

Document downloaded from:

<http://hdl.handle.net/10251/169019>

This paper must be cited as:

Martín Furones, ÁE.; Luján García Muñoz, R.; Anquela Julián, AB. (2020). Python software tools for GNSS interferometric reflectometry (GNSS-IR). *GPS Solutions*. 24(4):1-7.
<https://doi.org/10.1007/s10291-020-01010-0>



The final publication is available at

<https://doi.org/10.1007/s10291-020-01010-0>

Copyright Springer-Verlag

Additional Information

GPS Solutions

Free Python software tools for GNSS interferometric reflectometry (GNSS-IR)

--Manuscript Draft--

Manuscript Number:	GPSS-D-20-00148
Full Title:	Free Python software tools for GNSS interferometric reflectometry (GNSS-IR)
Article Type:	Original Article
Keywords:	GNSS-IR reflectometry; Python software; soil moisture; signal-to-noise ratio (SNR)
Corresponding Author:	Angel Martin-Furones Politechnical University of Valencia Valencia, Valencia SPAIN
Corresponding Author Secondary Information:	
Corresponding Author's Institution:	Politechnical University of Valencia
Corresponding Author's Secondary Institution:	
First Author:	Angel Martin-Furones
First Author Secondary Information:	
Order of Authors:	Angel Martin-Furones Raquel Luján, Master degree Ana Belen Anquela, Dr
Order of Authors Secondary Information:	
Funding Information:	
Abstract:	<p>Global Navigation Satellite System (GNSS) interferometric reflectometry, also known as the GNSS-IR technique, uses data from geodetic-quality GNSS antennas to extract information about the environment surrounding the antenna. Soil moisture monitoring is one of the most important applications of the GNSS-IR technique. This manuscript presents the main ideas and implementation decisions needed to write the Python code for software tools that transforms the RINEX format observation and navigation files into an appropriate format for GNSS-IR (which includes the SNR observations and the azimuth and elevation of the satellites), and to determine the reflection height and the adjusted phase and amplitude values of the interferometric wave for each individual satellite track. The main goal of the manuscript is to share the software with the scientific community to introduce new users to the GNSS-IR technique.</p>

Free Python software tools for GNSS interferometric reflectometry (GNSS-IR)

Angel Martín^{1*}, Raquel Luján¹, Ana Belén Anquela¹

¹Department of Cartographic Engineering, Geodesy and Photogrammetry

Universitat Politècnica de València

Camino de Vera s/n, Valencia 46022, Spain

*Corresponding author: aemartin@upvnet.upv.es

Angel Martin ORCID number: 0000-0001-9379-0694

Ana Belén Anquela ORCID number: 0000-0001-6024-3790

Abstract

Global Navigation Satellite System (GNSS) interferometric reflectometry, also known as the GNSS-IR technique, uses data from geodetic-quality GNSS antennas to extract information about the environment surrounding the antenna. Soil moisture monitoring is one of the most important applications of the GNSS-IR technique. This manuscript presents the main ideas and implementation decisions needed to write the Python code for software tools that transforms the RINEX format observation and navigation files into an appropriate format for GNSS-IR (which includes the SNR observations and the azimuth and elevation of the satellites), and to determine the reflection height and the adjusted phase and amplitude values of the interferometric wave for each individual satellite track. The main goal of the manuscript is to share the software with the scientific community to introduce new users to the GNSS-IR technique.

Keywords GNSS-IR reflectometry · Python software · soil moisture · signal-to-noise ratio (SNR)

Introduction

The use of a geodetic Global Navigation Satellite System (GNSS) antenna for soil moisture monitoring in the top 5-10 cm of the soil column, GNSS interferometric reflectometry (GNSS-IR), has become an interesting and complementary remote sensing technique due to its advantages over classical satellite or

airplane images. GPS-IR resolution is higher (a scale of approximately 1000 m² around the antenna); it can be used for continuous monitoring and is independent of weather conditions (the technique is valid in rainy and foggy conditions) and illumination (day or night).

GNSS satellites signals are transmitted in the L-band (microwave frequency), so the signal reflected by nearby surfaces and recorded by the antenna contains information about the environment surrounding the antenna. This information can be obtained by processing the signal-to-noise ratio (SNR) recorded in the antenna as interferograms.

