Authors:
C. Kermabon (ODE/LOPS/TOIS)
P. Lherminier (ODE/LOPS/OH)
P. Le Bot (ODE/LOPS/OH)

24 Nov., 2023

Ifremer

CASCADE

Chaine **A**utomatisée de **S**uivi des

Courantomètres **A**coustiques

Doppler **E**mbarqués

CASCADE V7.2: Software for processing, qualifying and visualizing SADCP data

User's guide

LOPS

SOMMAIRE

1 General Introduction

CASCADE is a MATLAB software developed at the Laboratory for Ocean Physics and Satellite remote sensing (LOPS), designed to qualify and analyze current measurements from onboard acoustic profilers (VM-ADCP: Vessel Mounted Acoustic Doppler Current Profiler or SADCP, SADCP for Ship ADCP). It was initiated in 1998 (Kermabon and Gaillard, 2001) for research projects, and then evolved towards a more operational use via a relatively intuitive graphical interface and more powerful visualizations. However, the processing steps are very similar to those carried out in the CODAS software, developed for the same purpose and at the same time (Firing et al., 1995). CASCADE is currently used by SISMER to process all the data from the French oceanographic fleet, as well as by research groups from Ifremer, IRD and CNRS.

On most ships, in particular French ships, VM-ADCPs from Teledyne RD-Instruments (RDI) are installed. Data acquisition is then carried out by the shipbuilder's VmDas software. This generates binary output files (STA and LTA files) containing, in particular, the ship speeds and current speeds relative to the ship along the ship's trajectory.

The CASCADE software allows you to qualify and exploit the data in these STA and LTA files. When these files have been acquired with the same configuration, it is advisable to concatenate them into a single binary file, because the accuracy of the estimation of possible alignment and amplitude errors depends on the number of data. The binary file is then converted in CASCADE into a NetCDF cruise file in OceanSite format. This file will become the working file, which will be completed during processing with auxiliary data (bathymetry, tide) and quality flags. The current data will eventually be corrected by rotating the ADCP marker (trim, alignment) and amplitude correction, the values of which will be suggested by calculations made in the software.

Most of the processing parameters have standard mostly satisfactory values in most cases, allowing inexperienced users to take the software into their own hands. The cruise file is a standard product that can be used by a large community. It is the starting point for scientific or operational analyses.

Once the cruise file has been qualified, a second set of operations allows different phases in the cruise to be identified by defining sections and stations. For the sections, data are spatially averaged, while for the stations, they are temporally averaged. CASCADE offers the possibility to visualize these sections and stations with current vectors on a geographical map averaged over a slice of water. These can also be displayed as contouring or image with depth as Y-axis and latitude, longitude or distance on X-axis.

Initially, CASCADE consisted of 2 parts:

- **A PROCESSING part**: This part allows to generate absolute current velocity data from the raw ADCP data, in single-ping beam coordinates, of the old VM-ADCP. Currently, this part is only necessary for ADCP data acquired with the RDI TRANSECT acquisition software, which does not allow to benefit in real time from the best possible external data (heading, roll, pitch, ...etc). As the VmDas RDI acquisition software does not have this handicap and is generalised on ships, the PROCESSING part is only rarely used today. The current version is version 5.6 and there are no plans to upgrade it.
- **An OPERATION/ANALYSIS part**: This part provides the tools necessary to produce validated and qualified ADCP data from files issued from the VmDas acquisition software or generated by CASCADE PROCESSING.

This document is associated with the **OPERATION/ANALYSIS part of** CASCADE, version 7.2.

2 What you need

2.1 The different files

We present below the different files used and/or generated by the CASCADE software.

2.1.1 Files from VmDAS used in CASCADE:

2.1.1.1 STA and LTA files

STA (Short Term Averaged) and LTA (Long Time Averaged) files are generated by the VmDas acquisition software. They are binary files containing the current velocities relative to the vessel, averaged over a time interval defined by the user during data acquisition. At LOPS, STA files correspond to files averaged over 2 minutes while LTA files correspond to files averaged over 5 minutes. The user decides whether he wants to work on LTA files or STA files. In LOPS, we use the STA files, which have a better temporal resolution.

2.1.1.2 LOG files

Although not compulsory, they are very useful to check the configuration of the instruments for all the files mentioned above, that are summarized in a csv file by a tool of CASCADE.

2.1.2 Files created by the user

2.1.2.1 Station list ascii file

Station list files are ASCII files that define a user-created list of stations. Each station is defined by a line indicating the station number and the associated station start and end dates (dd/mm/yyyy hh:mm:ss). In order for CASCADE to automatically proposes these files in the interface, the nomenclature of these files must end with _sta.list.

2.1.2.2 Section list ascii file

Section list files are ASCII files that define a user-created list of sections. Each section is defined by a line indicating the section number and the associated section start and end dates (dd/mm/yyyy hh:mm:ss). In order for CASCADE to automatically proposes these files in the interface, the nomenclature of these files must end with _sec.list.

2.1.2.3 External heading file (if necessary)

The external heading file is a NetCDF file containing the ship's heading at different dates of the cruise. This file is to be created by the user in case the heading taken into account in real time during data acquisition, via the VmDas software, is incorrect. This file is then used to correct the current velocities of a vessel heading error. The structure of this file is presented in appendix 5.8.

