# GDP

An Open Source **G**NSS **D**ata **P**reprocess Toolkit

## User Manual

written by Zhengsheng Chen & Yang Cui

Xi'an & Chongqing China

April 2020

# Content

# 1 Introduction

Over the last decades, GNSS signals, frequencies and constellations become more and more abundant with the advent of BDS and Galileo as well as the modernization of GPS and GLONASS. The quality of GNSS raw observation are easily affected by several factors, such as the availability of satellite signal, transmission error, multipath, receiver performance and others. Due to the diversity of observation environment, GNSS data preprocessing plays an important role for subsequent data processing. Proper preprocessing of GNSS raw observations is the premise of high precision GNSS positioning or orbit determination. GDP is a multi-GNSS data parallelizable preprocessing software design to process raw GNSS observation data based on RINEX 2.x to 3.0x standard. Written and design in object-oriented programming language, it mainly includes multi-GNSS file or IGS products automatically acquisition, format conversion, file selection, data visualization and analysis on GNSS observation files. It is designed as a multi-platform application and provides both Window form interface and command shell interface in Windows, Linux or MacOS operating system. This document provides a brief operating guide. If there are higher requirements, welcome to join the developer sequence to further improve the program code.

# 2 List of GDP package

After downloading and unzip, the program list is as follows


Figure 1 list of GDP V1.0 program package

Among them, there are two runnable programs, a source code package and a protocol. Runnable programs include window interface program (in the GdpWinUI directory) and command window program (in the GdpShell-netcoreapp3.1 directory).

The sample data is stored in the "…\GdpWinUI\Data" directory, the program will automatically load "hers0010.18O" observation data after running.

# 3 Operating environment of GDP

## 3.1 Operating system and environment

The window application of GDP need to run in a Windows .NET 4.7 environment, which you can obtain from the following URL. If you use windows 10, you can run it directly.

| https://dotnet.microsoft.com/download/dotnet-framework/net47 |
|---|

The direct download link for the .NET 4.7 runtime as follow:

| https://dotnet.microsoft.com/download/dotnet-framework/thank-you/net47-web-installer |
|---|

Command line program of GDP can run on Windows, Linux, MacOS and other operating systems, but require support from .NET Core 3.1. You can get it from the following website.

| https://dotnet.microsoft.com/download/dotnet-core/3.1 |
|---|

For the normal user, just download the .NET Core Runtime to run GDP, as shown in Fig.2.

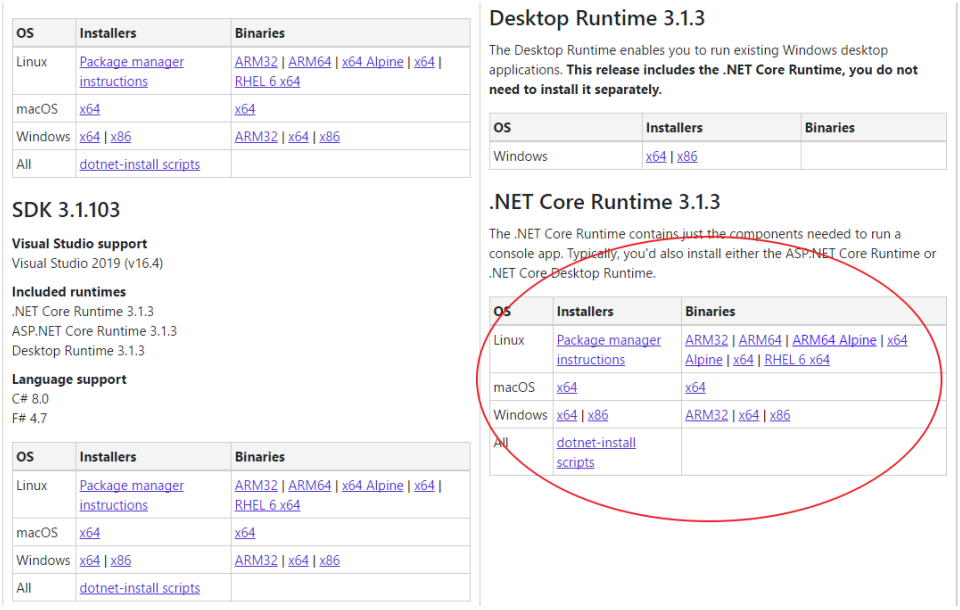


Figure 2 download interface for .NET Core Runtime

Once the environment is successfulyu installed, you can run the GDP program directly.

## 3.2 Program running interface

1）Window application program

In the GdpWinUI directory, double-click on the GdpWinUI.exe to run the desktop interface as shown in Fig.3.


Figure 3 main interface on the desktop side

2）Command line program

After ensuring that the installation of .NET Core is completed, open the Shell window, under the directory "GdpShell-netcoreapp3.1", type the following command. If the installation is correct, the interface of the command line program is shown in Fig.4.

```
dotnet gdp.dll -v
```

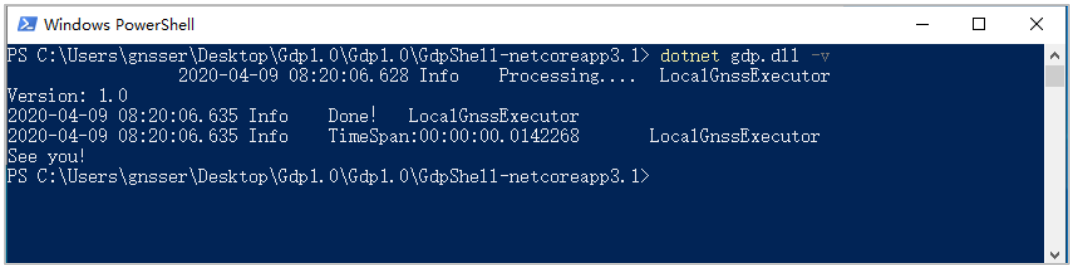


Figure 4 command line program

The sample data required for the GDP program come from IGS and is stored in the Data directory of the running program.