The GNSS-IR technique, first developed by Larsson et al. (2008a, 2008b), and based on the procedure detailed in Larson et al. (2010), Larson and Nievinski (2013), Chew et al. (2014, 2015, 2016), Vey et al. (2015), Small et al. (2016), Wan et al. (2015), Chen et al. (2016), Roussel et al. (2016), and Zhang et al. (2017), is summarized as follows:

1. SNR observations from the GNSS satellites should be selected for satellites with elevation angles ranging from 5 to 30 degrees. This is one of the main drawbacks of the technique, it can be used only for low elevation satellites, since this range is the only one useful for obtaining a valid interferogram for soil moisture analysis. Data from satellite elevations below 5 degrees are discarded to avoid strong multipath effects in the SNR data.
2. Rising or setting satellite tracks are separated (or tagged for the post-processing), since the interferogram pattern can differ for rising and setting tracks of the same satellite.
3. SNR data are converted from observed dB-Hz units to a linear scale in Volts by the expression $SNR_{linear} = 10^{SNR/20}$ (Vey et al. 2015).
4. The reflected signal $SNR_{linear}^{reflected}$ is isolated by fitting a second-order polynomial to the SNR_{linear} in order to eliminate the direct satellite signal (Wan et al. 2015; Chew et al. 2016).
5. A Lomb-Scargle periodogram is computed from the $SNR_{linear}^{reflected}$ of each satellite track in order to check that only a clear primary wave is observed. Tracks with multiple peaks or low maximum average power should be discarded.
6. The selected tracks can be modelled as:

$$SNR_{linear}^{reflected} = A \cos\left(\frac{4\pi h}{\lambda} \sin e + \phi\right) \quad (1)$$

where A and ϕ are the amplitude and phase of the primary wave, λ is the GNSS signal wavelength, e is the satellite elevation, and h is the reflector height: the vertical distance between the GNSS antenna phase centre and the horizontal reflecting surface, which is assumed to be the distance between the antenna and the floor due to the low signal penetration on the ground (Chew et al. 2014; Roussel et al. 2016; Zhang et al. 2017). A and ϕ are estimated by a least-squares algorithm with initial values of 1 for A and 0 for ϕ .

7. The final step is to derive the relationship between soil moisture variations (more specifically, volumetric water content, $VWG \text{ m}^3/\text{m}^3$) and variations in the estimated phase of the primary reflected wave. To do this, in situ observations are needed as a reference data set. Those observations can be obtained from conventional water content reflectometer sensors (Vey et al. 2015, Larson et al. 2010), from soil data samples (Martin et al. 2019) or estimated based on the soil type (URL 1). Considering a linear relationship between GNSS-IR estimated phase variations and reference VWG variations, several weeks (or months) of both types of data are necessary to obtain a good relationship. The slope is adjusted using the satellite tracks for which the phase variations presented a stronger linear correlation with in situ soil moisture variations. For example, in Zhang et al. (2017) this correlation is set at 0.9, so only the ascending tracks of GPS satellites 13, 21, 24 and 30 and the descending tracks of GPS satellites 05, 09, 10, 15 and 23 are used. Finally, the slope to be used for all valid tracks should be the mean slope value obtained for the highly correlated satellite tracks.

In Nievinski and Larson (2014) an open Matlab/Octave source code is developed which can produce simulated SRN, carrier phase and pseudo-range GPS observations that agree with a multipath model for the near-surface reflectometry and positioning applications.

In Roesler and Larson (2015) a free software tool is presented to translate GPS (or GNSS) observations into a format usable for reflections research (written in Fortran 77), map GNSS-IR reflection zones around the antenna (written in Matlab), and estimate dominant frequencies and reflection height from GNSS data (written in Matlab).

In this manuscript, we explain the main ideas and implementation decisions to write the python code for a software tools that implement the first six steps of the previous procedure (the last one requires in

situ data to obtain an accurate linear relationship), with the goal of sharing it with the scientific community. The developed software works in Python 2.7 and Python 3.

Software development

The proposed software tools have two main modules:

The first module transforms the RINEX format observation and navigation files (Gurtner and Estey 2007) to a unique file containing the epoch of each observation, the satellite identification, SNR observation, and computed azimuth and elevation of the satellite from the navigation RINEX file. A line in the output file is written only if the satellite elevation falls between 5 and 30 degrees. The input can be a single file (one observation day, for example) or several (one week or month of continuous observation data separated by daily files), but the output is always a single file.

