**Matteo Merlo**

**ICARUS\_POLITO – DOCUMENTATION**

The following documentation shows how the process of calculating the best path given two points that the plane must travel, analyzing a cloudy sky.

**Sky mapping**

For the weather we use the mapping provided by the American NOAA service. It provides the data in grib2 file format (file.grb2). Xplane has its own weather plugin (obviously more precise than the native one of the program), which uses the same data: noaawheather ( <https://x-plane.joanpc.com/plugins/xpgfs-noaa-weather> ), to install it you should download the zip file and manually copy the files to the folder. Here is a tutorial how to install it (<https://www.youtube.com/watch?v=8Nv_fM4RtaQ>)

The grib files are binary files and you have to "unpack" them (degrib, like unzipping file) and save the data you are interested in a .csv or ASCII file.

To view grib files use the **XyGrib** GUI (<https://opengribs.org/en/>).

*Installation:*

* *LINUX:*

Look at Ubuntu part: <https://github.com/opengribs/XyGrib/blob/master/INSTALL.md>

* Windows (both)

Download installer

**Download GRIB file**

The grib files are generated by NOAA and downloaded through the website: <https://nomads.ncep.noaa.gov/> (This is the only custom way that I found)

Select Gfs 0.25 degree filter: https://nomads.ncep.noaa.gov/cgi-bin/filter\_gfs\_0p25.pl

So select the most recent date of the last update and then select the hours. The files are stored for the previous ten days and then updated every 6 hours: from midnight (00) until 18 PM in the evening (18).

On the page, select one of the next 120 forecast hours from the meteorological sampling date.

***WARNING***: do not select the .anl file, but only those ending with f000 - f0120

Now select the barometric pressure and altitude, then select the parameter corresponding to what you want to view. Parameter table: <https://www.nco.ncep.noaa.gov/pmb/docs/on388/table2.html>.

Select the region by ticking the box and then download. If you want to view the corresponding URL, check the box next to it.

Now the file is downloaded (without extension), but you need to add the .grb2 extension to the bottom. Now the file is displayed as grb2. To view the file you can open it in xygrib.

**Degrib**

Here there is the official NOAA page:

<https://www.weather.gov/mdl/degrib_tutorial>

<https://www.weather.gov/mdl/degrib_home>

In order to work on grib binary files, we use a degrib program that decompresses the file in ASCII.

**Download Degrib** program: <https://www.weather.gov/mdl/degrib_install>

Two executables are created (in the installation folder) from the installation:

* *degrib.exe*
* *tkdegrib.exe*

The first is the program that run from the command line, the second is the same program with the user interface (GUI). The degrib.exe file must be in the same folder of compiled code.

**Convert a message to ASCII Comma Separated file**:

You can use degrib to convert a given message to a .csv file, which is a comma delimited ASCII file that contains a x, y, lat, lon, value line for each cell in the grid. This file can be imported into Excel (limited to the first 65,536 lines), or read by other files to populate a SQL database.

* **In the GUI tkdegrib:**

Click on the "GIS" tab, and browse for your file. Double click on it in the top half, and it should fill out the inventory part in the bottom half. Select the message that you want in the bottom half, and choose the output name (or press "Recommend" to have tkdegrib "recommend" one. Next choose "CSV" for "File Type", and press the "Generate .csv file" button at the bottom.

* **From the command line:**

/ degrib / bin / degrib "GRIB file" -C -msg 1 -Csv

/ degrib / bin / degrib "GRIB file" -C -msg all -nMet -Csv

where "GRIB file" is replaced with the name of your GRIB file.

This should read the file, and extract the requested messages. The -Csv option tells it to create the .csv file. See the "degrib Man Page" (particularly the "CONVERT OPTIONS" section) for more details.