2.1.2.4 Trinav file (if necessary)

The Trinav file is a NetCDF file containing the ship's speeds at different dates of the cruise. It is to be created by the user when the ship's speed taken into account during data acquisition, via the VmDas software, is incorrect. This file is then used to correct the current velocities for an error in the ship's speed. The structure of this file is presented in appendix 5.9.

2.1.3 Files generated by CASCADE

2.1.3.1 Cruise files (netCDF)

Cruise files are NetCDF files in Oceansite format. They include the current speeds (absolute and relative speeds) as well as the ship's speeds. A first cruise file is directly resulting from the conversion of the STA/LTA file concatenation from binary to NetCDF format. As analysis progresses, auxiliary data (tide, bathymetry) and data quality indicators can be added to this file.

From this file, other cruise files can be generated by, for example, filtering the data or misaligning the ADCP with respect to the ship's axis.

2.1.3.2 Working file

The working file is a cruise file that the user modifies via CASCADE. From this file, the user can view the data, add auxiliary data and quality indicators, and generate other cruise files including, for example, current speeds corrected for any misalignment of the ADCP with the ship's axis or containing filtered current speeds.

2.1.3.3 Cascade configuration file (conf_exploit.mat)

The configuration file is a Matlab file. It is automatically generated by the CASCADE software. Initially, it includes default values. As the analysis steps are completed, these are replaced by the information entered by the user and stored for a future analysis session.

2.1.3.4 Station files

Station files are NetCDF files in OceanSite format. They are generated by CASCADE from the working file and a Station list file, averaging the current velocities for each station during X-second, where X is user-defined. From these files, the user can view the temporal evolution of the current for each station.

2.1.3.5 Section files

Section files are NetCDF files in OceanSite format. They are generated by CASCADE from the working file and a Section list file by averaging, for each section, the current velocities in X km, where X is user-defined. From these files, the user can visualize the spatial evolution of the current for each section.

2.1.4 Ancillary files

2.1.4.1 Bathymetry files

Bathymetry files are NetCDF files containing the bathymetry for a geographical area. The structure of these files is in appendix 5.7. Some bathymetry files are proposed by default, but the user can also create his own files.

2.1.4.2 Tide files

Tide files are binary files to be imported by the user from the site: http://volkov.oce.orst.edu/tides/global.html.

2.2 Minimum configuration required

2.2.1 Matlab configuration

To use the OPERATION part of CASCADE Version 7.2, you need:

- MATLAB version greater than or equal to 2008b.
- the MATLAB toolbox m_map accessible via the Internet:
  https://www.eoas.ubc.ca/~rich/map.html
  Be careful to also load the GSHHS high resolution rib lines:
  https://www.ngdc.noaa.gov/mgg/shorelines/data/gshhs/
  (*.b files) and install them in the m_map/private/ directory.
- bathymetry files. By default, the software offers 5 choices of bathymetry to be loaded separately:
  - bathy_etopo1 and bathy_etopo2: bathymetry created by the MERCATOR team with a resolution of 1 and 2 minutes
  - « Bathy6min »: bathymetry created from the etopo2 file, averaging each point with its 8 neighbours (with the 3 or 5 neighbours for the edges) then decimating every 3 points.

- « Bathy18min »: Smoothed bathymetry file at 18 minutes resolution, created as before but from Bathy6min.
- gebco: bathymetry from the GEBCO CDROM with a resolution of 1 minute.If the user wishes, he can create his own bathymetry file. This file must be in NetCDF format and contain the variables:
  - *latitude* with dimension *latitude*
  - *longitude* with dimension *longitude*
  - *z* (negative bathymetry in m) with dimension *(latitude, longitude)*
- tide files. We propose to use the tide model files generated by the inverse tidal model, based on the Topex/Poseidon satellite data from G. Egbert and L. Erofeeva, which are available on the internet (see section 2.1.9).

2.2.2 - Initial data

To execute the OPERATION part of CASCADE Version 7.2, the user must have input files. These files differ depending on the software used to acquire ADCPs.

2.2.2.1 - For ships equipped with TRANSECT

Raw data from TRANSECT (single-ping current velocity data in beam coordinates) must have been previously processed by the PROCESSING part of CASCADE Version 5.6. The processing produced a cruise file called <filename>.nc. This file contains the current velocities in geographical coordinates averaged over time. The user must convert this file to an Oceansite compliant NetCDF file as described in the OPERATION part. This conversion can be done using the conv_trait2exploit.m function provided with CASCADE.

conv_trait2exploit(filename.nc) generates the file filename_**osite**.nc compatible with the OPERATING part of CASCADE

