GPS Geodesy - LAB 5

From GPS ephemerides to ECEF satellite positions

GPS satellite ephemerides are broadcast to users as part of the GPS signal in the navigation message. They are calculated by predicting orbits forward on the basis of a curve-fit to 4 to 6 hours of actual orbit data. GPS broadcast ephemerides are accurate to ~10 m. They include the following parameters:

- t_{oe}: epoch of reference of the ephemerides in seconds of the current GPS week. It is the center of the interval over which the ephemeris is valid.
- μ_0, Δn, e, $sqrt(a)$, Ω_0, i_0, ω_0: orbital elements (= Keplerian parameters) at t_{oe}.
- Ωdot, $idot$: rate of right ascension and inclination angle.
- $C_{\omega c}$ $C_{\omega s}$ C_{rc} C_{rs} C_{ic} C_{is}: corrections coefficients to the argument of perigee, orbit radius and inclination for the perturbations caused by variations in the Earth's gravity field, solar radiation pressure, and attraction from Sun and Moon.

Assignment:

1. Write a program to convert the ephemerides given in a RINEX navigation file into an ECEF coordinate system. Input file = `epgga2.010`:
   - Write a function to read the broadcast ephemeris file. This function should return a matrix with 17 rows (the parameters needed later on in the calculations) and as many columns as there are ephemerides. You will need the 17 following parameters:

`svprn`	satellite PRN number
`Mo`	mean anomaly
`roota`	sqrt(semi-major axis)
`deltan`	variation of mean angular velocity
`ecc`	eccentricity
`omega0`	argument of perigee
`cuc, cus, crc, crs, cic, cis`	correction coefficients
`i0`	inclination
`idot`	rate of inclination
`Omega0`	right ascension
`Omegadot`	rate of right ascension
`toe`	time of ephemeris

The call to that function could look like: `eph = read_rinexn('epgga2.010');`

   - Write a function to convert the ephemerides matrix (output of previous function) for a given satellite at a given time (i.e. 3 input arguments). This function should return the corresponding X,Y,Z coordinates in ECEF frame, for instance:

```
[X,Y,Z] = get_satpos(t,sv,eph);
```

A possible algorithm is given below, together with the formulas that give the values for *Ω, ω, i, E, r,* and *v* needed in the equations above:

1. Extract needed parameters from the GPS broadcast ephemerides file. Note that the RINEX navigation files provide information every 2 hours only. Therefore, you first need to find the correct time of ephemeris (t_{oe}) in the navigation file given the time for which you need to extrapolate the satellite position (t_{data}).

2. Compute basic parameters at request time t_{data}:

Time elapsed since t_{oe}: $t = t_{data} - t_{oe}$

Mean anomaly at t: $\mu = \mu_0 + (\sqrt{GM / a^3} + \Delta n) \times t$

$GM = 3.986004418 \times 10^{14} m^3 / s^2$

Iterative solution for E: $E = \mu + e \sin E$ (use iterative solution)

True anomaly v: $v = \arctan\left(\frac{\sqrt{1 - e^2} \times \sin E}{\cos E - e} \right)$

3. Correct for orbital perturbations:

Argument of perigee: $\omega = \omega_0 + C_{wc} \cos 2(\omega_0 + v) + C_{ws} \sin 2(\omega_0 + v)$

Radial distance: $r = a(1 - e \cos E) + C_{rc} \cos 2(\omega_0 + v) + C_{rs} \sin 2(\omega_0 + v)$

Inclination: $i = i_0 + \dot{i} \times t + C_{ic} \cos 2(\omega_0 + v) + C_{is} \sin 2(\omega_0 + v)$

4. Compute the right ascension, <u>accounting for Earth's rotation</u> (ω_e = mean angular velocity of the Earth) and variations of Ω (= *Ωdot*):

$$\Omega = \Omega_0 + (\dot{\Omega} - \omega_e) \times t - \omega_e \times t_{oe}$$

$$\omega_e = 7.2921151467 \times 10^{-5} rad / s$$

5. Convert satellite position from orbital frame to ECEF frame:

Write the satellite position in the orbital frame: $\vec{r} = \begin{bmatrix} r \cos v \\ r \sin v \\ 0 \end{bmatrix}$