Degrib tutorial: <https://www.weather.gov/mdl/degrib_tutorial>

Degrib manual shell commands: <https://www.weather.gov/mdl/degrib_manpage_degrib>

The NOAA site freely provides data with maximum resolution 0.25 degrees which correspond to squares of 30 km per side. Higher resolution is provided by French system via AROME 0.025 degree: <https://donneespubliques.meteofrance.fr/?fond=produit&id_produit=131&id_rubrique=51>

Other resolutions here is a list: <https://opengribs.org/en/gribs>

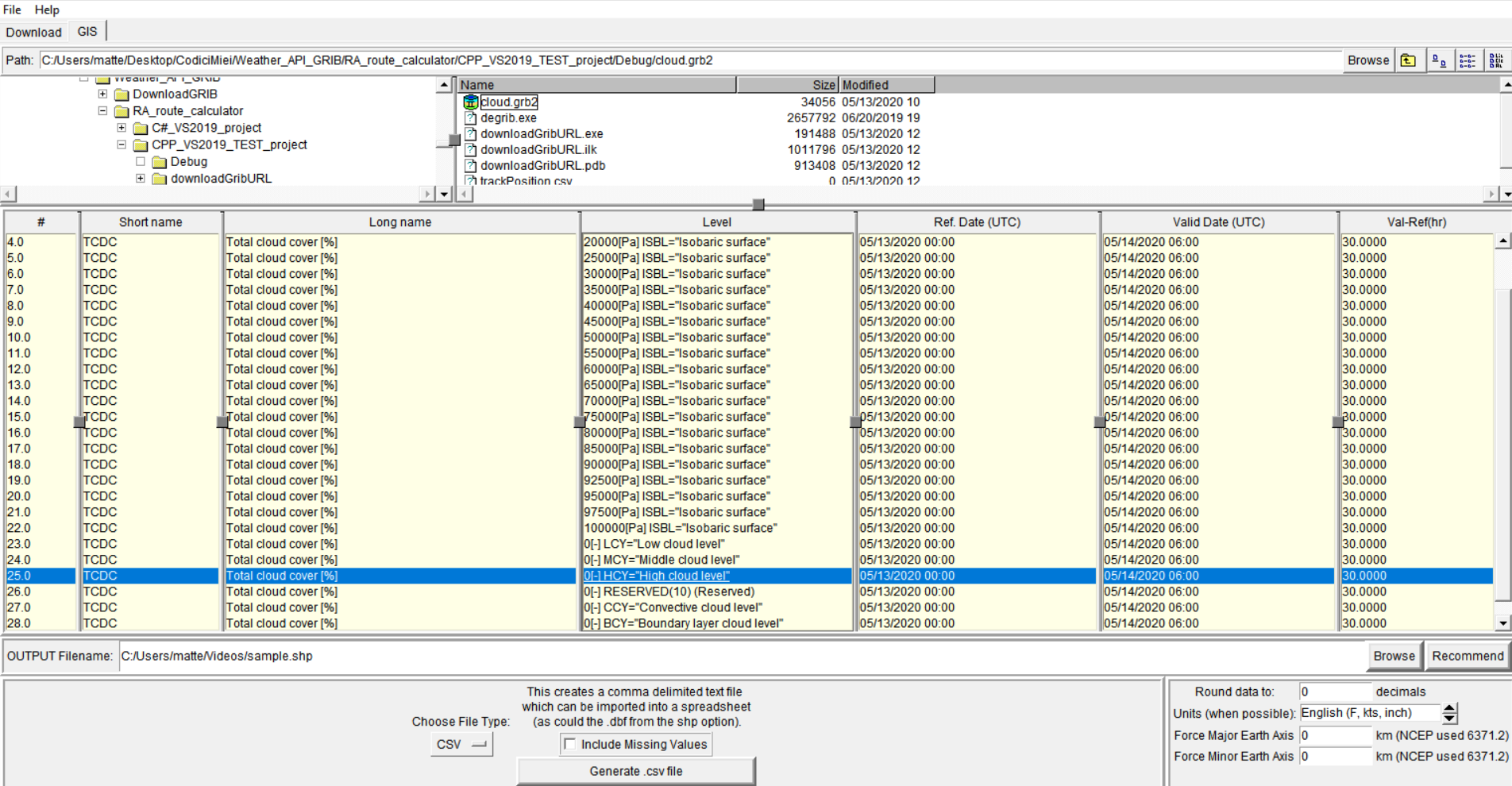
**TKDEGRIB**

It is the GUI of degrib.exe

Passage to open GRB2 file already downloaded:

* Click on GIS at the top left, next to download
* Select the file by browsing inside the folders
* Select the desired parameter and values that you want to extract
* Click on the bottom left to convert it to the desired format (CSV)
* Click on generate file

# Parameter to extract File explorer



Convert in CSV Generate file Change unit measurement

**NOTA**: the parameter number is to be passed as an argument to degrib.exe. Use tkdegrib to decide which parameter to extract.