The input for the second module is the output of the first module; it generates an output file per satellite containing the reflection height and the adjusted values for phase and amplitude of the interferometric wave for each individual satellite track. The software also generates a graphical output for each track containing the direct SNR signal, the indirect SNR signal, the computed interferogram and the wave adjusted to the indirect SNR signal.

First module: Information extraction from RINEX observation and navigation files

The developed software works with the version 3 format of the RINEX observation and navigation files. The open source software GFZRNX can translate RINEX version 2 data to version 3 (Nischan 2016). Although we reference to GNSS in the manuscript, the first version of the software only works for GPS satellite constellation.

The user has to decide the frequency to be used prior to running the software; this can be done by opening a RINEX observation file and inspecting the header information. Once the frequency has been selected, the user should determine the observations related with this frequency and stored in the observation RINEX file that will be used in the process. Those observations are the SNR and the pseudo-range or code observation related to the selected frequency.

The algorithm reads the observation and navigation files stored in a folder (`\data\input\` by default) with the only condition being that the observation and navigation files have the same name (with different extensions) for the same observation period, and the names and extensions should follow the RINEX standard format: the name is composed of four characters for the station name, three for the day of the year and one character to identify the session. The extension is composed of two characters corresponding with the last two digits of the year and an “o” for the observation files or “n” for the GPS navigation file. The algorithm orders the input files in chronological sequence and opens them in order.

The algorithm reads each GPS satellite observation for every epoch for every observation file, and writes in the output file (located in the `data\output1\` folder by default):

1. A numerical identifier related with the epochs of the observed file. This integer identifier number starts with zero for all observations of the first epoch of the first observation file, increases one by one, and ends with the last epoch of the last observation file. This numerical number can be used as the identifier in the second software module. The software writes as many lines, with the same numerical identifier, as there are observations from different satellites in the same epoch.
2. The time related to the previous lines. The year, month, day and a float number containing the hour (calculated with the integer hour, integer minute and float seconds information from the RINEX observation file).
3. The two-digit satellite GPS numerical identifier.
4. The observed SNR.
5. Based on the observation epoch, the pseudo-distance or code observation to the satellite, the station coordinates located in the header of the observation file and the navigation file, the algorithm computes the azimuth and elevation of the satellite from the antenna. This calculation must determine the emission time and the satellite coordinates in the Earth-Fixed-Earth-Centered (ECEF) reference system. For the emission time, an iterative “pseudo-range-based algorithm” is used (Sanz et al. 2011). The algorithm described in Leick et al. (2015) on pg. 240 is used for the ECEF satellite coordinates computation. Finally, the Earth’s rotation during the signal travel is taken into account to obtain the final satellite coordinates in the receiver time. Based on the ECEF coordinates of the antenna and the satellite, the azimuth and elevation are computed and

stored in the output file. The module that computes the azimuth and elevation requires two extra modules, one to compute the Julian and GPS time and other to compute the spherical geodetic coordinates from the geocentric coordinates and compute the azimuth and elevation from the geodetic coordinates of the antenna and the satellite.

In the output file, only data from satellites with elevations between 5 and 30 degrees are stored.

Second module: GPS-IR reflector height calculation and wave adjustment to observed SNR

The second software module used the first module's output file as an input file.

The program identifies all valid tracks by satellite and will solve steps 2 to 6 described in the introduction section.

First, the software creates an output folder structure: one folder for each satellite identified by its numerical identifier, creating 32 folders numbered from 1 to 32. If these folders already exist, the program will fail, since they are created during execution; this prevents previous results from being overwritten or generating more files within the folders with each successive execution.