# 4 Function of window program and example

## 4.1 Downloader

Click "Downloader" in the main interface to open the "IGS products downloader" sub-interface, as shown in Fig.5. The interface consists of three parts of view, which are respectively setting,

executing and result showing.

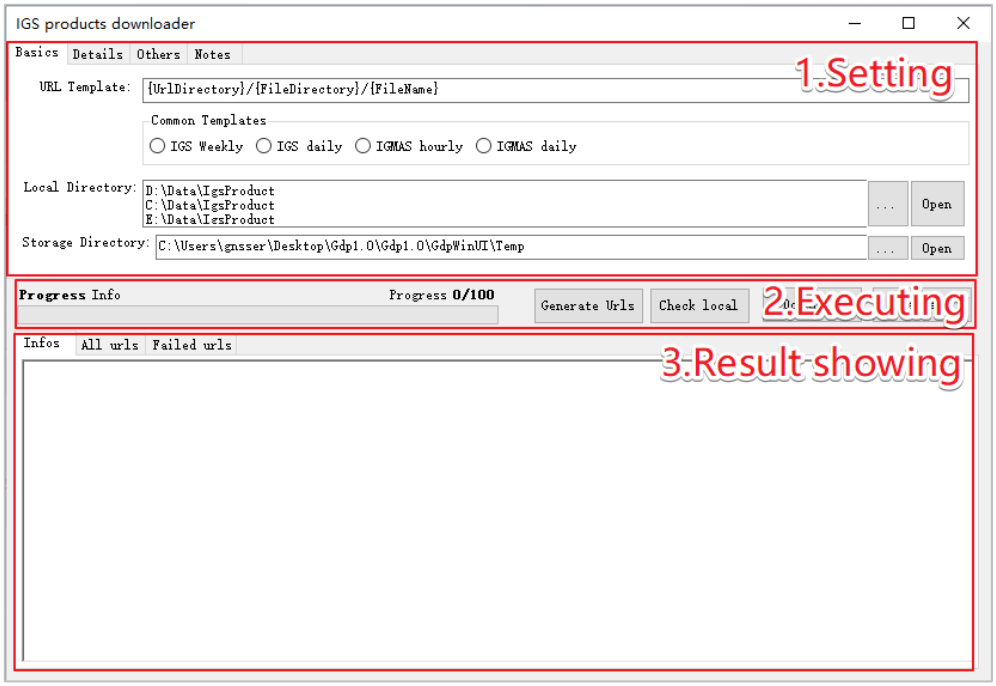


Figure 5 products downloader interface

Download the data according to the following steps:

**Step1: Generate the URL of the resource to be downloaded**

The URL is generated based on the content of the Setting page. The template EL (Expression Language) is adopted, and the template parameters are identified by the braces of "{}". Parameters can be set manually. Supported parameters are as follows Tab.1, where the time variables, including {Year}, {Month}, {Day}, {week}, etc., are based on GPST (GPS time) as the same with IGS, which are start form the epoch of 1980.01.06.0; The variable marker of {SourceName} can be expressed as igs, igr, com, etc., which stands for different IGS analysis center; The variable marker {ProductType} can be expressed as sp3, clk, erp, etc., which stands for different product file type. With the EL templates and the variable, GPD can automatically build URLs, and store in one text file. Then you can download these resources by download tools in GDP or other downloaders.

Tab. 1 EL parameter name and meaning

| parameter name | parameter description | parameter name | parameter description |
|---|---|---|---|
| {UrlDirectory} | internet resource directory | {Week} | GPS 周 |
| {SourceName} | data source identity | {DayOfWeek} | day of week |
| {ProductType} | product type | {DayOfYear} | day of year |
| {Year} | four digit year, such as 2000 | {WeekOfYear} | week of year |
| {SubYear} | two digit year, such as 99 | {Hour} | hour |
| {Month} | month | {Minute} | minute |

| {Day} | day | {Second} | integer seconds |
|---|---|---|---|
| {BdsWeek} | BDS week | | |

The program provides four templates by default, which are IGS daily and weekly products and IGMAS daily and weekly products.

> Note: these templates are time-sensitive. If they are not accessible, please replace them with available parameters. It can also be fed back to the developer for modifications.

In the detailed setting page, see Fig.6, UrlDirectory is the network directory to be download, such as IGS data center. SourceName is the parameter data identifier, such as the three glyphs of IGS analysis center. ProductType is the products type, such as sp3, clk, etc. The parameter of TimePeriod is the period of download. The Step parameter is the time step of products.

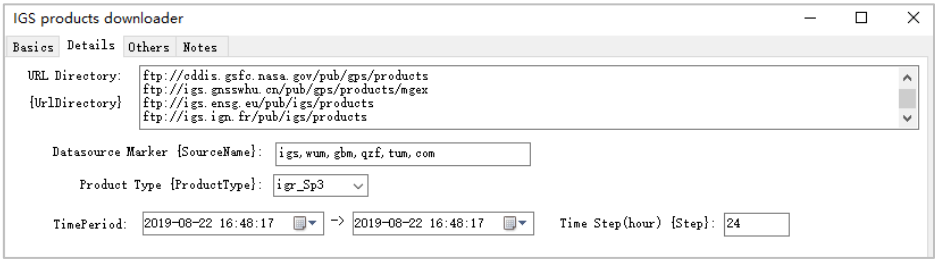


Figure 6 details setting view of products download

Once set up, click on “Generate Urls” to generate the address. The “All Urls” panel in the results area can be viewed, as shown in Fig.7.