**NOTA**: The recommended parameter to be extracted is number 26, the one with a RESERVED level, contains information on the entirety of the atmosphere, or extract 25 which contains only the information of the upper layers, useful when flying at high altitude.

Files can be converted to SHP files and opened in Google chrome.

Guide to open SHP File in google earth (Not tried yet): <https://www.google.com/earth/outreach/learn/importing-geographic-information-systems-gis-data-in-google-earth/>

<https://sats.nws.noaa.gov/~WGDS/C_sourceCode/degrib/docs/NDFD_GRIB2Decoder.html>

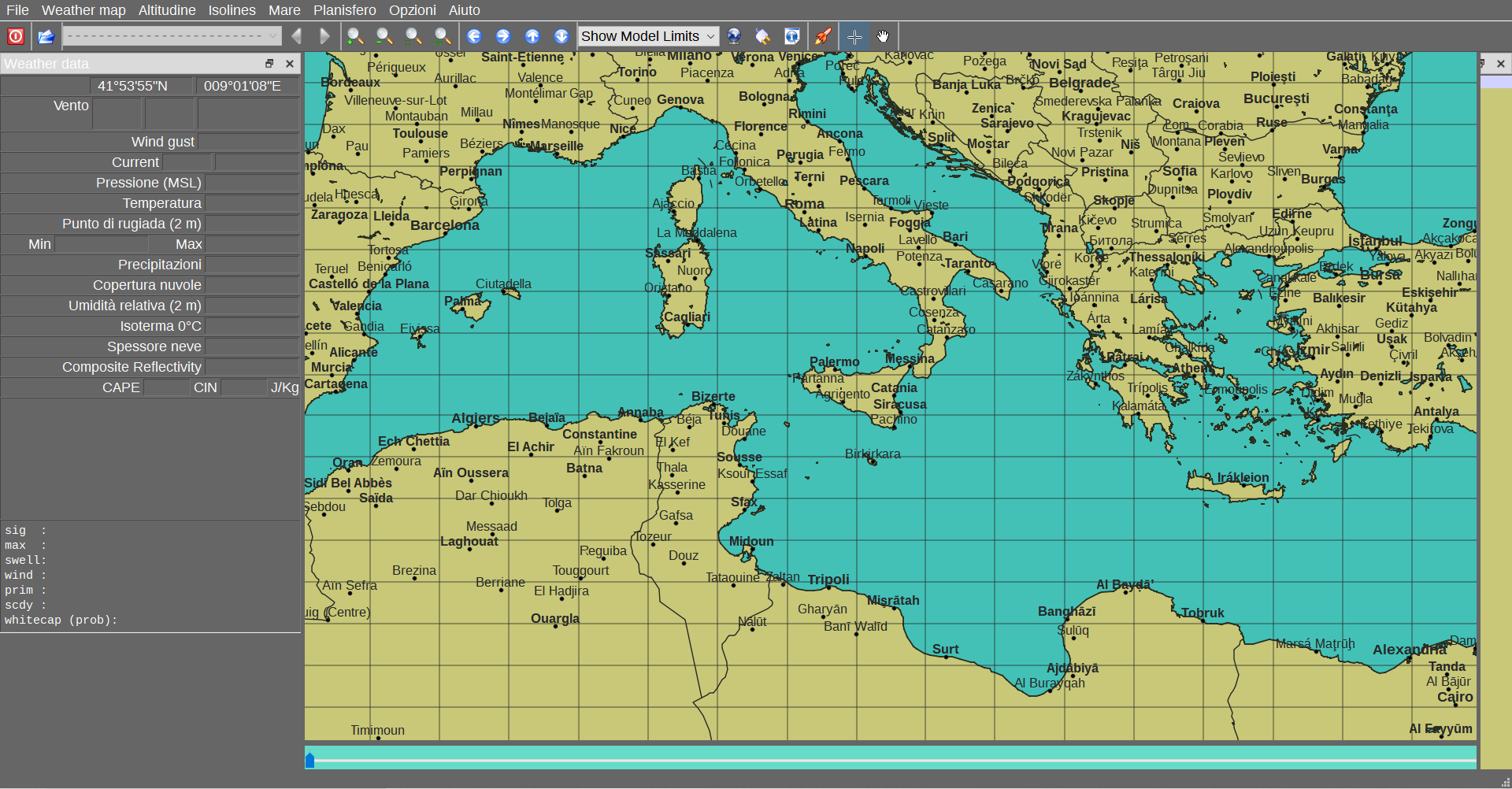
