Python Based GPS Standard Position Solution of RINEX files

INTRODUCTION

PURPOSE

OBJECTIVES

* Develop Python functions to process RINEX observation and navigation files and store the data in a data structure.
* Develop a Python function to determine satellite position using the navigation files.
* Develop Python functions to compute and apply ionospheric and tropospheric corrections, as well as satellite offsets, and apply them to observation pseudoranges where applicable.
* Run an epoch by epoch least squares adjustment to determine the location and clock offset of the receiver.
* Compute various DOP values using the covariance matrix from the least squares adjustment.
* Determine the accuracy and precision of the solution.
* Explain why the values were acquired.

SOFTWARE STRUCTURE

RINEX stands for Receiver Independent Exchange Format, it is stored in an ascii format and contains GNSS observations. The two RINEX files that the software reads are the observation and navigation files.

It was decided that the information stored in the observation and navigation files would be stored in a python dictionary data structure. A dictionary holds a set of unique “key-value” pairs that allow information to be accessed rapidly, without having to deal with messy array indexes. The implementation is not important, all that must be kept in mind is that dictionaries allow the information present in the two RINEX files to be stored and later rapidly accessed with meaningful keywords at a later date. The only downside of dictionaries is that there is no reliable way to iterate through a dictionary in a certain order, to counteract this, each dictionary that may need to be iterated over also includes a list of the keys corresponding to the key ‘LIST’ in the order they were placed into the dictionary.

OBSERVATION FILE READER

The observation RINEX file contains all of the GPS receiver’s observations that it has collected over its setup time. The file is split into two components, the header, which contains important information about the receiver itself, and the data section, which contains phase and pseudorange observations to various satellites at various epochs.

HEADER

The header contains information relating to the receiver, such as its position in a given coordinate system, its antenna height, or its make and model. It also contains clerical information pertaining to the observation set, and may include extra information present in comments. Most importantly, the header details the type of observations present in the RINEX file.

The header contains information for the first 60 characters (there are 80 on a line), the remaining 20 constitute a label which explains what is detailed in the previous 60 characters, each line is then terminated with a new line break (\n). To fully parse the header, the RINEX file is read in line by line. At each line, the label is checked to see what kind of header information is present on that line, this continues until END OF HEADER was reached, at that point, the software would switch from trying to read the header, to trying to read the observations. While the software attempts to read the header, it consults the last word of the label, and attempts to match it with a predefined list present in the software (with HTYPER method). In the event that two or more label types have the same last word, then the second last word is used instead. Once the label is identified, the software parses the line according to it (with ASSIGNDIC). The values present are extracted from the line with direct indexing (ex, character 0-16), are assigned meaningful “keys”, and are placed in the dictionary. In the event that a key already exists (or a multi-line label is reached), the value at that key is changed depending on the label type. For example, if a COMMENT label is reached, the current string value stored at the key ‘COMMENT’ has the new value appended to its end. In the event # / TYPES OF OBSERV occurs on more than two lines (with more than nine observation types), the list that holds the observation types has the extra values appended to it. While all information in the header section was stored in the obsHead dictionary, only a few important keys present in the header are used in the software, they are:

|  |  |  |
| --- | --- | --- |
| Key | Value | Description |
| ‘POS’ | Dictionary with sub keys  ‘X’, ‘Y’, and ‘Z’ | Contains the X, Y, and Z position of the receiver. |
| 'OBSTYP' | Dictionary with sub keys  ‘NUM’ and ‘OBS’ | ‘NUM’ contains the integer number of satellites  ‘OBS’ contains a list of the two digit alphanumeric GPS observation types (ie ‘C1’, ‘P1’) |

DATA SECTION

The data section contains the epoch by epoch observations by the receiver to the satellite. It contains measurements for several satellites, the measurements included are limited by what is present in the ‘OBS’ observation list mentioned above. The reader reads the first line, which has the epoch, the total number of satellites and the list of satellites. The epoch is made into a key in the form ‘YY:MM:DD:HH:MM:S.sssssss’, it maps to a dictionary that will be used include all satellite observations.. The reader then uses the number of satellites present and the number of observations expected to determine how many lines remain in the observation block. It then creates a list of the satellites and uses each three-digit PRN value as a key in the dictionary paired with the epoch key mentioned above. The reader reads line by line, and creates a dictionary that is matched to the correct PRN key for each satellite PRN. The dictionaries referred to by the PRN keys are then filled with key value pairs that refer to the observation information stored in the observation block. While all the information was stored in the observation dictionary only some of it was used, the key descriptions and the dictionary structure can be seen below.

|  |  |  |
| --- | --- | --- |
| Key | Value | Description |
| ‘YY:MM:DD:HH:MM:S.sssssss’ | Dictionary with sub keys  ‘PRN’,’PRN’,’PRN’, and ‘LIST’, ‘NUMSAT’ | The key refers to the observation block epoch. It maps to a dictionary which contains dictionaries referred to by satellite PRN numbers, each dictionary corresponds to a different satellite in the observation block.  ‘LIST’ maps to a list of all the satellite PRN numbers (for iterating)  ‘NUMSAT’ refers to the integer number of satellites |
| 'PRN' | Dictionary with sub keys  ‘C1’,’P1’,’P2’, etc | Contains the observations for the given satellite at the given epoch |

|  |
| --- |
| Dictionary Structure |
| {**'07:1:1:0:0:0.0000000'**: {  **'G06'**: {**'C1'**: 20857582.237,  **'P1'**: 20857582.016,  **'P2'**: 20857580.516},  ***'G07'***: {**'C1'**: 20418227.176,  **'P1'**: 20418226.502,  **'P2'**: 20418225.415},  **'G10'**: {**'C1'**: 23445664.779,  **'P1'**: 23445663.158,  **'P2'**: 23445663.052,  **'LIST'**: ['G06',  'G10',  'G07'],  **'NUMSAT'**: 10},  **'07:1:1:0:0:30.0000000'**: {  **'G06'**: {**'C1'**: 20868360.224, . . . (for all observation types and all epochs) |