Before running the software, the user must set some internal input parameters: The time interval between observations (can be obtained from the observation file header), the minimum and maximum satellite elevations (in case the input file contains values with other elevations, these parameters are set again, by default, to 5 and 30 degrees), the minimum and maximum azimuth of the satellite tracks considered (this allows the user to focus on a certain area around the antenna; these parameters are set by default to 0 and 360 degrees), the antenna height (measured from the ground to the antenna's reference mark in metres), the working frequency length (0.1904 m for L1, 0.2443 m for L2), a minimum angle of elevation to be covered by the satellite to consider the track as valid (set by default to 10 degrees), and a minimum number of epochs without observations to assume the observations belong to the same or a new satellite track. In certain cases, some epoch information is lost in the observation file due to signal interruptions or a malfunction, so the software identifies satellite observations that are temporally separated by less than the number of epochs set in this parameter as belonging to the same track. The value depends on the time interval between epochs; it can be set to 1 if the time interval between observation periods is 30 seconds, 7 if the interval is 5 seconds or 35 for 1 second.

The software selects each track and tags it as a rising or setting satellite tracks; it then performs the processes described in the following paragraphs.

Steps 3, 4, 5 and 6 described in the introduction section are performed for every track; finally, A and ϕ parameters and their standard deviations are estimated by the least-squares algorithm.

One of the biggest advantages of using Python is the many free scientific libraries available for easy access. The most complex process in this case is computing of the Lomb-Scargle periodogram: step 5 in the introduction section. For this, the `LombScargle` function of the Astropy library is used (URL 2). The input for the computing the Lomb-Scargle periodogram for each track are the sine of the satellite elevation angle on the X axis, and the $SNR_{linear}^{reflected}$ on the Y axis. With this configuration, the result converts the frequency into the antenna reflector height in metres on the output X axis. However, due to the use of the sine of the satellite elevation angle, the grid spacing on the X axis is irregular. To determine the appropriate grid spacing to use, the library introduces an option through keywords passed to the `autopower()` method. By default (after some experimental probes), the highest frequency is fixed to two times the average Nyquist frequency: `Nyquist_factor=2` in the code. Additionally, the maximum reflector height allowed in the output periodogram is fixed to 2.5 metres, but this parameter can be changed if the antenna elevation is higher. A theoretical discussion of the frequency extraction from GNSS-IR can be found in Roesler and Larson (2018).

The most important step in this module is to establish conditions for selecting "good" tracks. In this case, based on different experiments, a track is considered valid if: the satellite track contains more than 30 minutes of observation and more than the minimum previously set angle of elevation value to be covered by the satellite, the power of the dominant frequency is 6 times larger than the media background noise, and the adjustment of the theoretical wave to the $SNR_{linear}^{reflected}$ signal presents a residual vector with a mean less than 1.3 Volts/Volts and a standard deviation less than 25 Volts/Volts.

An output file is generated, in which each line corresponds with one track and contains: a numerical identifier related to the epochs of the observed file, a yes/no text field related to the previous conditions (yes indicate "good or valid" tracks), the epoch time, month, day, satellite identification number, initial satellite track azimuth, final satellite track azimuth, initial satellite track elevation, final satellite track elevation, adjusted A value and its standard deviation, adjusted ϕ value (in degrees) and its standard deviation, the computed reflector height and a satellite-falling/satellite-rising text field. A second output

file is also created, containing the same information but only for the tracks considered “good or valid” tracks.

For all valid and invalid tracks, a plot is generated containing four figures: the complete signal (SNR), the reflected signal ($SNR_{linear}^{reflected}$), the Lomb-Scargle periodogram for the reflected signal, and the reflected signal with the adjusted wave. The figure name includes the numerical identifier, year, month, day, satellite identifier, the initial azimuth and the final azimuth to allow the user to clearly and reliably identify the corresponding line in the output files.

GPS-IR example

The data we provided with the software is part of an experiment performed in the installations of the Cajamar Center of Experiences, Paiporta, Valencia, Spain, Figure 1, Martín et al. (2019). We provide 7 days (from a total of 66 in the experiment) of GPS observations with a geodetic GNSS receiver (Trimble R10, from the Department of Cartographic Engineering, Geodesy and Photogrammetry of the Universitat Politècnica de València). RINEX observations (with a 5 seconds sample rate) and navigation files are provided. L1 frequency is used for the example.

The experiment was performed from December 3, 2018 to February 6, 2019. The observations files included with the software are daily data files from January 14 to 20, 2020, and the height of the antenna to the ground was 1.8 m.

There was no occurrence of rain during the observation.