**XYGRIB**

Xygrib is a graphical interface for using grib files.

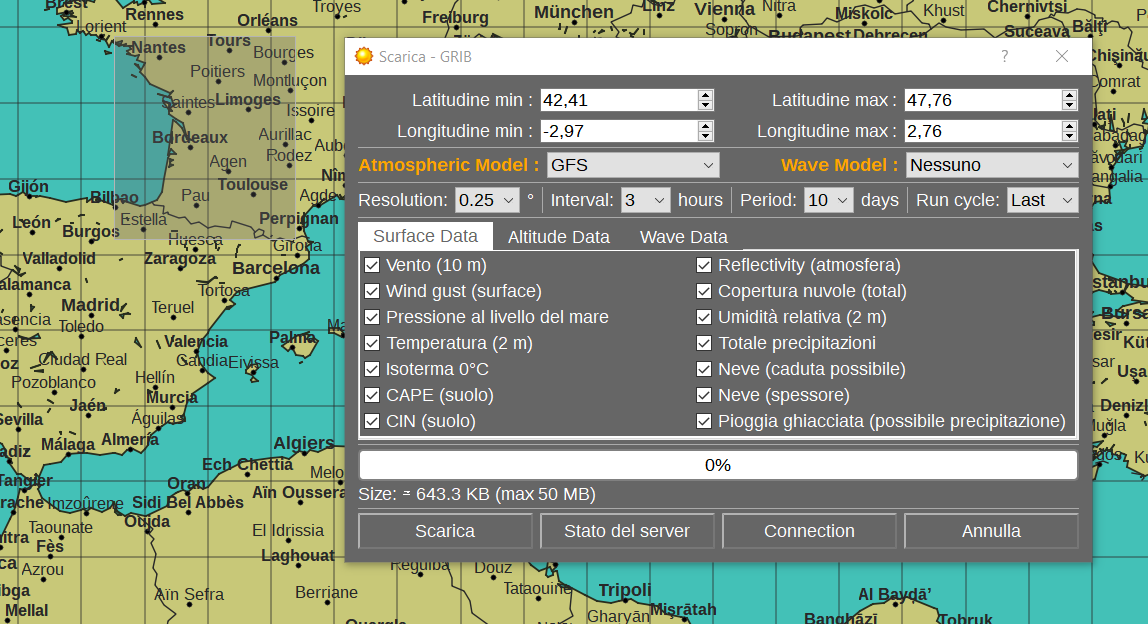
On the map you can select through the Weather map menu what to see between wind, precipitation, cloud cover… and see the situation at various altitudes. To move on the map use the arrows at the top. To download a new map, select the desired area and then download by clicking on the globe icon

Menu select and change view Download new weather map after have select bound





Leave GFS as model and then select the time interval and period, check the interesting data to be displayed.