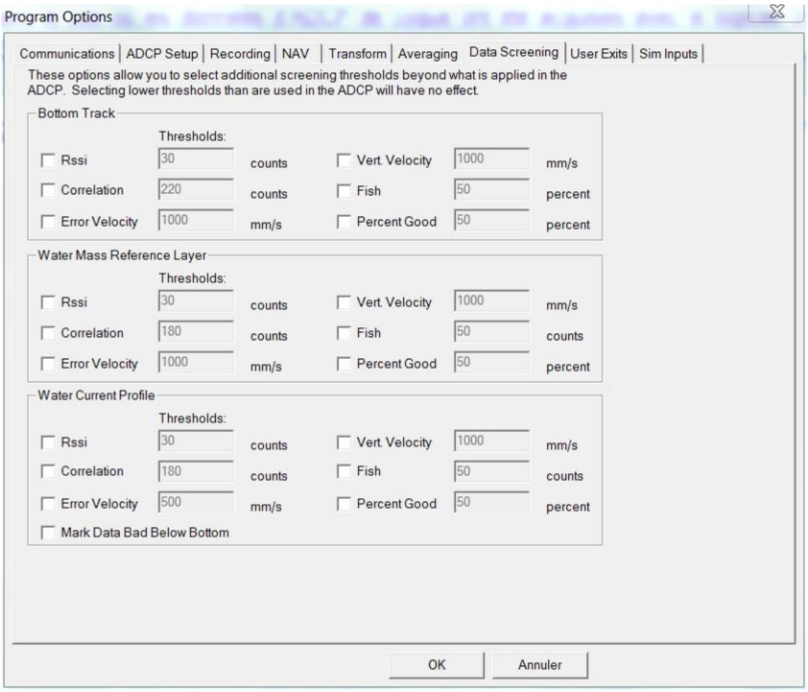
2.2.2.2 - For ships equipped with VmDas

The user must have all the *.LTA (or *.STA) files of the cruise. It is then possible to concatenate them, in chronological order (in Linux, use "cat *.STA > cruise.STA", or in Windows use "copy *.STA /B cruise.STA /B" in the command prompt). The user enters the concatenated file as input in the 1st tab of CASCADE to create an OceanSite standard NetCDF file. Although not essential for data analysis, *.LOG files are useful to know the acquisition configuration associated with each file.

In the case where the ADCP data were acquired with the VmDas software, we chose to use LTA/STA files which contain current velocity data in geographical coordinates averaged over a user-defined time period. We could have started from ENS files containing mono-ping current velocity data in beam coordinates and regenerated LTA/STA type files via CASCADE PROCESSING (not documented here). We did this work for a few cruises and found that VmDAS did a reasonable work in averaging the raw data (ENS) to create LTA/STA files. We thus consider the LTA/STA files as a good compromise between simplicity of data processing (conversion of current data from beam coordinates to geographical coordinates being directly performed by VmDas) and accuracy. Note that if the user wants a finer temporal resolution than the one defined in the LTA/STA files, he can very easily regenerate them by replaying the data with the 'Reprocess Data' option of the VmDas software.

Notes on VmDas:

- When acquiring data with VmDas,
  - the ADCP filters the data internally according to a correlation criterion. For an ADCP OceanSurveyor, the minimum correlation threshold is 120/255. The received signal must have a minimum of 47% correlation with the transmitted signal to be declared valid.
  - VmDas then applies the thresholds defined by the user in the DataScreening tab if these are activated.



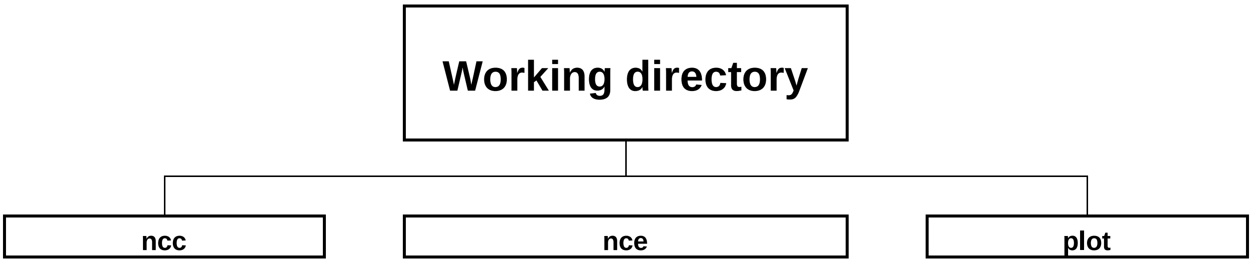
Note that the thresholds defined in DataScreening are applied only if the data is acquired in BroadBand mode. These thresholds are ineffective if the acquisition is made in NarrowBand mode (VmDas bug reported in RDI in April 2017).
The values of the thresholds actually taken into account are saved in the LTA/STA files except for the amplitude (RSSI) and vertical speed thresholds.

Important notes:

- STA and LTA files must be files configured in **NarrowBroad (NB) or BroadBand (BB)** mode. In the case of files configured in mixed mode (alternating NB and BB pings, which is possible with the OS), CASCADE only reads the data associated with the first mode transmitted.
  Compared to BroadBand mode, NarrowBand mode allows for greater depth but the data is less accurate. If the user wishes to focus on depth, NarrowBand mode is the preferred mode. If the user wants greater accuracy, they will use BroadBand mode.
- Be careful, **only STA (or LTA)** files with the same cell size can be concatenated together. These files can contain bottom-ping data or not, it doesn't matter because bottom-ping doesn't affect the cell size. If the user is analysising a cruise where the *.LTA (or *.STA) files have different cell size, then he will have to analyse these files by group of identical cell-size and perform an analysis for each group. A MATLAB function provided with CASCADE, ReadLogs.m, allows to read all the *.LOG files and generate a synthesis file: LogCruise.csv which summarizes the main acquisition parameters file by file. The size of the cells as well as the transmission mode (NB or BB) are notably indicated (cf. 5.1.3).
- If the user concatenates files with the same cell size but a different mode (some files in NarrowBand mode and others in BroadBand mode), there may be an inconsistency in cell depths. This is because the first cell's depth is dependent on the transmission mode. It is therefore not advisable to concatenate together files with different modes (although the offset rarely exceeds 2 meters). Therefore, it is preferable to perform an analysis by mode.
- STA files should not be concatenated with LTA files, at the risk of having date issues. The user decides whether to work with STA files or LTA files; STA files have a better temporal resolution than LTA files.

