Anomaly at ToA** [degrees]
8) 15.49533599328593 **Mean motion** (satellite angular speed of rotation) in [Revolutions/day]

Notice that the "mean motion" (n) relates to the orbit period (T) according to the expression: $n = 2\pi / T$ [rad/s].

4 Finding the ECEF coordinates of GPS satellites

After completing this task you will be able to compute the ECEF coordinates of any satellite at a given time. Also, you will learn how to plot the satellite ground track for a given period of time.

In this section we will work with NAVSTAR (GPS) satellite orbits. The orbits are very well known and characterized. You can get the satellites ephemeris from the U.S Navigation Center (1) or from http://celestrak.com/GPS/almanac/Yuma/2020/. The MATLAB subroutine to download the almanac for the current time is provided.

4.1 ECEF coordinates of a satellite from its Keplerian elements

The orbits are characterized by the six Keplerian parameters. Using these Keplerian parameters **you must write a MATLAB function to compute the ECEF coordinates of any GPS satellite at current time.** In this work the orbit corrections to account for slow changes in the Keplerian elements will not been considered.

It is advisable to write the matlab subroutine in a way that it can be used regardless if the keplerian elements are obtained in the almanac format or in the TLE format. For that purpose the subroutine should accept the following parameters:

Kepler2ECEF(a, i_0, e, Ω_0, $\dot{\Omega}_0$, ω, M_0, *n, dt*), where:

- a Orbit major semi axis [m]
- i_0 Orbit inclination [rad]
- e Orbit eccentricity
- Ω_0 Longitude of the ascending node (AN) at the ToA [rad]
- $\dot{\Omega}_0$ Rate of change of the right ascension of the AN [rad/s]
- ω Argument of the perigee at ToA [rad]
- M_0 Mean anomaly at ToA [rad]
- n Satellite mean motion [rad/s]
- dt Elapsed time from the ToA [s] (negative if the ToA is in the future)

Before calling the "Kepler2ECEF.m" subroutine, apply the following steps in order to prepare the arguments that the subroutine will need:

1) If using the almanac format: express the current time *(t)* in seconds from the start of the current GPS week (there is no need to program anything since *t=esec* is an output of the "*almanac.m*" subroutine). Then get the time difference between the ToA (t_0, also in seconds within the current GPS week) and the current time:
$$dt = t - t_0$$
The time difference can be negative.
If using the TLE format: just call the function *"time2toa.m"*, which accepts a single argument (the ToA in days from the start of the current year) and returns *dt*, which is directly the elapsed time (in seconds) from the ToA (negative if the ToA is in the future).

2) If using the almanac format: then get the semi major axis:

$$a = \left(\sqrt{a}\right)^2$$

and then compute the mean motion (n = satellite mean angular speed) using:

$$n = \sqrt{\frac{G \cdot M}{a^3}}$$

where G = 6.67384e-11 [m^3 kg^{-1} s^{-2}] is the Gravitational constant and M = 5.972e+24 [kg] is the Earth mass.

If using the TLE format: just compute the orbit period (T) form the mean motion ($n=2\pi/T$) using the right units, and then isolate a from the previous equation.

3) If using the almanac format: convert from the longitude of the ascending node at the GPS week epoch (Ω'_0) to the longitude of the ascending node at the ToA. This is as simple as computing:

$$\Omega_0 = \Omega'_0 - \dot{\Omega}_e \cdot t_0$$

where $\dot{\Omega}_e = 7.2921151467 \cdot 10^{-5}$ [rad/s] is the angular speed of Earth rotation.

If using the TLE format: convert from the **right ascension** of the ascending node at the ToA ($\Omega"_0$) to the **longitude** of the ascending node at the ToA (Ω_0) using:

$$\Omega_0 = \Omega"_0 - \text{GAST}$$

where GAST is the right ascension of the Greenwich meridian at the ToA. The GAST parameter (in degrees) is also returned when calling the provided function "*time2toa.m*".

4) Notice that, If using the TLE format, the rate of change of the right ascension ($\dot{\Omega}_0$) is zero.