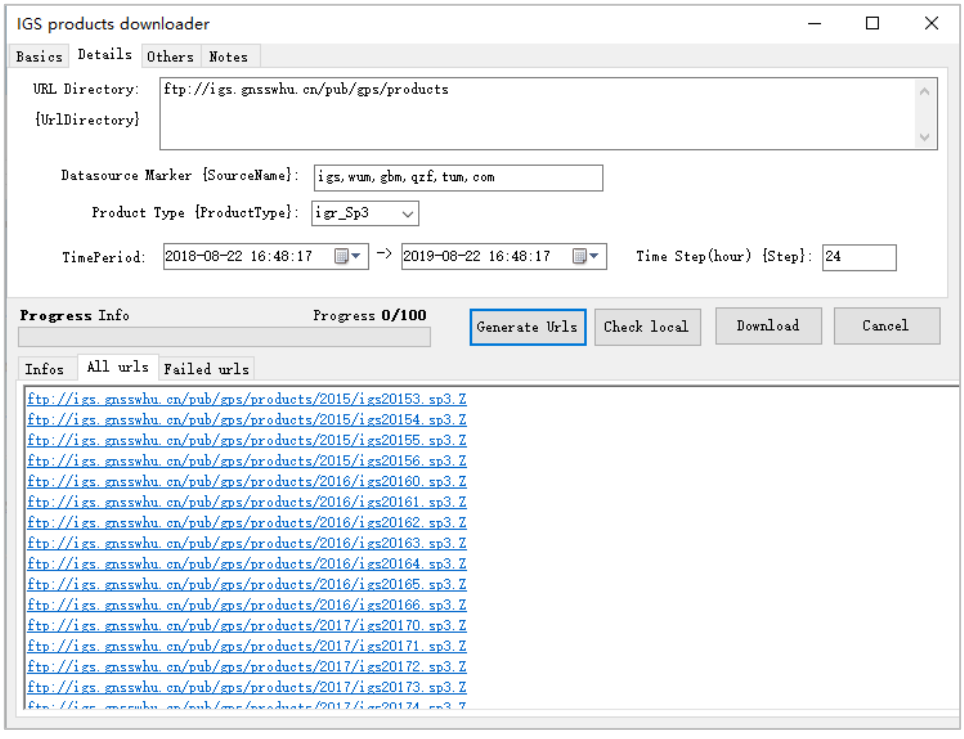


Figure 7 generate Urls

**Step2: Check existing resources (avoid duplicate downloads)**

In the “Basic” panel of setting page, you can set the local directory. Click the “Check local” button and the program will look for files with the same name in the specified local directory. Existing files will be removed from the “All urls” panel, as shown in Fig.8.

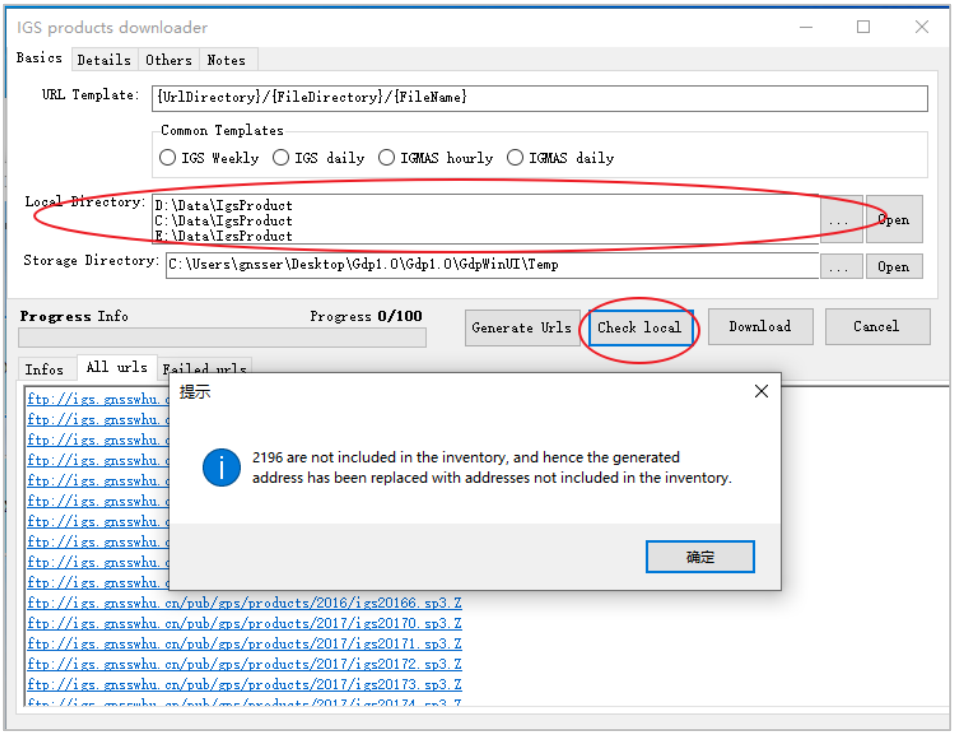


Figure 8 check existing resources

**Step3: download**

Set up the local storage directory and then click the “Download” button, the download progress is shown in the “Info” panel, and the failed addresses are viewed in “Failed urls”. The download process is shown in Fig.9.