In addition to the software, the first software module’s output file is provided, along with the figures and the two output files for satellite number 15 of the second module, so that the user can validate the software operation. Figure 2 shows the track with identifier epoch 88518, 19 January 2019, starting azimuth of 281 degrees and ending azimuth of 300.1 degrees. Specifically, Figure 2a is the SNR data in dB-Hz, Figure 2b is the indirect SNR data in Volts, Figure 2c is the Lomb-Scargle periodogram for the SNR reflected signal and Figure 2d depicts the SNR reflected signal with the adjusted wave in Volts.

With the established criteria for selecting the correct tracks, the confusion matrix obtained for all satellites during the 7 observation days presents the following values: 45% of tracks during the seven days are valid and classified as such, 43% are invalid and are classified as such, 9% are invalid tracks classified as valid and only 3% of the tracks are valid but classified as invalid.

Conclusions/final remarks

Our intention is to share the software with the scientific community to introduce new users to the GNSS-IR technique. This technique can be used not only for soil moisture monitoring but also, for example, vegetation water content monitoring (Wey et al. 2015), snow depth measurements (Larson et al. 2009), or tide measurements (Roussel et al. 2015), illustrating that users can easily adapt the software for other purposes.

The software can also be extended to work with GALILEO, GLONASS and BEIDU GNSS satellite observations and frequencies, by modifying only the first software module.

Availability of data and material

Software tools and GNSS data can be freely download from:

<https://gnss-ir-2019.s3-us-west-2.amazonaws.com/software/SNRpy.rar>

Author contribution

Angel Martín and Raquel Luján wrote the software tools, Angel Martín and Ana Belén Anquela have designed, collected and processed the observations of the Experiment in Paiporta, Angel Martin wrote the manuscript.

Competing interests

The authors declare that they have no conflict of interest.

References

- Chen Q, Won D, Akos DM, Small EE (2016) Vegetation using GPS interferometric reflectometry: experimental results with a horizontal polarized antenna. *IEEE J Select Top Appl Earth Obs Rem Sens* 9(10):4771-4780
- Chew CC, Small EE, Larson KM, Zavorotny VU (2014) Effects of near-surface soil moisture on GPS SNR data: development and retrieval algorithm for soil moisture. *IEEE T Geosci Rem Sens* 52(1):537-543

- Chew CC, Small EE, Larson KM, Zavorotny UZ (2015) Vegetation sensing using GPS-interferometric reflectometry: theoretical effects of canopy parameters on signal-to-noise ratio data. *IEEE T Geosci Rem Sens* 53(5):2755-2764
- Chew CC, Small EE, Larson KM (2016) An algorithm for soil moisture estimation using GPS-interferometric reflectometry for bare and vegetated soil. *GPS Solut* 20(3):525-537
- Gurtner W, Estey L (2007) RINEX: the receiver independent exchange format version 3.03. <ftp://igs.org/pub/data/format/rinex303.pdf>. Accessed 1 Sep 2018.
- Larson KM, Small EE, Gutmann ED, Bilich AL, Axelrad A, Braun JJ (2008a) Using GPS multipath to measure soil moisture fluctuations: initial results. *GPS solut* 12(3):173-177
- Larson KM, Small EE, Gutmann ED, Bilich AL, Braun JJ, Zavorotny VU (2008b) Use of GPS receivers as a soil moisture network for water cycle studies. *Geophys Res Lett* 35, L24405. <https://doi.org/10.1029/2008GL036013>
- Larson KM, Gutmann E, Zavorotny VU, Braun J, Williams M, Nievinski FG (2009) Can we measure snow depth with GPS receivers? *Geophys Res Lett* 36, L17502. <https://doi.org/10.1029/2009GL039430>
- Larson KM, Braun JJ, Small EE, Zavorotny VU (2010) GPS multipath and its relation to near-surface soil moisture content. *IEEE J Selec Top Appl Earth Obs Rem Sens* 3(1):91-99
- Larson KM, Nievinski FG (2013) GPS snow sensing: results from the EarthScope plate boundary observatory. *GPS solut* 17(1):41-52
- Leick A, Rapoport L, Tatarnikov D (2015) *GPS satellite surveying*. John Wiley & Sons, fourth edition, 840 pp
- Martín A, Baixauli C, Blanc A, Ibañez S, Anquela AB (2019) Soil moisture monitoring using GNSS-IR technique and low-cost sensors. A case study in Valencia (Spain). 7th international colloquium on scientific and fundamental aspects of GNSS. 4-6 september 2019, Zurich, witzerland. <https://atpi.eventsair.com/QuickEventWebsitePortal/19a07---7th-gnss-colloquium/website> accessed on April, 2020
- Nievinski GG, Larson KM (2014) An open source GPS multipath simulator in Matlab/Octave. *GPS Solut* 18:473-481. <https://doi.org/10.1007/s10291-014-0370-z>
- Nischan T (2016) *GFZRNEX - RINEX GNSS Data Conversion and Manipulation Toolbox (Version 1.05)*. GFZ Data Services. <http://doi.org/10.5880/GFZ.1.1.2016.002>