Next, within the "*Kepler2ECEF.m*" subroutine, apply the following steps to get the satellite ECEF coordinates at the current time (see Figure 1):

1) Compute the current mean anomaly:

$$M_k = M_0 + n \cdot dt$$

2) Find the current eccentric anomaly (solve iteratively for E_k):

$$M_k = E_k - e \cdot sin(E_k)$$

(Take $E_k(0) = M_k$ and then compute repeatedly $E_k(n) = M_k + e \cdot \sin\left(E_k(n-1)\right)$ until $\left|E_k(n) - E_k(n-1)\right| < 10^{-8}$)

3) Find the true anomaly (from the sine and cosine expressions the true anomaly can be extracted without any ambiguity):

$$sin(\upsilon_k) = \frac{\sqrt{1-e^2} \cdot sin(E_k)}{1 - e \cdot cos(E_k)} \qquad cos(\upsilon_k) = \frac{cos(E_k) - e}{1 - e \cdot cos(E_k)}$$

To extract the true anomaly without ambiguity, you can use the complex number functions. In other words, convert the sinus and cosine of the true anomaly into a complex number and ask MATLAB to compute the argument. The useful matlab functions are angle() or atan2().

4) Find the argument of latitude:

$$u_k = v_k + \omega$$

5) Find the orbit radius (current distance to Earth center):

$$r_k = a \cdot (1 - e \cdot cos(E_k))$$

6) Compute the current longitude of the ascending node using:

$$\Omega_k = \Omega_0 + \dot{\Omega}_0 \cdot dt \ - \dot{\Omega}_e \cdot dt$$

7) Get *x* coordinate within the orbital plane:

$$x_p = r_k \cdot cos(u_k)$$

8) Get *y* coordinate within the orbital plane:

$$y_p = r_k \cdot sin(u_k)$$

9) ECEF x-coordinate:

$$x = x_p \cdot cos(\Omega_k) - y_p \cdot cos(i_0) \cdot sin(\Omega_k)$$

10) ECEF y-coordinate:

$$y = x_p \cdot sin(\Omega_k) + y_p \cdot cos(i_0) \cdot cos(\Omega_k)$$

11) ECEF z-coordinate:

$$z = y_p \, sin(i_0)$$

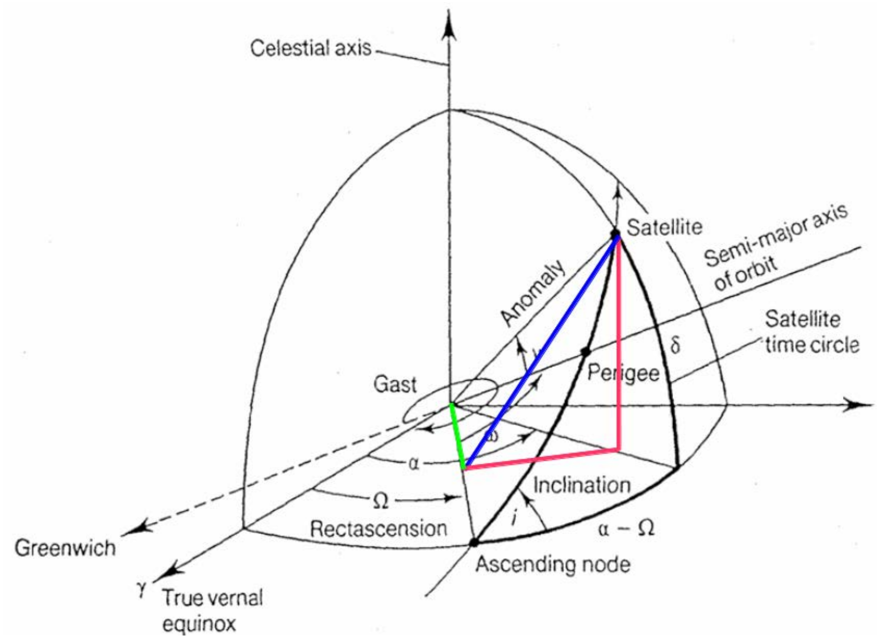

Figure 1. From Keplerian elements to ECEF coordinates

4.2 Plotting the satellite ground track

Once you have the current satellite ECEF Cartesian coordinates, you should **convert from the ECEF Cartesian coordinates (x, y, z) into LLA (Latitude, longitude and altitude) coordinates**. The algorithm for the ECEF to LLA coordinate conversion is:

$r = \sqrt{x^2 + y^2}$
$E^2 = a^2 - b^2$
$F = 54b^2z^2$
$G = r^2 + (1-e^2)z^2 - e^2E^2$
$C = \frac{e^4Fr^2}{G^3}$
$s = \sqrt[3]{1 + C + \sqrt{C^2 + 2C}}$
$P = \frac{F}{3\left(s + \frac{1}{s} + 1\right)^2 G^2}$
$Q = \sqrt{1 + 2e^4P}$
$r_0 = -\frac{Pe^2r}{1+Q} + \sqrt{\frac{1}{2}a^2\left(1 + \frac{1}{Q}\right) - \frac{P(1-e^2)z^2}{Q(1+Q)} - \frac{1}{2}\mathrm{Pr}^2}$
$U = \sqrt{z^2 + (r - r_0e^2)^2}$
$V = \sqrt{z^2(1-e^2) + (r - r_0e^2)^2}$
$z_0 = \frac{b^2z}{aV}$