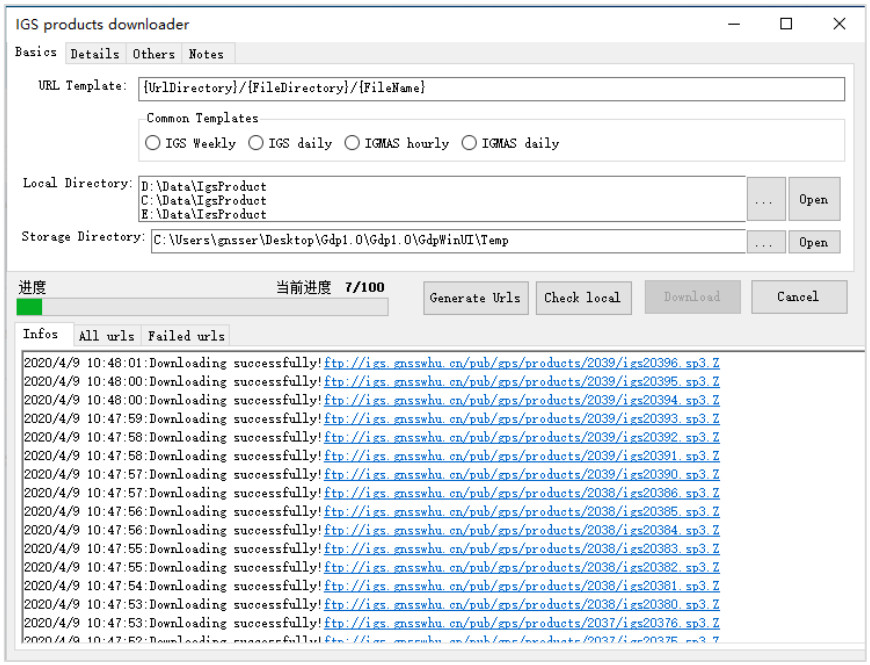


Figure 9 download the products

## 4.2 Formator

The Formator window of GDP is simple, as shown in Fig.10, but has complex formatting capabilities. The input/output panel can optionally specify the observation files (O or RNX) to format and determine the result output path.

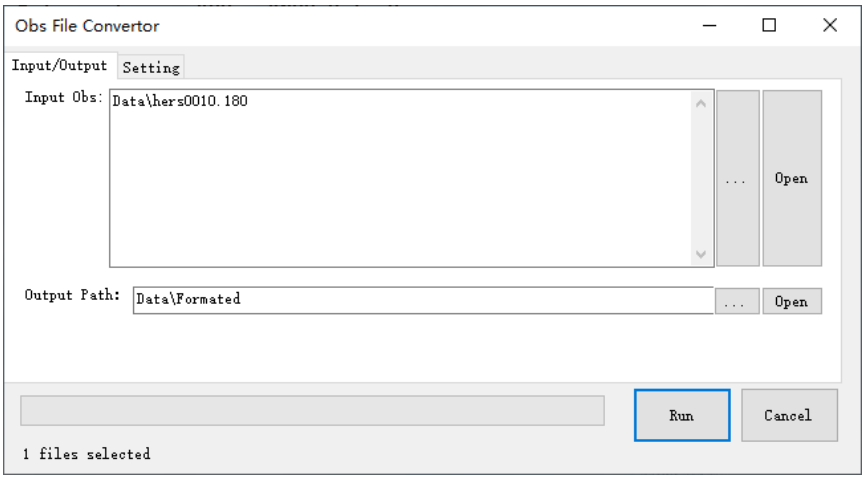


Figure 10 RIENX file format interface

Click the “Details” button in the Setting panel to set detailed formatting parameters, as shown in Fig.11. If you are interested in only a few satellite navigation systems in the multi-system files, select the corresponding navigation system in “Enabled System” and enable it. Similarly, the observation value type, observation period, sampling rate, version of output file, observation code and frequency can be set. In “Other”, details can be further set, such as deleting the data of some

satellites, removing the data whose arc segment is less than the threshold value, deleting observation value that cannot form the ionospheric combination, naming format of the station files, etc.

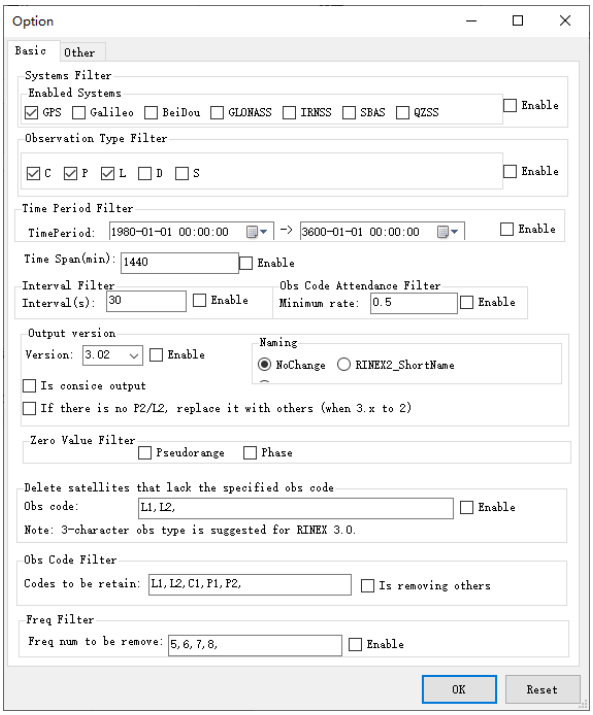


Figure 11 format options of observe file

## 4.3 Selector

Filter the observation files that meet the requirements in the “Selector” panel, which differs from the format panel in that the selector does not change the observation files itself. As shown in Fig.12, click “…” button, select input all GNSS files, specify the output directory through the “Output Folder”, and set the relevant selection parameters through the Details in Setting.

