Yaseen Hull

HLLYAS001

apg2014

Assignment 2b

Assignment 2: tut2b

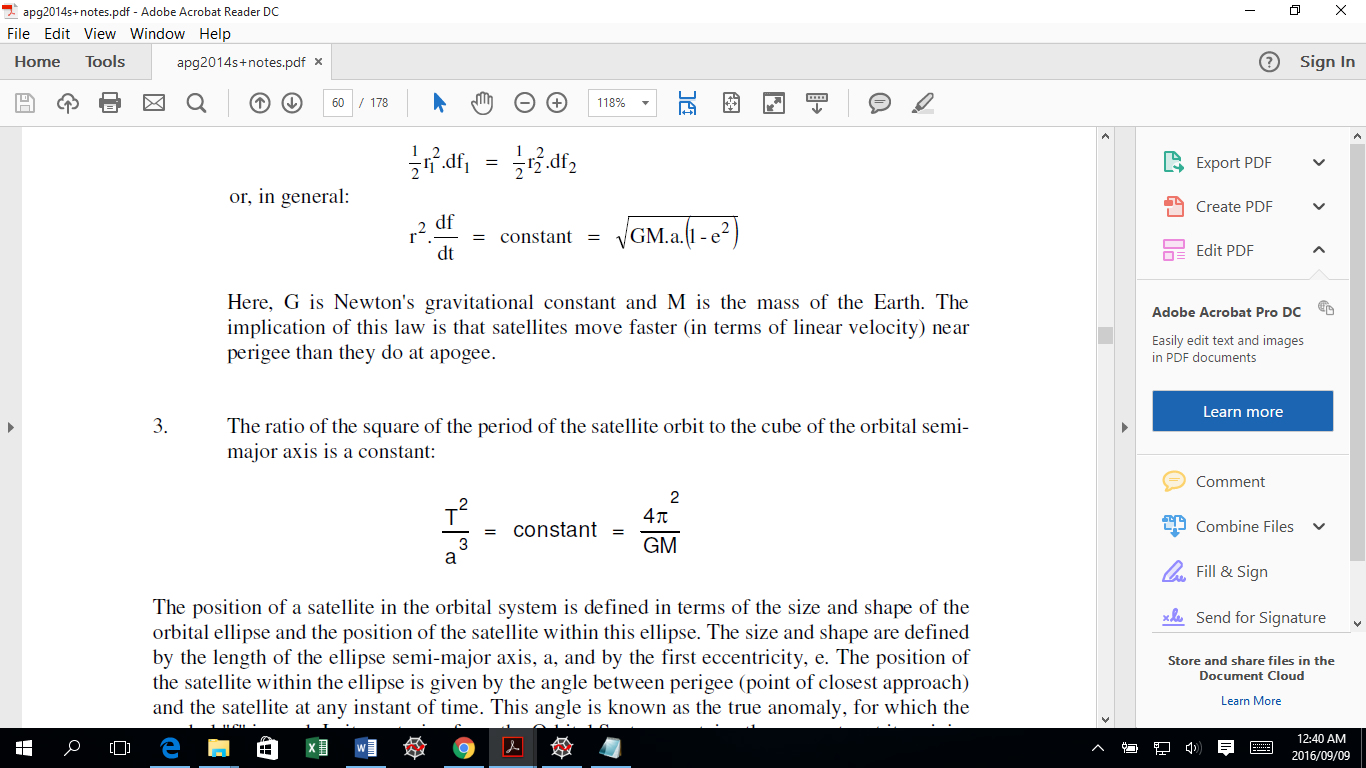
*Intro*

The outcome of the task is to transform Cartesian coordinates in the Right Ascension system into geographical coordinates in the conventional terrestrial system. The task is completed by deducing epoch values and error offsets provided by IERS (International Earth Rotation Service) for calculation of a series of rotation matrices.