$h = U\left(1 - \frac{b^2}{aV}\right)$
$\phi = arctg\left(\frac{z + e'^2 z_0}{r}\right)$
$\lambda = arctg\left(\frac{y}{x}\right)$

With *e* and *a* given by:

Datum	a	e^2
WGS-84	6378137 m	0.00669437999014

Remember that $a^2 = b^2 + c^2$, $e = c/a$ and $e' = c/b$. To compute *"arctg()"* it is advised to use the Matlab functions *"atan2"* or *"angle"*. See Matlab help on how to use these functions.

Even though Matlab has a specific script to convert from ECEF to LLA, **you should program it by yourself** and include your code when delivering your results. If you wish you can check if Matlab method gives the same results as your own script.
After that you must **select one of the satellites and plot the latitude versus the longitude in degrees** (the altitude is not relevant here). If you also plot the world map in the background (using the included file "world_110m.txt") you will obtain the satellite ground track. You should get a figure similar to Figure 2. It is advisable to plot the current position of the satellite with a bigger marker size and to add a text showing the satellite ID (as extracted from the almanac data), see Figure 2. The text can be added in Matlab with this statement:

text(long + 2, lat + 1, sprintf('%d', Eph(si, 1)));

where *lat, long* are the coordinates of the subsatellite point at current time and *Eph(si,1)* is the ID of satellite with index *si*.

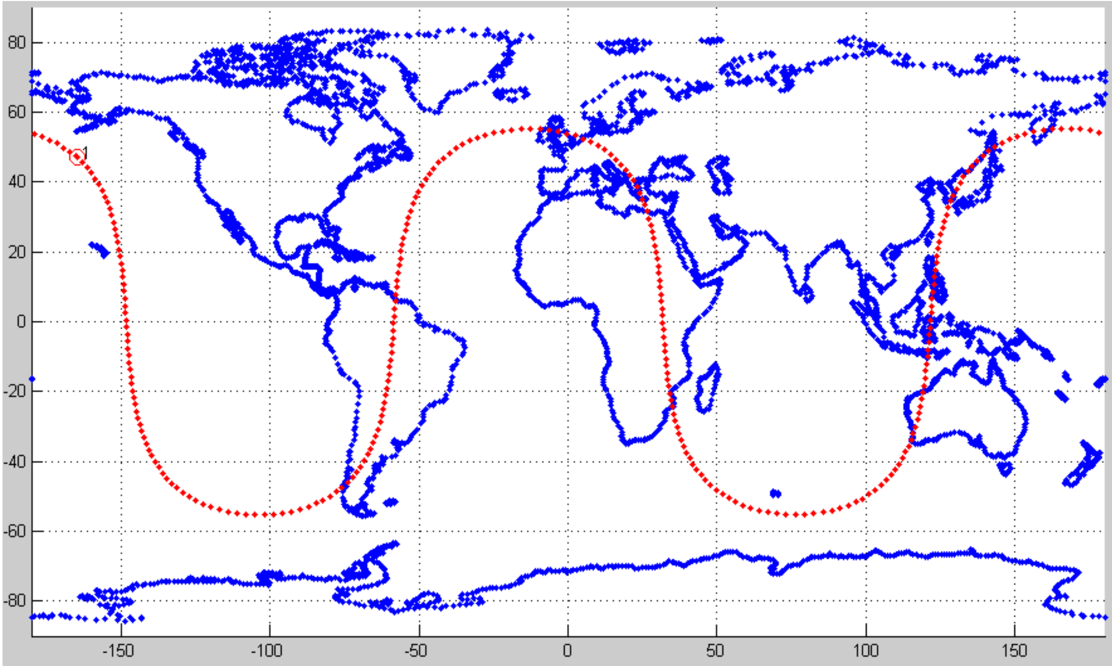

Figure 2. GPS satellite ground track

In order to get the ground track you must add an external loop to your Matlab program that increases the time from "*esec*" to "*esec+24h*" in steps of 5 minutes and re-compute the satellite ECEF coordinates at each step. Remember that "*esec*" is a Matlab variable, returned by the "*almanac.m*" function, which contains the current time in the GPS format (elapsed seconds from the beginning of the week). The period of the satellite orbit is ½ of the sidereal day (close to 12h), but since it turns the Earth in the prograde direction it takes 24h (two consecutive orbits) for the satellite to travel a complete ground track.