**CODES**

There are various code:

* A program executable in CPP with command line parameters (in downloadGrib folder)
* An identical program in CPP of TEST with dijkstra (in RA route computation folder)
* A program with a GUI in C # with dijkstra (in RA route computation folder)

A program with the openweather.org API has also been created (see other documentation)

**CPP (COMMAND LINE PROGRAM)**

In the program they are passed as parameters in order

0) Program name and path folder (obviously)

1) Hours from forecast (max 120)

2) Maximum north latitude area to explore

3) South latitude minimum area to explore

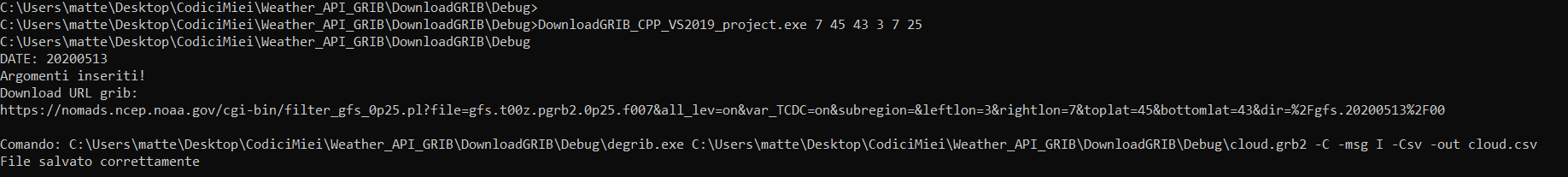
4) West longitude area to explore

5) East longitude area to explore

6) Parameter to extract (see on tkdegrib.exe which value to choose)

7) Where to save the file (optional - NOT TESTED)

In the example below, the weather forecast is requested in 7 hours with an area bounded to north 45, south 43, west 3 and east 7 degrees, taking only the parameter 25 (high level cloud). (see tkdegrib for the parameter)



