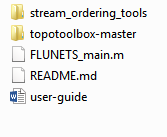
## User Guide to FLUNETS

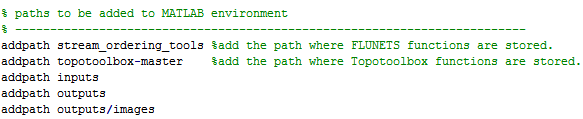
FLUNETS is a MATLAB-based tool built up in the form of 6 MATLAB functions. The aim of this tool is to provide a continuous channel network. The sorting hierarchy can be Gravelius/Hack ordering system or Horton hierarchy. Both systems offer a drainage network formed by continuous rivers. FLUNETS uses some functions from TopoToolbox (DOI: 10.5194/esurf-2-1-2014) and requires MATLAB 2011b version and the Image Processing Toolbox for some of its functions. TopoToolbox terrain analysis toolset is available for download on Wolfgang Schwanghart website, where you can find a direct download link from Github (<https://github.com/wschwanghart/topotoolbox>).

Before starting, it is necessary to set the paths where the FLUNETS functions and the TopoToolbox are located. These directories should be set at the start of the FLUNETS\_main.m file.

Something like:



And set the paths inside FLUNETS\_main.m:



**Important:**  If you get the following error message “Warning: Method directories not allowed in MATLAB path:”, delete the topotoolbox path from MATLAB paths (if you have previously added), rename the folder, and add the path again to MATLAB. You can do this in MATLAB command window, e.g.:

rmpath('topotoolbox-master')

rename the folder from ‘topotoolbox-master’ to ‘topotoolbox’

addpath ‘topotoolbox’

FLUNETS runs from the FLUNETS\_main.m file. In this file, the user sets the values of paths, the mandatory and optional parameters value. The **mandatory** ones are a total of 3 parameters. This set should be fulfilled previous to processing the tool, otherwise, a MATLAB error message will show up. These parameters are the name of the DEM in ASCII format with the extension – if the DEM file is not stored in the same path where the FLUNET\_mai.m file is, you should also specify the whole path and DEM name with extension-; the ordering hierarchy by which the rivers of the network will be sorted; and finally the hierarchy attribute, which may be upstream accumulation or upstream distance. This last parameter identifies the parent stream over the child stream when they join in a confluence.

For example:

% mandatory parameters

dem\_namefile = 'arlanza.asc';

sorting\_type = 'horton'; hierarchy\_attribute = 'distance';

In contrast, **optional parameters** can be left empty and a default value will be assigned to each of them. These are the following:

max\_trib\_order: It is the ultimate tributary order (the '-ith' order up to which the network will be sorted). If the value is set to 1, only the main stream and first-order tributaries will be ordered. If the value is set to 2, the master channel river and first and second-order tributaries will be sorted, and so on. The value should be an integer.

min\_drainage\_area: Is the minimum area in m2 that must have a tributary at its pour point to be sorted. The area is obtained from multiplying the accumulation at the pour point by the spatial resolution of the DEM, to the second power. If the computed value is equal or higher than the value set in this parameter, the channel is sorted. If no value is set, the default value will be 10^-4 of the total DEM watershed area. Can be integer, double or float.

maxbase: It is the highest elevation range to consider an outlet. For example, if the value is set to 20, only those outlets located at that certain height or below will be considered. On the other hand, if the parameter is left empty, all the outlets will be taken into account. Can be integer, double or float.

internal\_matrices: String type. If the parameter is set to ‘yes’, a set of flow-related matrices (flow direction, flow accumulation, flow distance (when choosing distance as hierarchy attribute) and Strahler (for Horton)) will be given alongside the ASCII and CSV as outputs. In contrast, if left empty, flow-related matrices will not be provided.

junctions\_points: String type. This parameter provides an ASCII file containing the pour points of all the sorted rivers. If willing to get the pour point matrix, set junctions\_points = ‘yes’.

output\_name: Name given to the ASCII and CSV output files. Should be a string of characters written without extension (‘.asc’ or ‘.txt’ at the end), either spaces or strange characters. If left empty, the name will be built by joining the DEM name with the parameter values, separated by an underscore.