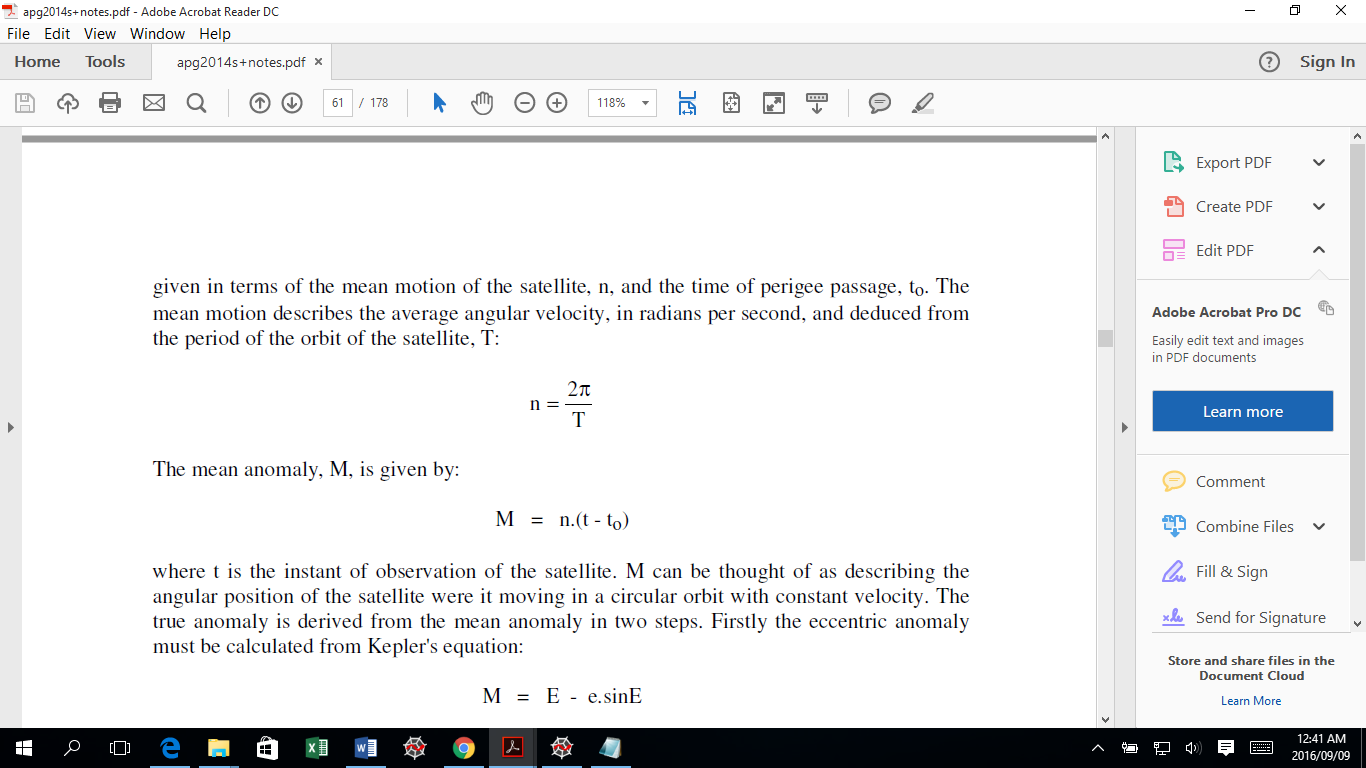
*Theory and Method*

We start off by correcting the previously calculated X, Y, Z data from tut2a. It has become apparent that these values were all processed under the assumption of ‘a’ (semi-major axis) being constant. To correct this minor error, a simple equation involving Keplers 3rd law is introduced. Note that since ‘a’ has been recalculated b was also re-calculated.