2.3 Recommended work tree

It is recommended that you choose the following work tree structure:



The working directory is the directory where CASCADE software saves all the files and plots generated during data analysis. It includes:

- three subdirectories:
  - **ncc**: includes:
    - the working file, different cruise files generated during data analysis (e.g. cruise file after filtering, cruise file after correction (misalignment, trim, amplitude), ...)
  - **nce**: includes Station and Section files.
  - **plot**: contains all the plots generated by the different stages of the operation.

  If these 3 subdirectories do not exist in the working directory, CASCADE creates them automatically.
  All files generated via CASCADE will be saved in one of these subdirectories in the working directory.

- The configuration file **conf_exploit.mat**. This file is a Matlab file generated by CASCADE. It stores the different parameters entered by the user. Initially, it contains default values. These values are replaced by the values entered by the user during analysis, and saved for a future data analysis session.
- The STA, LTA, Station List and Section List files can be located anywhere since during data analysis the user must explicitly indicate them (full name: path + filename).

2.4 Software tree structure

- **exploitation**: This directory is the "main directory" of the CASCADE software.
- **exploit**: This directory contains all CASCADE -specific calculation functions.
- **ihm:** Contains the functions more specifically dedicated to the user interface as well as the logos.
- **tools**: directory including general functions used by CASCADE (Julian day's calculation, average's calculation without taking into account NaN,).
- **bathymetrie:** Default directory used to read bathymetry files. Basically, it contains 5 bathymetry files. The user can add his own bathymetry files (be careful with the format) (cf. 2.1). If the user wishes, he can also define, via the interface, another directory from which CASCADE will retrieve the bathymetry files.
- **tide**: directory containing the functions useful to add the tide as well as a model subdirectory containing the default tide files (cf. 2.1). Note that the user can define, via the interface, his own tide directory in which he will have to put his tide files.