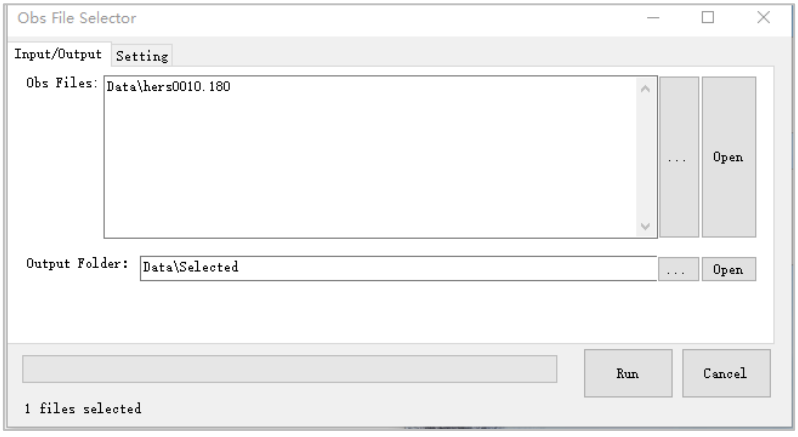


Figure 12 panel of select the observation file

The “Selector” panel does not make any changes to the observation files, but simply find and output the files that meets the requirements. The specific parameters settings are shown in Fig.13. You can

set the navigation satellite system, observation code, time period, station name, minimum observation epoch format, file size, number of satellites, frequency, etc.

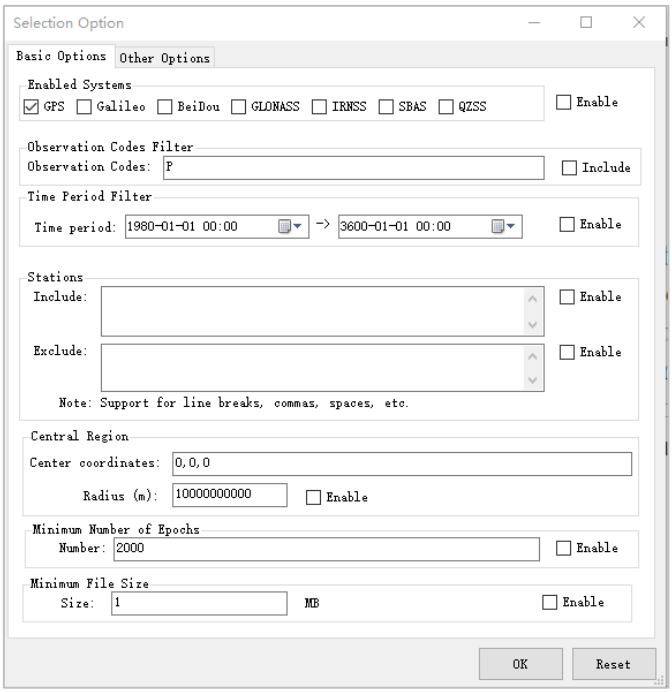


Figure 13 Selection settings of observation files

## 4.4 Viewer&Editor&Analyzer

After clicking the "Viewer&Editor&Analyzer" button on the main page, you can open the observation file view and edit panel, under which you can conduct a detailed view and analysis of a specific observation data file. For example, click "Read" to open the sample data directly, as shown in Fig.14.

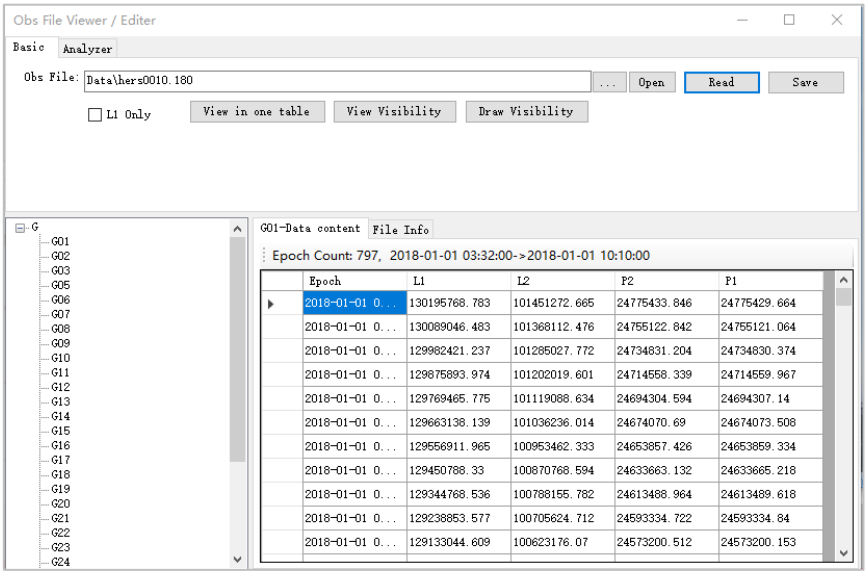


Figure 14 reading interface of sample data

The lower left corner of the panel shows the satellites number observed in the file, and the lower right corner of the panel shows the data information of one satellite (G01 satellite, for example). The file name, antenna, sampling rate, number of satellites, observation code and other basic information of the file are displayed in “File Info”.