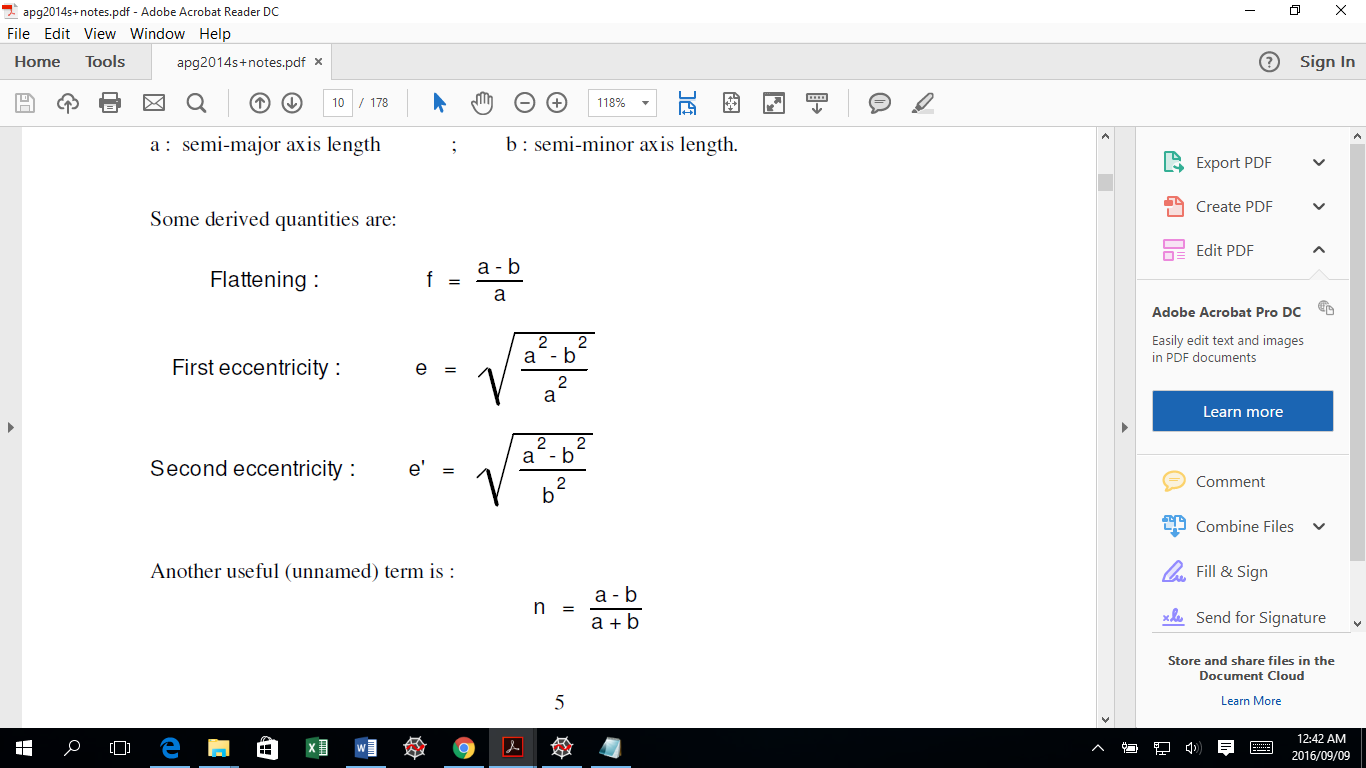
Keplers 3rd law states that “The ratio of the square of the period of the satellite orbit to the cube of the orbital semi-major axis is a constant:



And by rearranging the equation to calculate mean motion we can get the period of the satellite since n is known:



Equation of semi-minor axis b (since eccentricity and a is known):



Once the coordinates have been recalculated the next phase was initiated. To convert to GAST (Greenwich Apparent Sidereal Time) the Epoch number for each satellite must be attained. Information needed from IERS is the correction or difference data between UTC and UT1. By referencing the Epoch number to the Gregorian calendar via <http://www.epochconverter.com/days/2016>, the differences were easily obtainable from the provided text file and UT1 values could be calculated (see appendix A).