Build the rotation matrix to ECEF frame:

$$R = \begin{bmatrix} \cos\Omega \cos\omega - \sin\Omega \sin\omega \cos i & -\cos\Omega \sin\omega - \sin\Omega \cos\omega \cos i & \sin\Omega \sin i \\ \sin\Omega \cos\omega + \cos\Omega \sin\omega \cos i & -\sin\Omega \sin\omega + \cos\Omega \cos\omega \cos i & -\cos\Omega \sin i \\ \sin\omega \sin i & \cos\omega \sin i & \cos i \end{bmatrix}$$

Apply rotation:

$$\vec{\rho}_e = R\vec{r}$$

6. Check your program. Use values from the sp3 file `igs11484.sp3` as "ground truth":

`t=346500 and sv = 31 ⇒ X=11660.379642, Y=11313.211213, Z=-21326.822815`

2. Compute ECEF coordinates for satellite 31 every 15 minutes and compare with the coordinates given in the corresponding sp3 file (precise IGS orbits).
   - Write a function that extract the XYZ position of a satellite 'sv' from an sp3 file and returns that position:

     ```
     [Xs,Ys,Zs,Ts] = read_sp3(sp3file,sv);
     ```

   - Compute the 3-D residual position = vector difference between sp3 and broadcast XYZ position. Plot the norm of this vector as a function of time for satellite 31.

3. Comment on the differences between the 2 orbits.

Short description of the RINEX navigation file format. The format of fully described in http://www.ngs.noaa.gov/CORS/Rinex2.html.

TABLE A4

GPS NAVIGATION MESSAGE FILE - DATA RECORD DESCRIPTION

OBS. RECORD	DESCRIPTION	FORMAT
PRN / EPOCH / SV CLK	- Satellite PRN number	I2,
	- Epoch: Toc - Time of Clock	
	year (2 digits, padded with 0 if necessary)	1X,I2.2,
	month	1X,I2,
	day	1X,I2,
	hour	1X,I2,
	minute	1X,I2,
	second	F5.1,
	- SV clock bias (seconds)	3D19.12
	- SV clock drift (sec/sec)	
	- SV clock drift rate (sec/sec2)	
BROADCAST ORBIT - 1	- IODE Issue of Data, Ephemeris	3X,4D19.12
	- Crs (meters)	
	- Delta n (radians/sec)	
	- M0 (radians)	
BROADCAST ORBIT - 2	- Cuc (radians)	3X,4D19.12
	- e Eccentricity	
	- Cus (radians)	
	- sqrt(A) (sqrt(m))	
BROADCAST ORBIT - 3	- Toe Time of Ephemeris	3X,4D19.12
	(sec of GPS week)	
	- Cic (radians)	
	- OMEGA (radians)	
	- CIS (radians)	
BROADCAST ORBIT - 4	- i0 (radians)	3X,4D19.12
	- Crc (meters)	
	- omega (radians)	
	- OMEGA DOT (radians/sec)	
BROADCAST ORBIT - 5	- IDOT (radians/sec)	3X,4D19.12
	- Codes on L2 channel	
	- GPS Week # (to go with TOE)	
	Continuous number, not mod(1024)!	
	- L2 P data flag	
BROADCAST ORBIT - 6	- SV accuracy (meters)	3X,4D19.12
	- SV health (bits 17-22 w 3 sf 1)	
	- TGD (seconds)	
	- IODC Issue of Data, Clock	
BROADCAST ORBIT - 7	- Transmission time of message *)	3X,4D19.12
	(sec of GPS week, derived e.g.	
	from Z-count in Hand Over Word (HOW)	
	- Fit interval (hours)	
	(see ICD-GPS-200, 20.3.4.4)	
	Zero if not known	
	- spare	
	- spare	