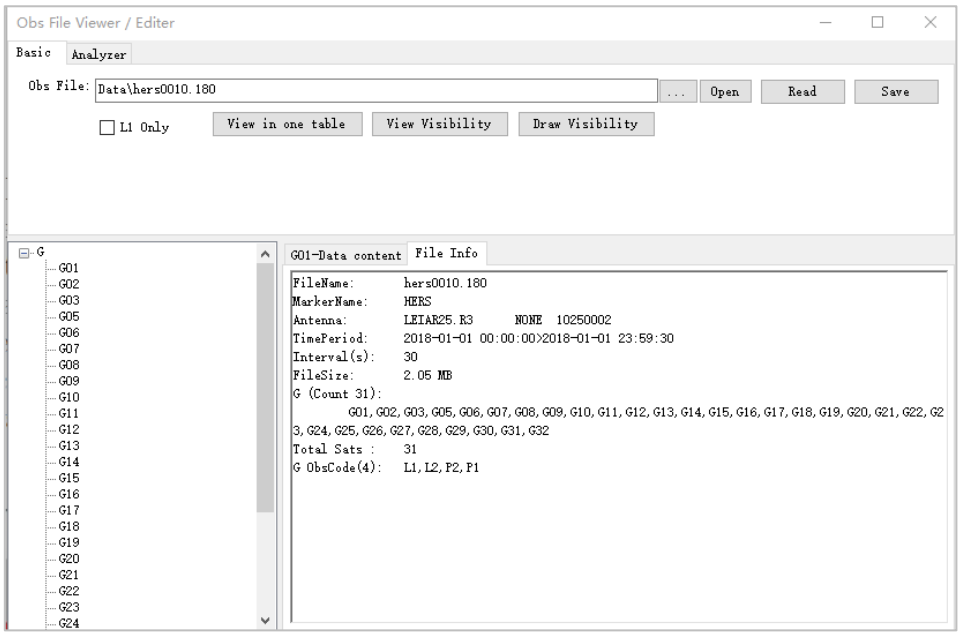


Figure 15 sample file information

Click “View in one table” to open the observation file in one table, as shown in Fig.16. The left side is the epoch, and the right side is the satellite corresponding data according the observation code. Click the “save as” button in the upper left corner of the interface to save the data as XLS file, which is very intuitive to read.

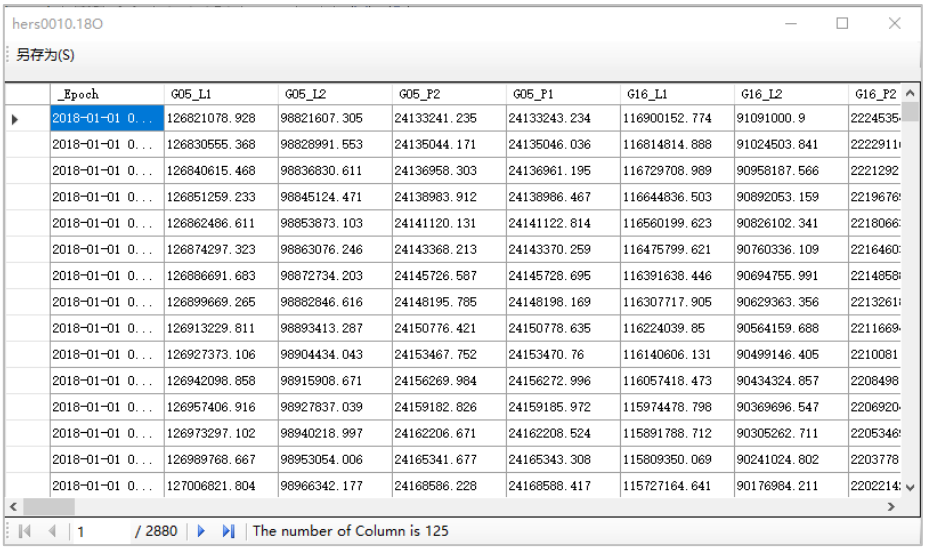


Figure 16 the sample data file is viewed as a table

Click the “View Visibility” button to view the number of arcs for each satellite, as shown in the Fig.

17. For example, the GPS satellite G01 has only one arc “2018-01-01 03:32:00>2018-01-01 10:10:00”, G02 has two observation arcs, which are “2018-01-01 09:10:30>2018-01-01 13:54:30” and “2018-01-01 19:06:00>2018-01-01 23:18:00”. Through this function, the arc length of the satellite can be quickly checked, and then the interested arc of the satellite can be selected for specific analysis.

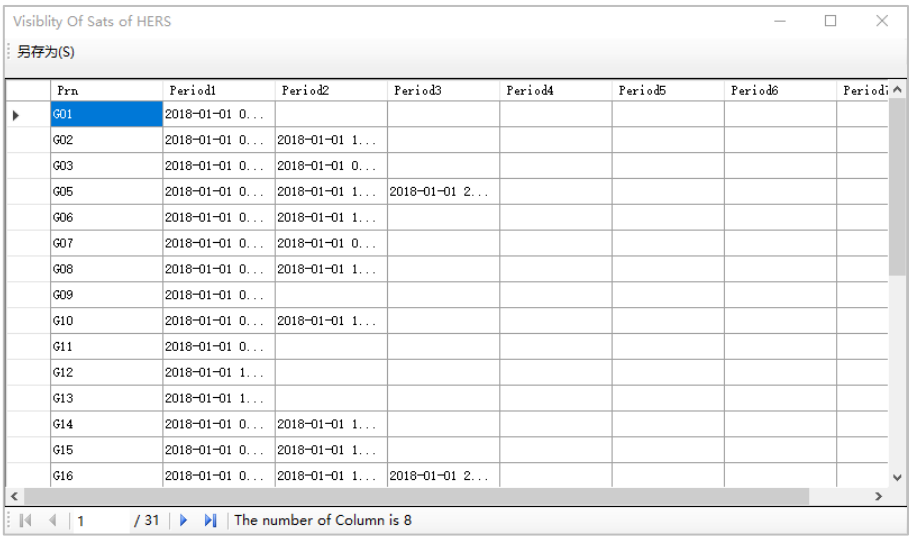


Figure 17 view the number of arcs in table mode

Click the “Draw Visibility” button to view the satellite arc of the observation file in the form of a picture, as shown in Fig.18. It is very intuitive to see the arc information of each satellite.