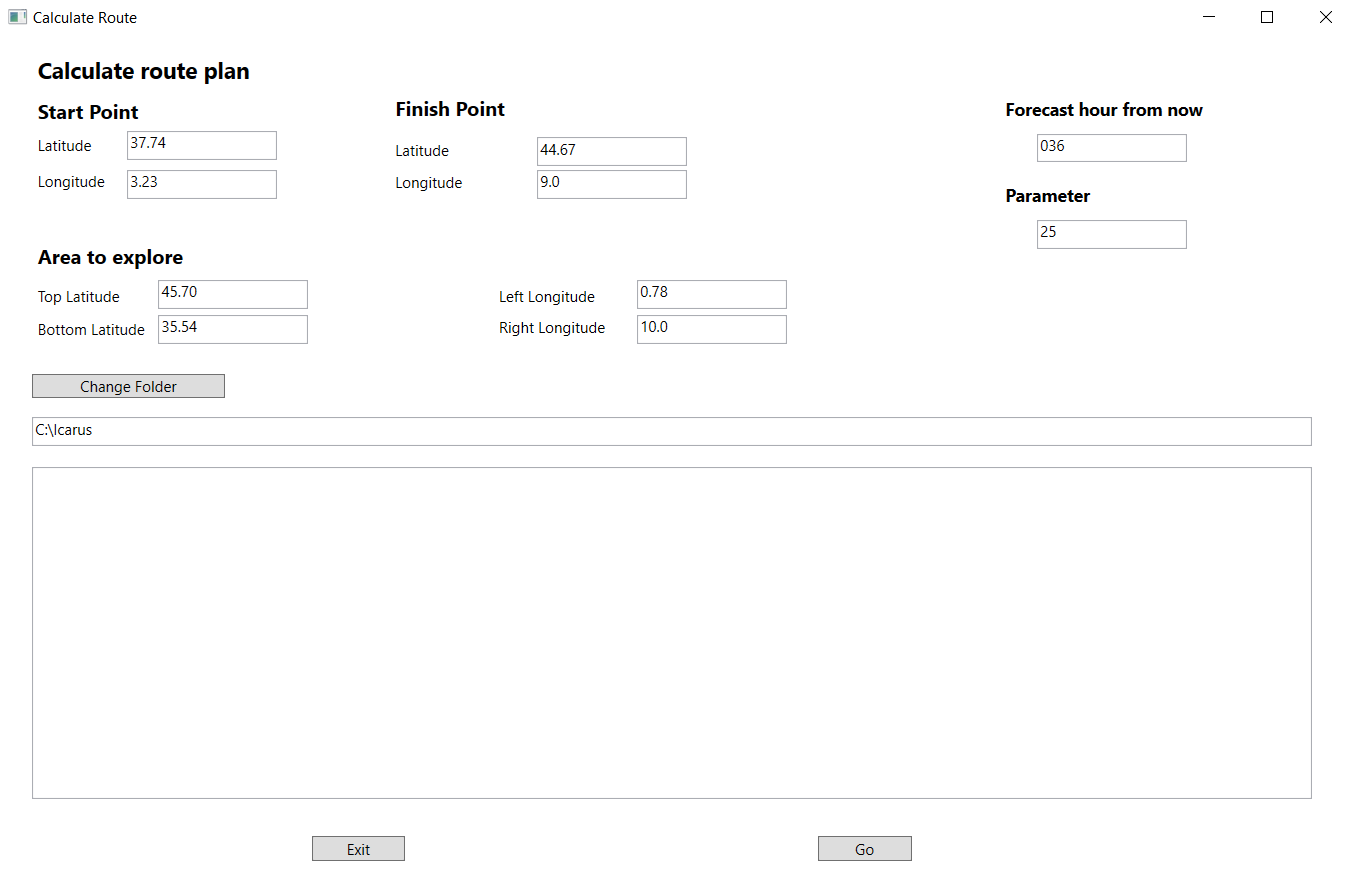
**CPP (con dijkstra)**

Then there is another test program with the dijkstra algorithm working. In it you must immediately set all the global variables at the beginning main class.

WARNING: The program is affected by a problem that causes it to crash by exiting prematurely (problem not solved on the allocation of the pathLenght variable)

**C# (GUI)**

Interface :

IMPORTANT: insert here always 3 digit Select the folder where the degrib.exe file is located Parameter to extract (see tkdegrib)

Main function is in class MainWindows.xaml -> MainWindows.xaml.cs

To view the path of the points on google earth:

* Import and select the file with all points of the trackPosition.csv path
* Select the comma as the separator
* Then select latitude and longitude in floating point
* Check the imported file under temporary file on the left drop-down menu

**WIKI**

**FILE GRIB**

The files have a .grib2 extension ( <https://en.wikipedia.org/wiki/GRIB> ). In short they are BINARY files containing information on the weather in general (humidity, rain, temperatures, wind, wind direction, etc ...).

WARNING: the packages need the related Json that I have not found a unique way. Packets are separated by separator code 7777 (32 bit 1) -> similar to TCP packets. **Beware of grib and grib2, they are slightly different and incompatible files.**

Here is command line training, if it can be useful:

<https://www.ecmwf.int/assets/elearning/eccodes/eccodes1/story_html5.html>

Here is a wiki that can be useful, with other related articles:

<https://confluence.ecmwf.int/display/CKB/How+to+read+or+decode+a+GRIB+file>

Source code XyGrib: <https://github.com/opengribs/XyGrib>

Source code grib manager: <https://github.com/mariokonrad/grib>

**ALGORITMO PATHFINDING**

To find the shortest path between two points I used a weighted grid pathfinding algorithm (type algorithm: weighted grid pathfinding). The algorithm most appropriate and similar to the needs of the program is the field D\*. TO BE COMPLETED ….

Currently in use Dijkstra ….

WARNING: Windows home edition can allocate maximum 4 GB of single variable (matrix in this case).

INT = 2Byte, allocated 8. -> 4GB/8 = 256MB -> SQRT(256 MB) = 16000 int, this is the dimension of adjacency matrix

So if we make a matrix of integer the maximum size is 16000 cell/vertex, example 160x100. So if the have 0.25 degree resolution, we can analyze only 40x25 degree area on world map.

**REFERENCE**

Documentation on file GRIB: <http://www.wmo.int/pages/prog/www/DPS/FM92-GRIB2-11-2003.pdf>

Parameter e abbreviation GRIB2: <https://www.nco.ncep.noaa.gov/pmb/docs/on388/table2.html>