The process of conversion to GAST continues after Julian dates are obtained via the website <http://www.onlineconversion.com/julian_date.htm> recommended by the facilitators of the course. This procedure was done manually once the epoch data (UTC) was converted to hours, minutes and seconds in code tut2a. A series of calculation was then conducted to find the values of GMST (Greenwich Mean Sidereal Time) and GAST (see appendix A). The GAST value was then transformed into an Hour angle in DMS for compatibility in the rotation matrix.

GMST (in seconds at UT1=0) = 24110.54841 + 8640184.812866 \* T + 0.093104 \* T^2 - 0.0000062 \* T^3

Where T is in Julian centuries from 2000 Jan. 1 12h UT1:

T = d / 36525,

d = JD - 2451545.0

GAST = GMST + (– 0.24)

Decimal Degrees = (GAST/3600) \*15

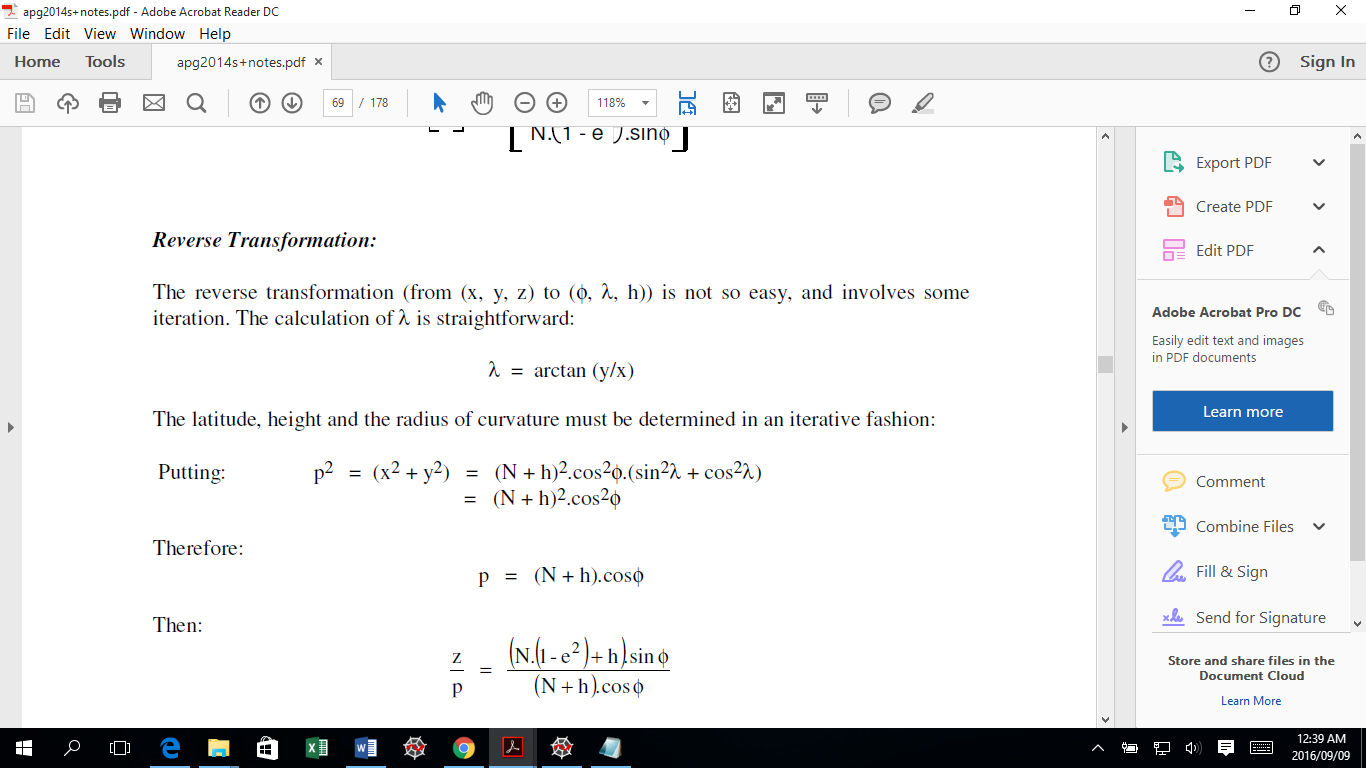
(There are 3600 seconds in an hour and 15 degrees in 1 hour angle)