CORRECTIONS

IONOSPHERIC CORRECTION

The effect due to the ionosphere can be mitigated in two ways. If using a single frequency receiver, the delay caused by the ionosphere can be modelled with the Klobuchar model. Since the software does not attempt to adjust single frequency receiver observations, the Klobuchar model will not be explained here. When using a dual frequency receiver, the delay due to the ionosphere can be nearly mitigated with a linear combination of the two observations, one on each frequency. The equation for the Ionofree linear combination that removes most of the ionospheric effect is:

|  |  |
| --- | --- |
| (S. Bisnath, 2014) | (1) |

PR is the ionofree corrected pseudorange, and are the pseudorange observations on two frequencies, and gamma is the squared ratio between the two carrier wave frequencies given by

|  |  |
| --- | --- |
| (S. Bisnath, 2014) | (2) |

LEAST SQUARES FILTER

The receiver coordinates and its clock offset must be estimated using an epoch by epoch least squares solution. This will allow for an iterative solution from an over determined system, that is, an equation system where there are more equations than unknowns (and thus, more than one correct answer). Least squares ensures that the solution will be the one that fits the model best.

The model equation is the formula for a pseudo range, shown below that describes the geometric explanation of a pseudorange. It includes a geometric component inside the square-root term, as well as a time component.

|  |  |
| --- | --- |
| (S. Bisnath, 2014) | (3) |

PR is the pseudorange between the satellite and the receiver, it can be likened to the geometric range, plus a correction in the time domain. The square root term is the geometric range from satellite to receiver, where terms with a subscript “sat” are the satellite coordinates, and the terms with a subscript “rec” are the receiver coordinates. The coordinates are assumed to be in the same coordinate system. The three receiver coordinates are part of what the least squares adjustment attempts to estimate. c is the speed of light in meters per second and was provided in the ICD. The variable is the time offset on the receiver, and is another parameter that must be solved for, finally, is the satellite vehicle clock offset, and provides a small error that must be compensated for.

Once the model has been chosen, it must be parameterized, that is, observables, unknowns, and constants must be chosen, after that, the mathematical adjustment model that will be used must be chosen.

The observables where chosen to be the pseudoranges (PR), they form a vector of observables , seen below

|  |  |
| --- | --- |
|  | (4) |

The receiver position in the x, y, and z, as well as its clock offset were chosen to be the unknowns

|  |  |
| --- | --- |
|  | (5) |

The constants were chosen to be the satellite positions, and the satellite clock offset.

Once the observables and unknowns were chosen, a suitable least squares model had to be chosen. Since all the observables are present on one side of the equation, and the other side of the equation is a function of only unknowns and constants, parametric adjustment was chosen, with the model

|  |  |
| --- | --- |
|  | (6) |

Because equation (3) is a non-linear equation (Variables are raised to a power other than 1 or -1), the equation must be expanded using Taylor series expansion using only the first two terms

|  |  |
| --- | --- |
|  | (7) |

Which consists of an initial estimate of the unknown parameter x, with a correction scaled by the difference between the difference of the “true” value of x and the initial estimate. The initial estimates were chosen to be the listed X,Y, and Z position of the station in the header, with a time offset of 0. In least squares adjustment, the difference between the “true” value and the estimated one is given by . The correction to the estimates is what is solved for in each iteration. Iterations usually continue until is significantly small, but in the case of this particular adjustment, is only iterated a few times. The formula shown in (7) also contains a part about the partial derivative of the function with respect to a variable x, in least squares adjustment, x is a vector, and a matrix must be composed of the partial derivatives of the functions with respect to each unknown in a vector. This matrix is called the first design matrix (A), and can be seen below.

|  |  |
| --- | --- |
|  | (8) |

Where is the pseudorange function of the first satellite, is the pseudorange function of the satellite, and so on. , , and are the unknowns mentioned in (5).

The partial derivatives of the pseudorange equation with respect to each unknown can be seen below.

|  |  |
| --- | --- |
|  | (9) |
|  | (10) |
|  | (11) |
|  | (12) |

One other matrix must be created before can be solved for, the misclosure vector w which is a measure of how well the unknown estimations fit with the model and the pseudorange observations. w is given by

|  |  |
| --- | --- |
|  | (12) |

Where is the matrix of observables, and is the matrix of pseudoranges computed using (3), with the estimated receiver coordinates and clock offsets.

Once the required matrices have been created, can be given by

|  |  |
| --- | --- |
|  | (13) |

Every time is calculated, the initial estimations must be updated, is added to , w and A are recomputed, and the iteration continues.

After a solution for an epoch has been found, the covariance matrix of the unknowns can be computed. It is given by

|  |  |
| --- | --- |
|  | (14) |

It is a 4x4 matrix which contains the covariances of each unknown to themselves and all other unknowns, shown below

|  |  |
| --- | --- |
|  | (15) |

It can be seen that the variances of the unknowns are present in the diagonal components, and that all off diagonal components are the covariance’s between unknowns. The diagonal variances will be used to compute the DOPS of the epoch, which will be explained in the next section.

The adjustment process is completed for each epoch. Initial estimations from the previous epoch are used as estimations for the next epoch.