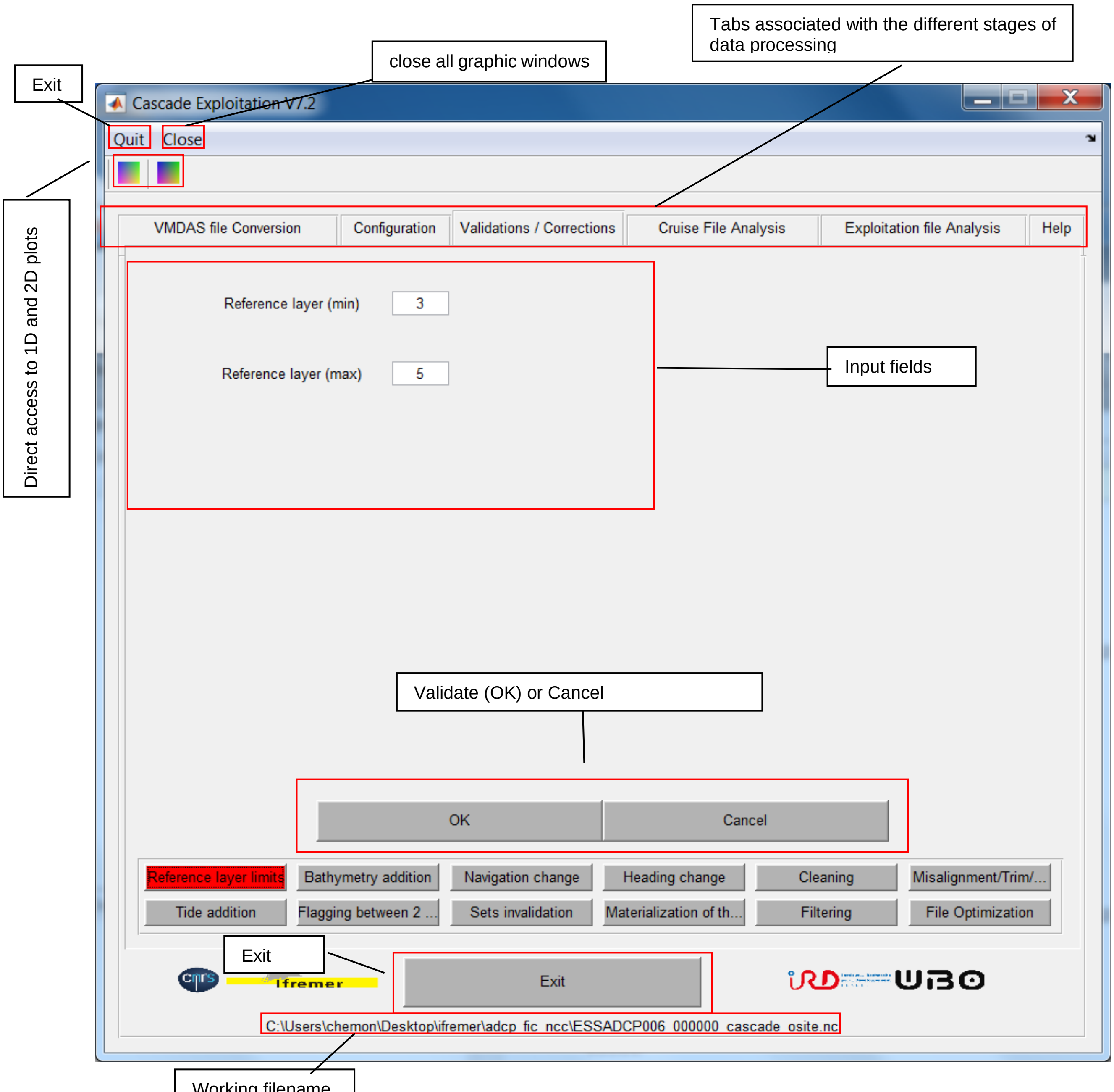
3 Data Exploitation

To be able to run CASCADE in MATLAB, the user must first change the MATLAB path in order to access the MATLAB functions developed for CASCADE.
If **CASCADE** _PATH is the directory containing the CASCADE software, it can be done via the MATLAB command: **addpath(genpath(CASCADE _PATH))**
To run CASCADE, in MATLAB, the user must then go to his or her working directory and start the application by:

- **ce** for the french version
- **ce('en')** for the english version

3.1 General Interface

The interface is as shown below



CASCADE 'S INTERFACE

If the texts overlap in the interface (depending on the screen size), the user can set the DefaultUicontrolFontSize property via the MATLAB command:

set(0, 'DefaultUicontrolFontSize',7): the user will adjust the size (7) according to his screen.

At each step, as long as the "Validate" button has not been selected, it is possible to return to the previous values by clicking on "Cancel".

In the upper part of the window, the tabs described below allow you to move from one step to another:

Tab « VmDas File Conversion »

It allows the STA or LTA files's conversion. The user indicates his working directory, where the files generated by CASCADE will be written, as well as an already concatenated LTA or STA file. This will be converted into a cruise file.

Tab « Configuration »

This step is used to configure the working environment, to select the bathymetry and tide directories, and the cruise file to be used (= working file).

Tab « Validations/Corrections »

This step works on the working file. This working file can be modified (quality flags and external data (bathymetry, tide) can be added). It can also be used to generate new cruise files following data filtering or correction (correction of the ship's speed, ADCP misalignment, and so on.). This step results in cruise files containing corrected and validated data.

Tab «Cruise file Analysis »

This step is used to visualize the variables in the working file, to compare the current speeds with those of the ship or with those from the Bottom-ping, ... These tools are used to check the quality of the analysis done in the previous tab: the working file is not modified. The Station and Section files are also created in this tab. These are saved in the subdirectory nce.

Tab « Exploitation file Analysis »

This step works on the Station and Section files created in the previous tab. The file's variables can be displayed on different plots (vector, contouring,...).

Tab « Help »

This tab allows the user to access at any time during the analysis to various information concerning the quality flags or to various information concerning the cruise file (analysis applied, % good data, ...).

When CASCADE is launched, the interface opens on the first tab.

Notes:

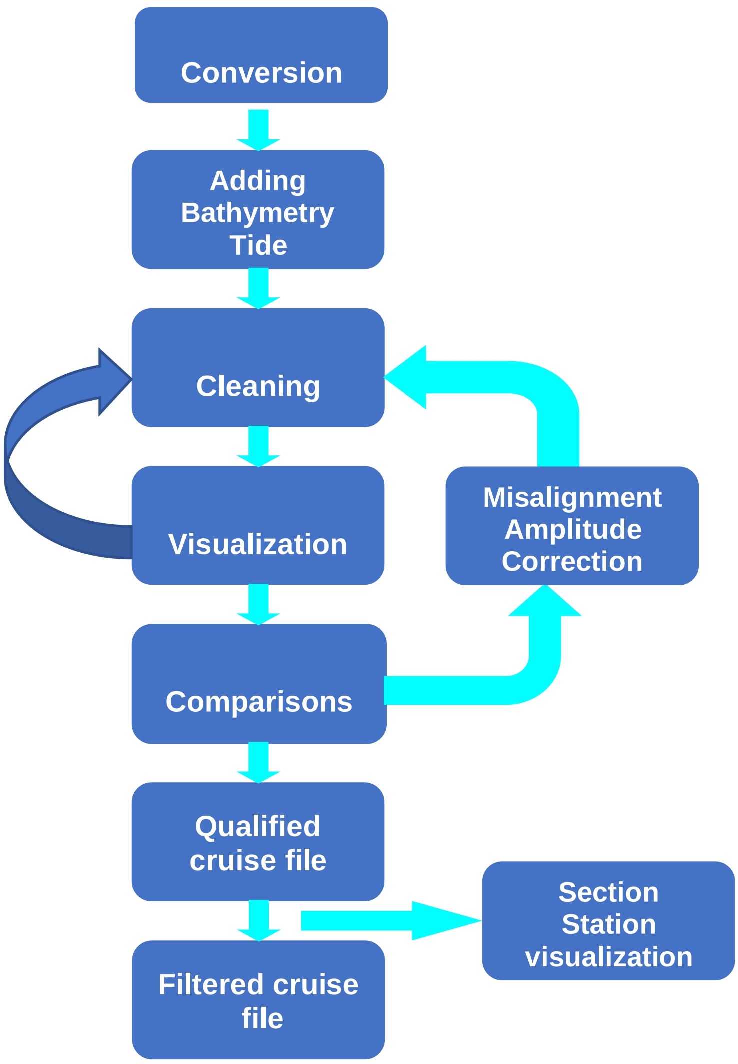
- If no conf_exploit.mat file exists in the working directory, CASCADE automatically creates one with default values for the different data flagging criteria. Note that the default values are correct for most of cruise. However, the user can modify them according to his or her knowledge of the data region.
- The values are saved for each session in the conf_exploit.mat file. Also, the next time the software is launched in the working directory, the user finds the last values he entered in the previous session.
- If the user retrieves a processed NetCDF cruise file without the associated conf_exploit.mat file, the ncc2confexploit.m function, provided in the tools directory, allows to create a conf_exploit.mat file from the information in the NetCDF cruise file. For example, information such as the cruise's name, the ship's name., as well as data analysis information, if any, will be taken directly from the NetCDF cruise file.
  This function is to be called from the user's working directory. The NetCDF cruise file must be located in the ncc sub-directory. The call to the function:
  ncc2confexploit(NetCDF_cruise_file).