4.3 Map of all satellite ground tracks

In this task we will plot a map including the current subsatellite point of all the GPS satellites and their ground track for the next hour. To achieve this you must restrict the time loop to cover only the next 60 minutes and also add a new external loop that repeats the computation for the available 31 satellites. You should get a figure similar to Figure 3.

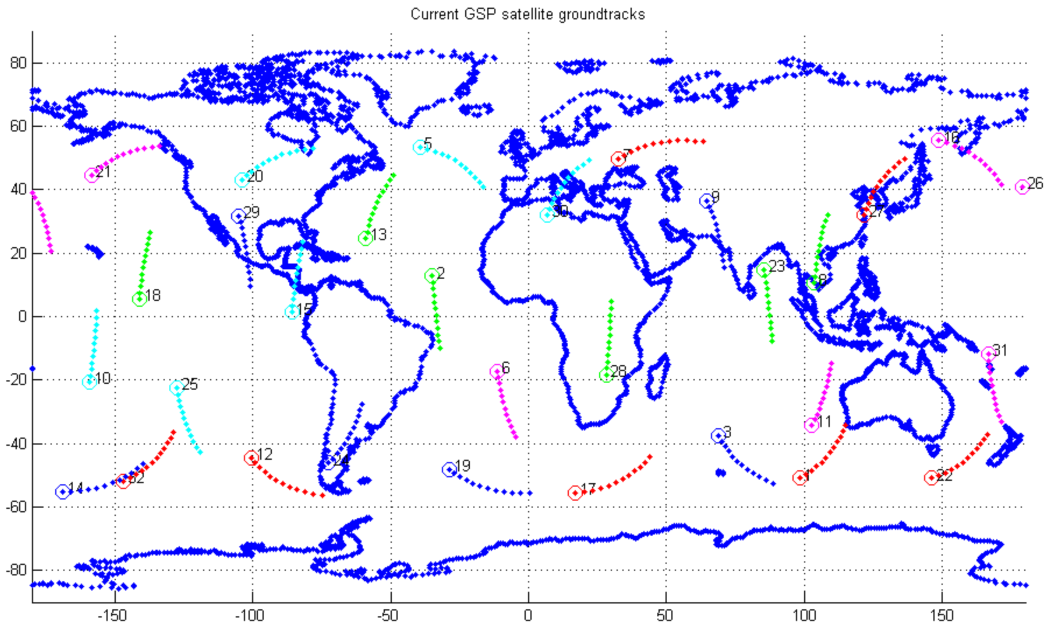

Figure 3. GPS satellite ground track

You can check that your results are correct by comparing with the online map (updated in real time) at http://www.nstb.tc.faa.gov/rt_waassatellitestatus.htm. Notice that this tool plots the ground track of satellites in the past hour while we are computing the ground track for the next hour. Alternatively, you can use the GNNS tools provided at http://gnssmissionplanning.com.

5 Plotting the ground track of a LEO satellite

In this task you will use your matlab code to plot the ground track of a LEO satellite using the TLE obtained from Celestrak. Each laboratory group of 2 students must use a different set of TLE according to the assigned group number. The list of TLE for all groups is:

1) CSG-2

```
1 51444U 22008A   22061.49149328 -.00082380  00000+0 -10283-1 0  9998
2 51444  97.8731 246.7398 0001460  21.5948 338.6308 14.82898484  4375
```

2) STARLINK-3167

```
1 51460U 22010E   22061.25001157 -.00068353  00000+0 -48542-3 0  9995
2 51460  53.2170  41.0390 0001133  45.6020  82.9465 15.73255308  5433
```

3) STARLINK-3181

```
1 51461U 22010F   22061.25001157 -.00060921  00000+0 -43183-3 0  9995
2 51461  53.2176  41.2166 0001396  47.1555 355.4204 15.73251698  5486
```

4) STARLINK-3166

```
1 51462U 22010G   22061.25001157  .00032535  00000+0  23530-3 0  9998
2 51462  53.2169  41.2135 0001359  60.3674 342.8609 15.73202981  5450
```

5) STARLINK-3419

```
1 51463U 22010H   22061.58334491  .00006540  00000+0  50969-4 0  9999
```

2 51463 53.2165 39.3817 0001627 45.1622 174.2006 15.73205569 5480

6) STARLINK-3189
1 51464U 22010J   22061.25001157 -.00068175 00000+0 -48410-3 0 9995
2 51464 53.2161 41.2084 0001354 47.1202 357.3447 15.73256177 5437