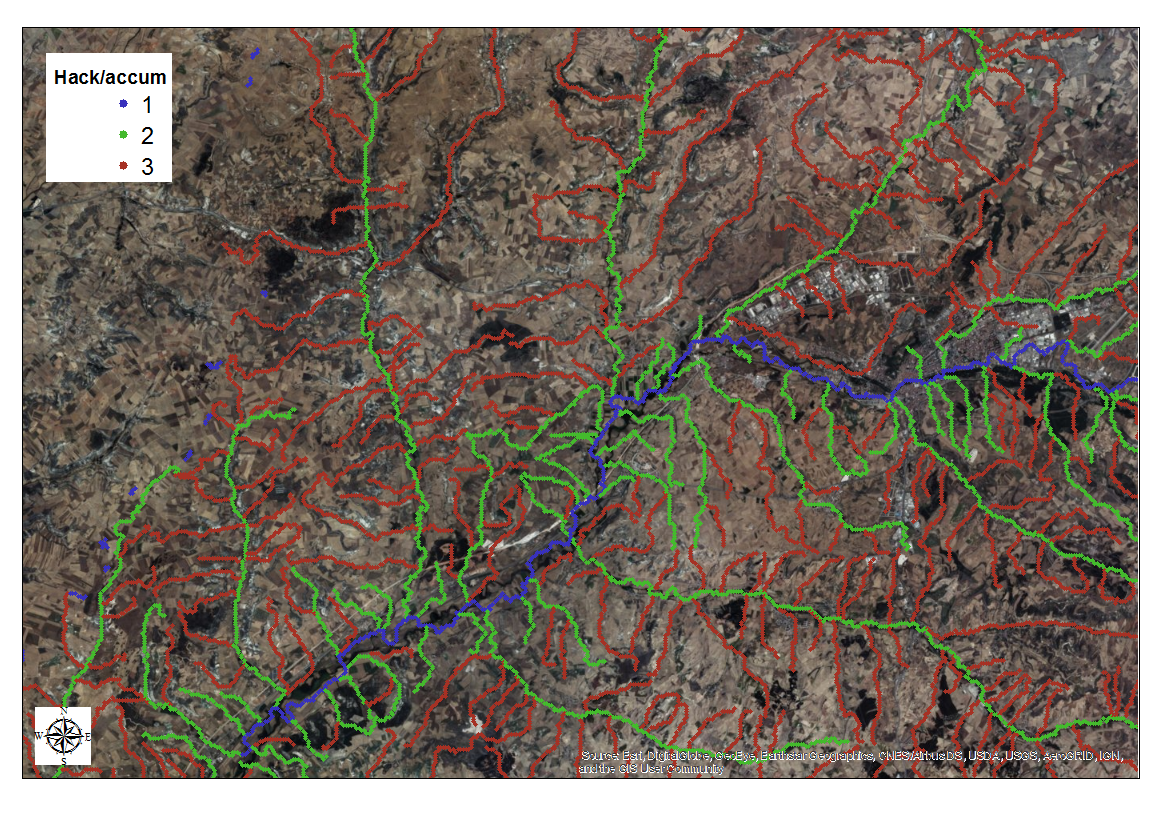
Output files:

The output files are given in ASCII fomat and in a CSV. Both files can be drawn directly in any GIS software. The ASCII has the ordered value for each pixel, whereas the CSV file contains multiple information about the obtained network. In the CSV, there is a register (row) for each pixel of the channel network. The meaning of each of the columns is as follows:

* Field1: is the x coordinate (in a projected system).
* Field2: is the y coordinate (in a projected system).
* Field3: is the elevation.
* Field4: is the river value (Hack or Horton order).
* Field5: is the accumulation value upstream direction.
* Field6: is the drainage area in square meters.
* Field7: is the id number of each river (unique).
* Field8: is the distance upstream direction.
* Field9: the pour points location.
* Field10: the outlets location.