3.2 General concept

The user (called "he" later on for simplicity) concatenates all the STA (or LTA) files of his cruise associated with the same configuration (same cell size, same transmission mode (BroadBand or NarrowBand)). He can also choose a single file (several hours long) to test. He launches the CASCADE software under MATLAB. He converts the STA (or LTA) file into a cruise file that becomes the working file. He adds external data such as bathymetry and tide.

He then performs an iterative process: data cleaning, data visualization, he can compare current data with ship data or bottom data. As a result, he corrects the data for possible misalignment and amplitude factor, bad navigation, incorrect ship's heading, ...; he cleans the data again, visualizes them, cleans them again if some good data has been removed or if some bad data hasn't been removed, until he is satisfied with the results. He can then evaluate the current data along sections on the ship's track and/or during stations, a station corresponding to a period of time when the ship is at a fixed point (time operation).

Cleaning consists in assigning a quality flag to the data (good, doubtful, bad data, ...etc). In no case does CASCADE delete the data.
The iterative process is presented below:



3.3 *The different steps*

3.3.1 Tab « VmDas file conversion»

This step corresponds to the STA or LTA files's conversion. The user indicates:

- the working directory, where conf_expoit.mat is saved and where all the files generated by CASCADE will be saved (in the subfolders)
- the full path of the concatened STA (or LTA) file (cf. 2.2.2.2)
- the ADCP's type
  - OceanSurveyor: most of the current ADCPs (also known as OS)
  - NarrowBand: This type of ADCP corresponds to the older NarrowBand ADCPs that were installed on Genavir ships prior to the installation of the OceanSurveyor ADCPs. The NBs are normally only operated with the old TRANSECT software and therefore do not generate STA files.
  - BroadBand: This type of ADCP corresponds to the older BroadBand ADCPs that were installed on Genavir ships prior to the installation of the OceanSurveyor ADCPs.
  - UNKNOWN: in case the user does not know the ADCP's type.

The ADCP's type is used to generate an error estimate based on the characteristics of the instrument (cf below). For Ocean Surveyors ADCP, be careful not to confuse the ADCP's type (OS) with its transmission mode (BB, NB or mixed) which will be automatically detected.

The input STA (or LTA) file (filename) is converted to a cruise file with the following nomenclature: <filename_**osite**>.nc. It is generated in the ncc subdirectory of the working directory.

Notes:

- The date (resp. position) assigned to a ADCP ensemble is the average between the 2 dates (resp. positions) associated with the first and last pings of the ensemble.
- The cells's depth corresponds to the cell's middle.
- Ship's speeds for each ADCP ensemble are estimated from the first and last positions of the ensemble and associated dates.

  Remark:

  WinADCP determines ship's speed using the direction and the modul of the ship's speed calculated by VmDas. The ship's speed between CASCADE and WINADCP are therefore different. According to our tests, the ship's speed calculated by CASCADE are more consistent and more similar to the ship's speeds over the bottom measured by the bottom-ping.

- In STA (or LTA) files, current's speeds are the relative current's speeds with respect to the ship. The ship's speed is added to these relative speeds to calculate absolute current speeds:
  U_absolute = U_relative + U_ship
  V_absolute = V_relative + V_ship
- The correlations and intensity echoes of the 4 beams are averaged, for each cell, to create the CORR and ECI variables in the NetCDF cruise file, corresponding to the average correlations and intensity echoes on the 4 ADCP beams. The correlation ensures that the signal received by the ADCP is indeed the echo reflected by the particles of a signal that the ADCP itself has emitted. The intensity echo measures the backscatter intensity of the signal in "counts", a unit unique to each instrument. This variable is used as a relative value to detect bottom, interference and possibly zooplankton migration or the presence of particles or bubbles.