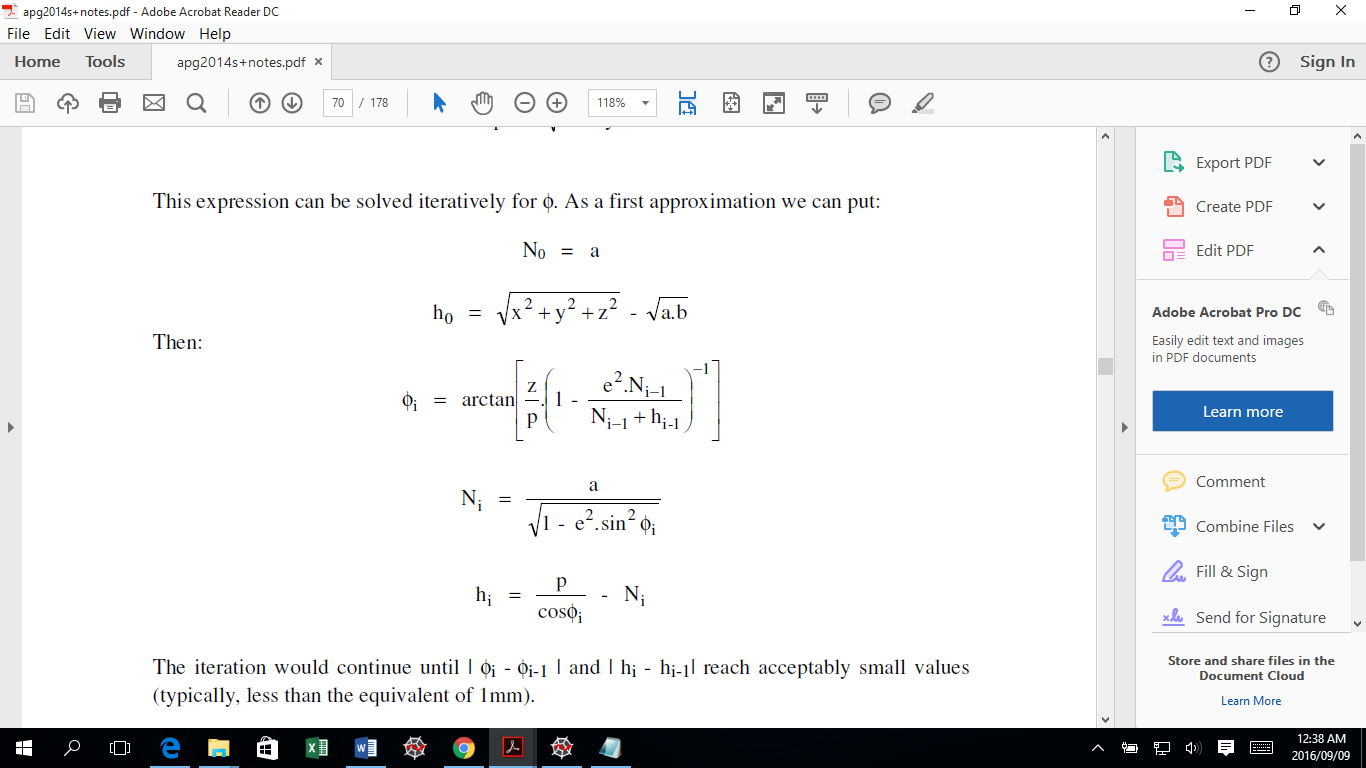
In addition to obtaining the difference data (UTC-UT1) from the IERS text file, xp and yp corrections were also filtered out for the required Epoch. These values were in arc-seconds and had to be converted by a factor of:

1 radian = 206264.806247 arc-seconds

The rotations were fairly easy and converted the RA system coordinates to CT system. The first rotation is about the z-axis using the GAST angle in radians. The second was about the x-axis using the radian angle of –yp. And lastly the rotation was about the y-axis using the radian angle of –xp. The reason for these rotations after the first rotation is to bring the z-axis coincident with the CT system z-axis.