7) STARLINK-3401
1 51465U 22010K   22060.91668981 .00044704 00000+0 32108-3 0 9999
2 51465 53.2160 42.8537 0001385 38.7374 277.5140 15.73211584 3510

8) STARLINK-3178
1 51467U 22010M   22061.25001157 -.00723008 00000+0 -56487-2 0 9993
2 51467 53.2171 41.0392 0001645 44.4428 86.3602 15.73018611 5491

9) STARLINK-3174
1 51468U 22010N   22060.91668981 -.00027208 00000+0 -18974-3 0 9999
2 51468 53.2158 42.6808 0001395 54.1734 346.1288 15.73239663 3516

10) STARLINK-3182
1 51469U 22010P   22061.25001157 .00034112 00000+0 24667-3 0 9990
2 51469 53.2163 41.2110 0001631 57.3385 346.5378 15.73181574 5483

11) STARLINK-3165
1 51471U 22010R   22061.24114142 -.00026632 00000+0 -18523-3 0 9995
2 51471 53.2133 39.7483 0001275 38.0125 322.0985 15.73288172 5508

12) STARLINK-3415
1 51472U 22010S   22061.25001157 -.00022131 00000+0 -15355-3 0 9998
2 51472 53.2159 41.0342 0001485 39.8746 89.9529 15.73218929 5492

13) LIGHT-1
1 51509U 98067TG 22060.79773342 .00019256 00000+0 32540-3 0 9991
2 51509 51.6488 148.1546 0008393 190.6175 169.4635 15.51591225 4144

14) GT-1
1 51510U 98067TH 22061.49317944 .00054926 00000+0 85687-3 0 9996
2 51510 51.6442 144.5501 0005277 186.9968 173.0961 15.53278708 4270

15) COSMOS 2553
1 51511U 22011A   22061.33779755 -.00000070 00000+0 -96440-3 0 9990
2 51511 67.0922 191.3122 0004148 260.5410 99.5093 11.32853788 2840

16) ONEWEB-0410
1 51622U 22012A   22061.58334491 -.00791921 00000+0 -35458-1 0 9997
2 51622 87.3575 354.7363 0017487 169.6024 240.5127 15.23709341 4313

17) ONEWEB-0411
1 51623U 22012B   22061.58334491 -.00933975 00000+0 -52419-1 0 9991
2 51623 87.4067 354.7928 0015851 174.1229 323.4812 15.16007997 4298

18) ONEWEB-0415
1 51624U 22012C   22061.58334491 -.00958981 00000+0 -52919-1 0 9991
2 51624 87.4065 354.7900 0017043 174.0716 350.4278 15.16648065 4295

You must plot the ground track for a complete orbit period and use the Celestrak orbit visualization tool (in 2D) to check if the obtained plot matches the ground track shown in Celestrak. Remember to capture the screen showing the plot comparison.

6 Final remarks

To summarize you must:

1) Write a Matlab function to compute the ECEF coordinates of any GPS satellite from its Keplerian elements.
2) Write a Matlab function to convert from ECEF Cartesian coordinates into LLA coordinates. DO NOT USE THE FUNCTION INCLUDED IN MATLAB BUT FOR TESTING.
3) Write a Matlab main program to:
    - Plot a complete ground track of a single GPS satellite.
    - Plot the ground track of the full GPS constellation for the next hour.
4) Tune your Matlab program to plot the ground track during one orbit period of the LEO satellite corresponding to your lab. group number.
5) Write a short report including the images of the ground tracks that you have obtained, including a screen capture of the online tool at http://www.nstb.tc.faa.gov/rt_waassatellitestatus.htm to show that your results compare well.
6) Print the listings of all the code you have written on paper (leaving enough white space for comments) and comment the code explaining the purpose of each statement. The annotations must be hand-written.
7) Scan the annotated listings and join it to the delivered stuff.

The results of the laboratory work will be delivered through a repository in Atenea. Just make a zip file including the annotated listings, the short report comparing your screen captures with the online tool and include the code of all the Matlab programs that you have written (as ".m" files) in order that they can be run on any computer to test its validity.

7 References

1. **U.S. Department of Homeland Security.** *GPS NANUS, ALMANACS, & OPS ADVISORIES.* [Online] http://www.navcen.uscg.gov/?pageName=gpsAlmanacs.

2. —. *Definition of YUMA almanac.* [Online] http://www.navcen.uscg.gov/?pageName=gpsYuma.

3. **Subirana, J. Sanz, Zornoza, J.M. Juan and Hernández-Pajares, M.** *Transformations between ECEF and ENU coordinates.* [Online] 2011. http://www.navipedia.net/index.php/Transformations_between_ECEF_and_ENU_coordinates.