- Since CASCADE version 7.1, the accuracy of ADCP measurements on a ping is introduced in the file (variable **VRMS_ADCP**). It corresponds to the standard deviation of the measurement in a single ping as given by the manufacturer. It depends on ADCP's type, transmission mode and cells size. The file contains one precision per ensemble. Below is a summary of this precision for different ADCPs and configuration:

OceanSurveyor ADCP - mode NarrowBand:

	38 Khz	75 Khz	150 Khz
Cell's size 4m			30 cm/s
Cell's size 8m		30 cm/s	16 cm/s
Cell's size 16m	30 cm/s	16 cm/s	
Cell's size 24m	20 cm/s		

OceanSurveyor ADCP - mode BroadBand:

	38 Khz	75 Khz	150 Khz
Cell's size 4m			15 cm/s
Cell's size 8m		15 cm/s	8 cm/s
Cell's size 16m	15 cm/s	7 cm/s	
Cell's size 24m	10 cm/s		

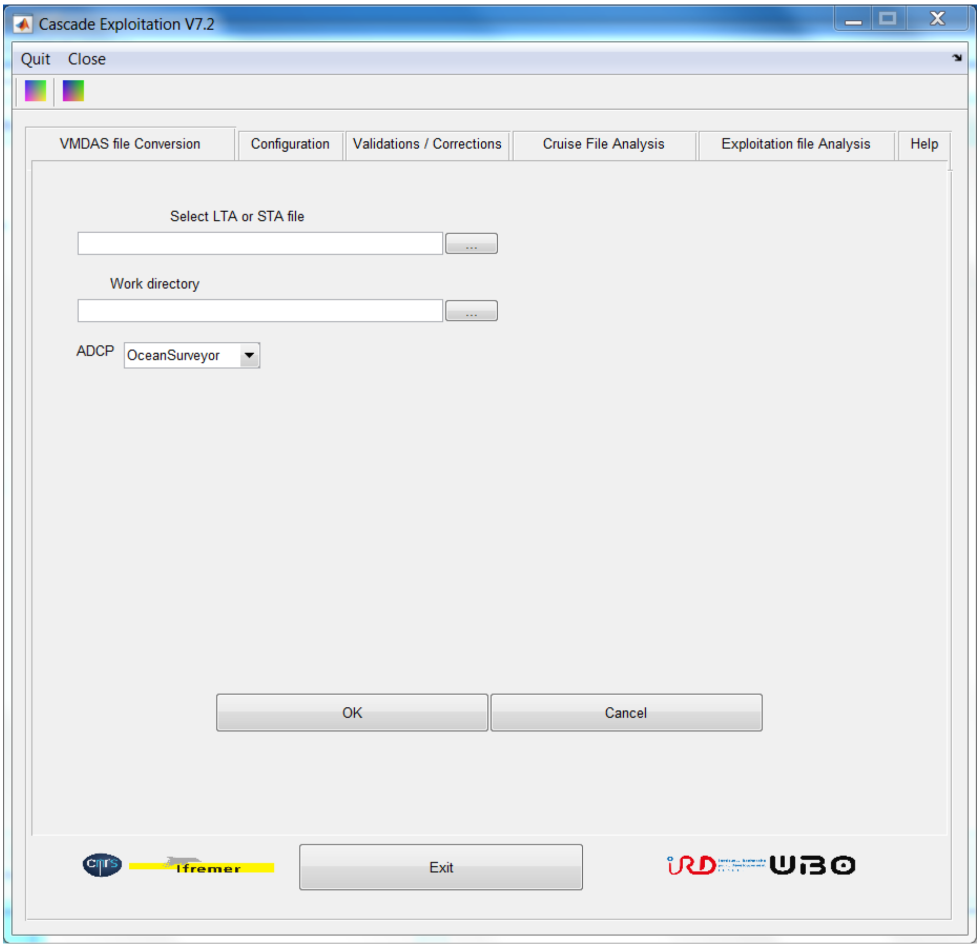
NarrowBand ADCP:

	75 Khz	150 Khz	300 Khz	600 kHz	1200 Khz
Cell's size 1m				26 cm/s	13 cm/s
Cell's size 2m			26 cm/s	13 cm/s	6.5 cm/s
Cell's size 4m		26 cm/s	13 cm/s	6.5 cm/s	
Cell's size 8m	26 cm/s	13 cm/s	6.5 cm/s		
Cell's size 16m	13 cm/s	6.5 cm/s			
Cell's size 32m	6.5 cm/s				

BroadBand ADCP:

	75 Khz	150 Khz	300 Khz	600 Khz	1200 Khz
Cell's size 0.12m					
Cell's size 0.25m					10 cm/s
Cell's size 0.5m				10 cm/s	4 cm/s
Cell's size 1m			10 cm/s	4 cm/s	2 cm/s
Cell's size 2m		10 cm/s	4 cm/s	2 cm/s	1 cm/s
Cell's size 3m	15 cm/s	4 cm/s	2 cm/s	1 cm/s	
Cell's size 8m	5 cm/s	2 cm/s	1 cm/s		
Cell's size 16m	3 cm/s	1 cm/s			