**Important tip!** To convert an ASCII channel network to vector format, in most GIS environments you will need the flow direction matrix. This direction matrix is the one computed internally to extract the channel network. Therefore, you need to use the parameter 'internal\_matrices' = ‘yes’ to obtain the flow direction matrix as output.

## Outputs

**Figure 1: CSV loaded in ArcMap**

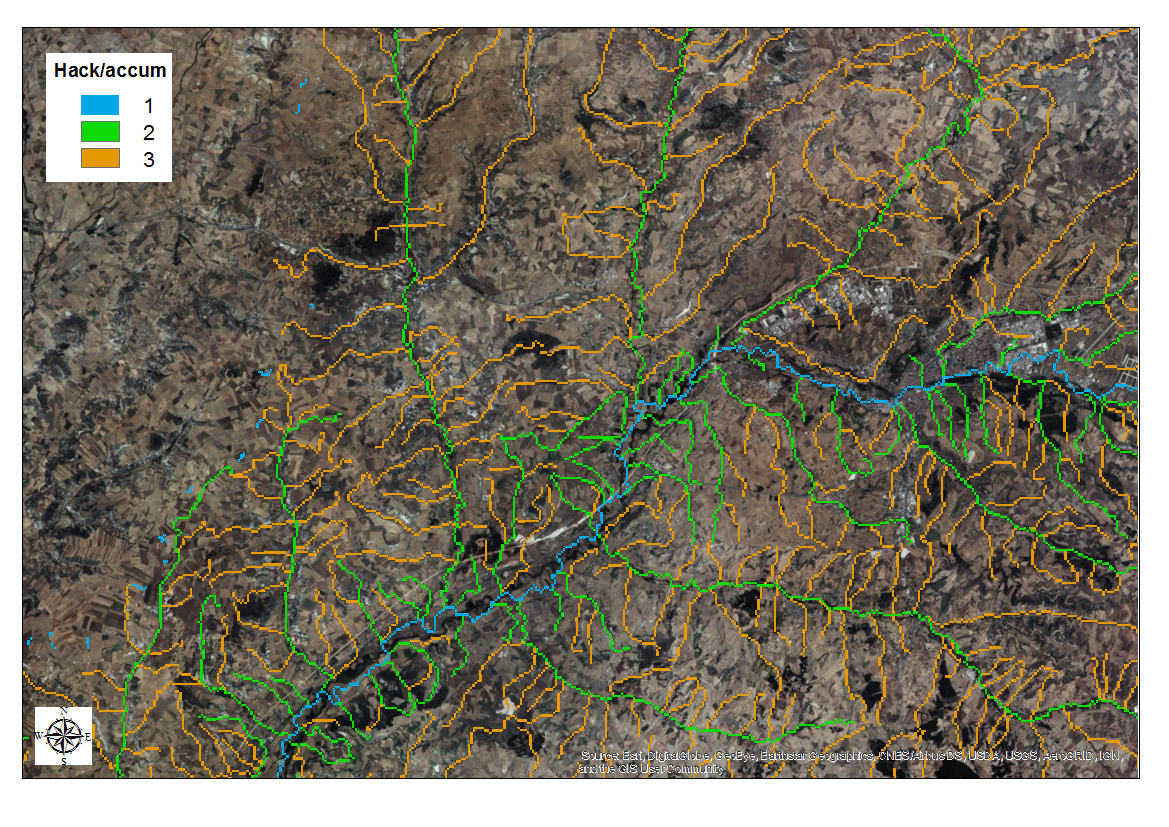


Figure 2 ASCII loaded in ArcMap