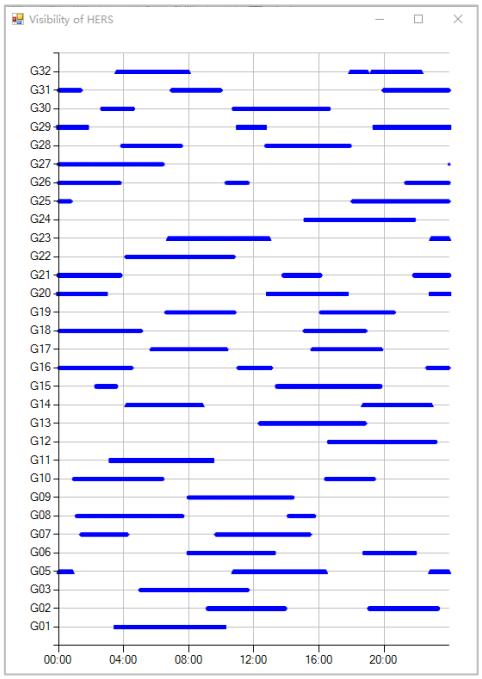


Figure 18 view the number of arcs in graph mode

Select the “Analyzer” panel to view and output cycle slip, multipath, etc. As shown in Fig.19, by entering the threshold value of LI or MW cycle slip detection algorithm and click the corresponding

cycle slip detection button, the preliminary cycle slip detection can be conducted and the results can be displayed in table form.

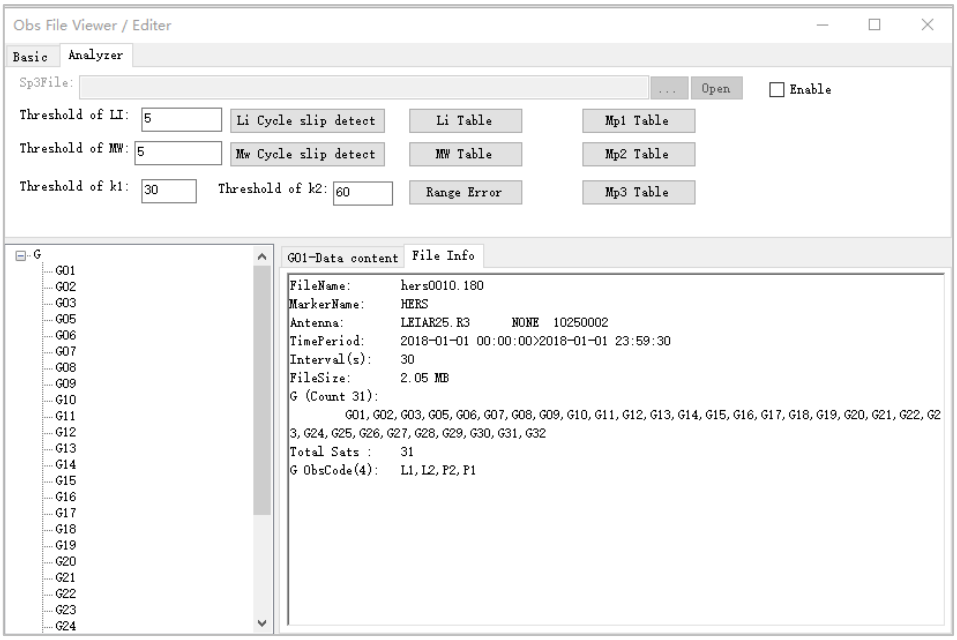


Figure 19 analyzer panel

Click the “Li Table” or “MW Table” button to display the LI or MW combined values of all satellites in table form. Here, if an ephemeris file is specified, the corresponding satellite altitude angle information is also displayed. The following Fig.20 shows the LI combination value of the sample data. The leftmost column is the observation epoch, and the right column displays the data of each satellite observed by the current epoch one by one. This table can also be saved as XLS table, which is very convenient to use other drawing software, such as Origin or Excel, to draw the graph of LI value changing with height angle of each satellite. Similarly, you can also view the MW combined value, multipath value of frequency 1, and so on. It is also possible to perform simple pseudorange coarse error detection by setting the $k_1$ and $k_2$ thresholds (units: meters) on the panel.

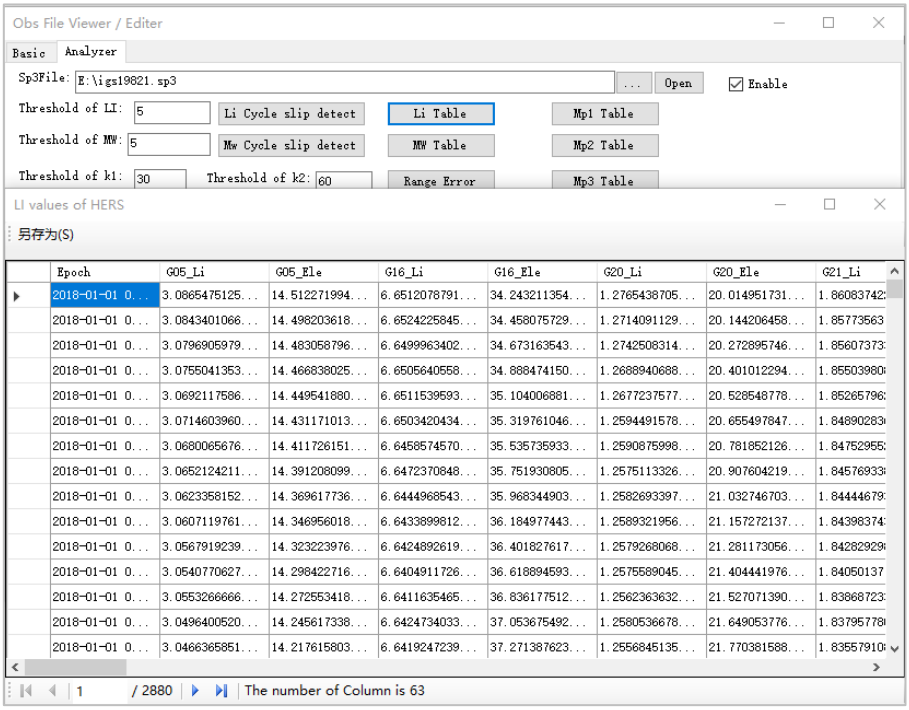


Figure 20 the LI combined value of the sample data

## 4.5 Others

The GDP main interface provides some other functions, such as “Extractor” button can be quickly read the header information of mass observation files, extract the name of each file, latitude and longitude, geodetic height, XYZ, type of receiver, antenna type and other information. If “GMT LonLatFirst” is checked in the Setting panel, the output results can be directly used to drawing the station distribution map in the GMT software.