Finally the coordinates can be converted to geographical coordinates following an alternative method to the bowring method (see appendix A).





*Code analysis*

Tut2b

The code contained in Tut2b.py relays the corrected RA coordinates by recalculation of semi-major and semi-minor axis. The code of Tut2a was reused and amendments were made to lines 96 -100 (open Tut2b.py to observe code changes).

Tut2b2

In Tut2b2.py things are a bit more complex as many text files have been accessed and only the required information has been sorted out and appended to lists and dictionaries. Lines 16 – 64 are mainly accessing and splitting data into lists for later use. The text files accessed include the “DATA3.txt” which includes all the new RA coordinates, semi-major, semi-minor axis and the eccentricity calculated in Tut2b.py, a while loop is immediately implemented thereafter in line 27 to sort all this data into a list. The “DATA.txt” file is also reopened to obtain epoch data as well as the “juliandates.txt” file to get the converted Julian dates information.

The tricky bit came with "tut+2b+DATA+XpYp.txt" as a lot of data had to be schemed to finally get the needed UTC-UT1 differences, xp and yp. Lines 56 -93 takes care of the sorting and looping through data. Four loops were created; two for pulling out a section of information and two for slicing these sections accordingly for the required data.

The main loop at line 96 runs 18 times to process coordinates for 18 different satellite epochs. Within the loop all the referenced data obtained is used to process UT1, GMST, GAST, CTS Cartesian coordinates and Geographical coordinates. After the calculation of GAST is done another dictionary (line 130) is created to store the GAST hour angle, xp and yp values. It should be noted that for UT1-UTC difference, xp and yp values careful cross referencing had to be made as more than one Epoch repeated or were close to each other hence for the repeated epochs the differences, xp and yp would be the same. Lines 118 -132 contain the relevant loops to address the condition of epochs repeating. The other code worth mentioning are the rotations (line 156 -166) and the geographical conversion with the iteration function which lie just beneath the rotations. For the iteration a function was created in which a previously set variable is declared global in the funtion, thereby updating the variable value each time the function is iterated.