- Roesler C, Larson KM (2018) Software tools for GNSS interferometric reflectometry (GNSS-IR). *GPS Solut* 22(80). <https://doi.org/10.1007/s10291-018-0744-8>
- Roussel N, Ramilien G, Frappart F, Darrozes J, Gay A, Biancale R, Striebig N, Hanquiez V, Bertin X, Allain A (2015) Sea level monitoring and sea estimate using a single geodetic receiver. *Remote Sens Environ* 171:261-277
- Roussel N, Frappart F, Ramillien G, Darrozes J, Baup F, Lestarquit L, Ha MC (2016) Detection of soil moisture variations using GPS and GLONASS SNR data for elevation angles ranging from 2° to 70°. *IEEE J Selc Top Appl Earth Obs Rem Sens* 9(10):4781-4794
- Sanz J, Juan JM, Hernández-Pajares M (2011) Emission time computation. https://gssc.esa.int/navipedia/index.php/Emission_Time_Computation. Accessed may 2014
- Small EE, Larson KM, Chew CC, Dong J, Ochsner TE (2016) Validation of GPS-IR soil moisture retrievals: comparison of different algorithms to remove vegetation effects. *IEEE J Selc Top Appl Earth Obs Rem Sens* 9(10):4759-4770
- URL1 <https://www.unavco.org/data/gps-gnss/derived-products/pbo-h2o/documentation/documentation.html#soil>. Accessed on December 2019.
- URL2 <https://docs.astropy.org/en/stable/timeseries/lombscargle.html>. Accessed on January 2017.
- Wan W, Larson KM, Small EE, Chew CC, Braun JJ (2015) Using geodetic GPS receivers to measure vegetation water content. *GPS Solut* 19:237-248
- Wei W, Larson KM, Small EE, Chew CC, Braun JJ (2015) Using GPS receivers to Measure Water content. *GPS Solut* 19(2):237-248. <https://doi.org/10.1007/s10291-014-0383-7>.
- Zhang S, Roussel N, Boniface K, Ha M C, Frappart F, Darrozes J, Baup F, Calvet JC (2017) Use of reflected GNSS SNR data to retrieve either soil moisture or vegetation height from a wheat crop. *Hydrol Earth Syst Sci* 21:4767-4784

Angel Martin received his Ph.D. degree in Geodesy and Cartography at Universitat Politècnica de València (UPV) in 2001. Actually, is a senior scientist and a lecturer professor at the Department of Cartographic Engineering, Geodesy and Photogrammetry at UPV, his main research areas are devoted at GNSS, Physical Geodesy and Big Data and Machine Learning applied to Geosciences.

Raquel Lujan received her double Master Degree from the Universitat Politècnica de València (UPV) and the University of Applied Sciences in Karlsruhe at 2019. Actually, she is a PhD student in the Department of Cartographic Engineering, Geodesy and Photogrammetry at UPV where she works in the field of GNSS.

Ana Belén Anquela received her Ph.D. degree in Geodesy and Cartography at Universitat Politècnica de València (UPV) in 2001. Actually, is a senior scientist and a lecturer professor at the Department of Cartographic Engineering, Geodesy and Photogrammetry at UPV, her main research areas are devoted at GNSS and geodetic monitoring. Actually, she is the director of the geomatic school at UPV.

Figure captions

Figure 1: Geodetic-quality GNSS antenna located in the experiment zone.

Figure 2: GPS satellite 15 track example results: a) observed SNR data in dB-HZ, b) reflected SNR signal in Volts, c) Lomb-Scargle periodogram for the SNR reflected signal, d) SNR reflected signal with the adjusted wave.