The “Table Viewer” button is used to open and view the “.xls” file that has been output saved.

The “Log” is a log file for the running of the code, which makes it easy to debug the code and see how the program is running.

# 5 Command line program

## 5.1 command line description

Command parameters are case insensitive in GDP, while whether the path is case sensitive depends on the operating system. For example, Windows systems are case insensitive, while most Linux systems are case sensitive.

```
usage: Gdp [options]
Gdp <module name> [<module-options>]
```

The following documentation can be viewed with the help command.


Figure 21 GDP command window with help information

The following shows the detail shell command rules of GDP.

## 5.2 Common arguments

This should be in the start of the command line.

| |
|---|
| -V #Version |
| -H #Help |
| -B #Urls builder |
| -D #download |
| -F #Format |
| -S #Select |

```
-C #convert to one table text file
-E #extract site into
```

## 5.3 B:Build Urls

Build Urls by indicated condition.

Options:

```
-B #Process marker
-OutputPath MyUrls.txt
-StartTime 2020 1 1
-EndTime 2020 1 5
-ProductType Sp3
-Source igs
-IntervalSecond 86400
-UrlDirectories ftp://igs.ensg.eu/pub/igs/products
-UrlModels {UrlDirectory}/{Week}/{SourceName}{Week}{DayOfWeek}.{ProductType}.Z
```

For example:

```
-T B -OutputPath MyUrls.txt -StartTime 2020 1 1 -EndTime 2020 1 5 -ProductType Sp3 -Source igs -IntervalSecond 86400 -UrlDirectories ftp://igs.ensg.eu/pub/igs/products -UrlModels {UrlDirectory}/{Week}/{SourceName}{Week}{DayOfWeek}.{ProductType}.Z
```

## 5.4 D:Download

Download urls in text line file from ftp server or web sites.

Options:

```
-D  #Process marker
-DownloadDirectory ./Download
-UrlTextPath BuildUrl.txt
-IsOverWrite False
For example:
-D -DownloadDirectory ./Download -UrlTextPath MyUrls.txt -IsOverWrite False
```

## 5.5 F:Convert/Format

Convert or format observation files that new files will be built.

Options:

```
-F  #Process marker
-InputPath ./Data/hers0010.18O ./Data/Sample/
-OutputDirectory ./Data/Formated/
-OutVersion 3.02
-Interval 30
-SatelliteType G R C E
-StartTime 2018 1 1 0 0 0
-EndTime 2018 1 1 12 0 0
```

For example:

```
-F -InputPath ./Data/hers0010.18O -OutputDirectory ./Data/Formated/ -OutVersion 3.02 -Interval 30 -SatelliteType G R C E -StartTime 2018 1 1 0 0 0 -EndTime 2018 1 1 12 0 0
```

## 5.6 S:Select

Select observation files to indicated directory, and do not change data content.

Options:

```
-S  #Processing maker
-InputPath ./Data/hers0010.18O ./Data/Sample/
-OutputDirectory ./Data/Selected/
-IncludeSiteNames hers hert
-ExcludeSiteNames algo
-SatelliteType G R C E
-StartTime 2018 1 1 0 0 0
-EndTime 2018 1 1 12 0 0
-ObsCodes L1C L2W
-MinFileSizeMB 1.2
-MinEpochCount 1000
-MinFrequencyCount 2
```

For example:

```
-S -InputPath ./Data/hers0010.18O ./Data/Sample/ -OutputDirectory ./Data/Selected/ -IncludeSiteNames hers hert -ExcludeSiteNames algo -SatelliteType G R C E -StartTime 2018 1 1 0 0 0 -EndTime 2018 1 1 12 0 0 -ObsCodes L1C L2W -MinFileSizeMB 1.2 -MinEpochCount 1000 -MinFrequencyCount 2
```

## 5.7 C:Convert to one table

Convert observation files to one table text file.

Options:

| |
|---|
| -C #Processing maker |
| -InputPath ./Data/hers0010.18O ./Data/Sample/ |
| -OutputDirectory ./Data/OneTable/ |
| For example: |
| -C -InputPath ./Data/hers0010.18O ./Data/Sample/ -OutputDirectory ./Data/OneTable/ |

## 5.8 E:extract site into

Extract observation files info to one table text file.

Options:

| |
|---|
| -E #Processing maker |
| -InputPath ./Data/hers0010.18O ./Data/Sample/ |
| -OutputDirectory ./Data/OneTable/ |
| For example: |
| -E -InputPath ./Data/hers0010.18O ./Data/Sample/ -OutputDirectory ./Data/Temp/ |

# 6 Declaration of further development of source code

GDP was developed in Visual Studio 2019 environment using C# programming language. Free IDE and developer tool Visual Studio Community are recommended. If you are developing under non-windows, you can use Visual Studio Code. The code of GDP software project is shown in Fig.22.


Figure 22 GDP source code list

After Visual Studio 2019 is installed, double-click “GDP.sln” to open the project code on the VS platform, as shown in Fig.23. It contains three projects, among which Gdp.Standard is the core business logic project, including RINEX, sp3 and other files reading, matrix operation, etc.

Gdp.Winform is the form project, and GdpShell is the command line project.


Figure 23 GDP solution explorer

# 7 References

[1]. IGS Formats. https://kb.igs.org/hc/en-us/articles/201096516-IGS-Formats, 2020-04-08.

[2]. Kouba, J., A guide to using International GNSS Service (IGS) products, http://kb.igs.org/hc/en-us/articles/201271873-A-Guide-to-Usingthe-IGS-Products, 2015.