*Conclusion and values*

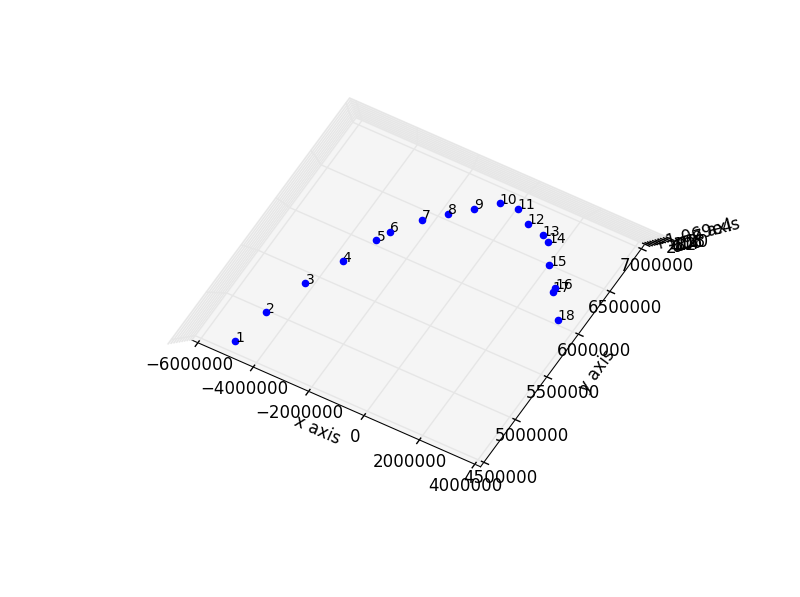
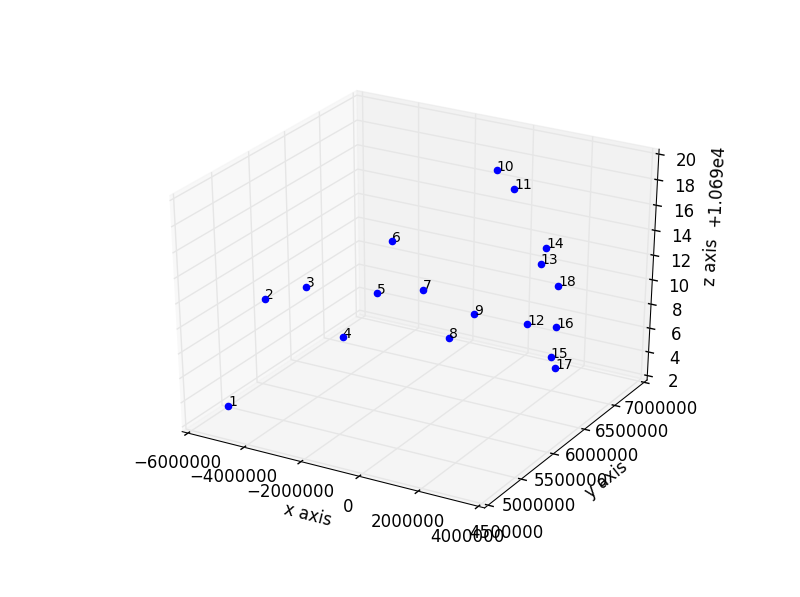
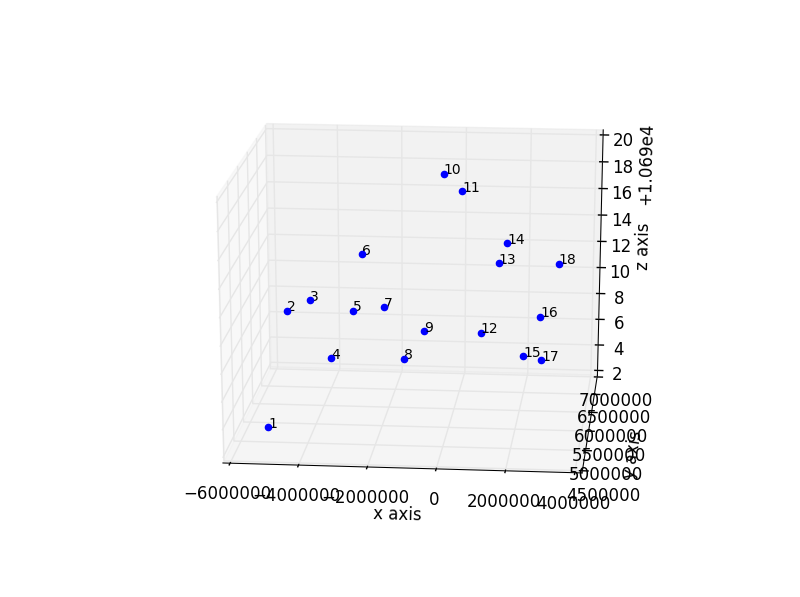
The values obtained are significantly close to the provided data although it can be improved. To check the information, all calculated values have been compared to other members of the course and the application of checks for the iteration have been done manually (e.g lat1 – lati-1, h1-hi-1). The difference of values calculated compared to the values provided is most likely the cause of the version of python being used (currently in use: v2.7) and can be resolved easily given more time or by importing the code to the more commonly used python version 3.5.

*Appendix A*

1. RA coordinates corrected for a and b:



1. Epoch data; UT1, GMST, GAST, GAST (dms), GAST (rad):
2. CTS coordinates and geographical coordinates:
3. 3D plots of RA system:



1. 3D plots for CT system