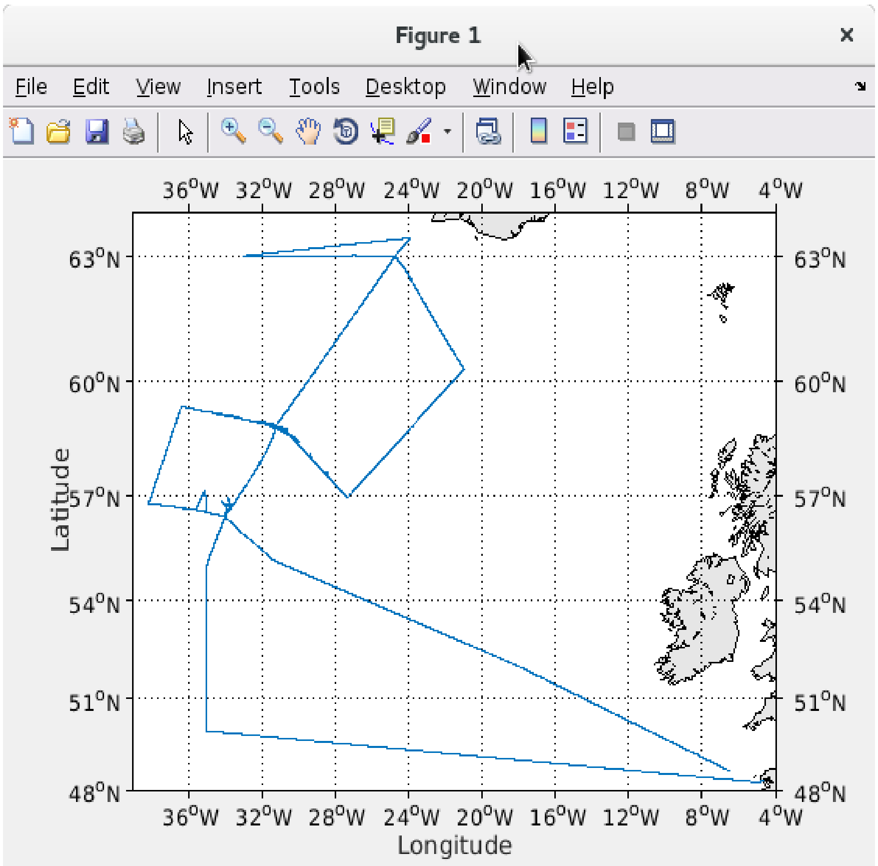
- The error on the current's speed measurement is also introduced in the file (variable **VError**). It is an instrumental error based on the manufacturer specifications of the OS (table above). It is calculated as follows:
  precision/sqrt(number_of_ping).
  The number of ping taken into account depends on the depth. The file therefore contains one measurement error by ensemble and by cell.



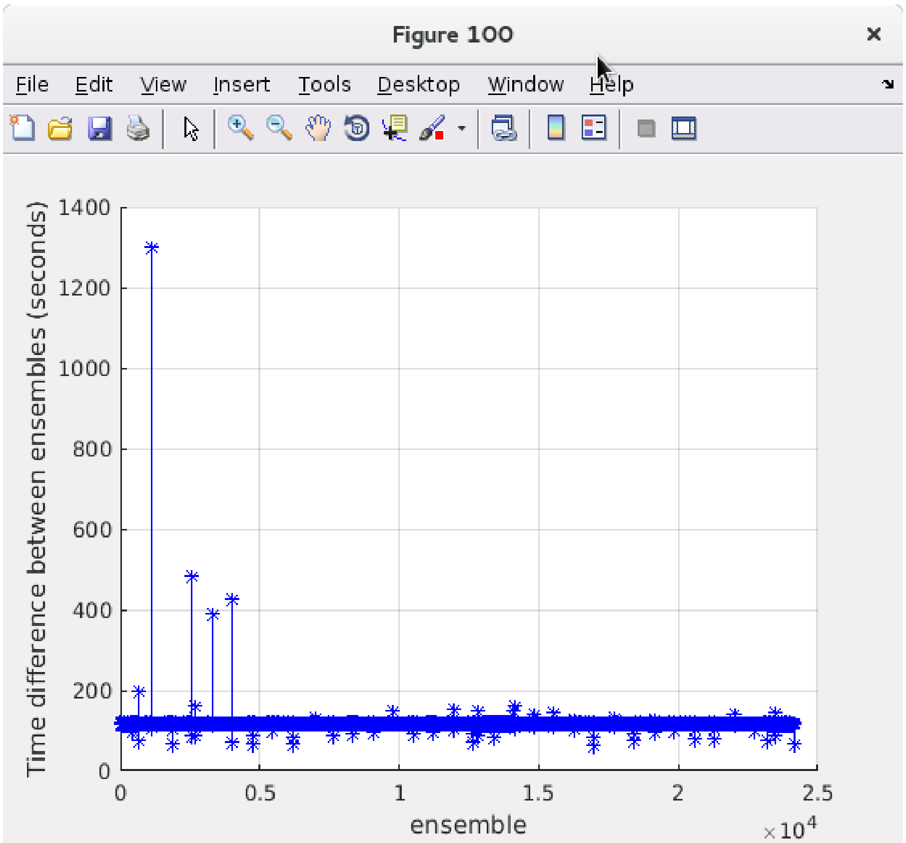
Interface

After the conversion, 4 plots are generated:

- A ship's track plot associated with the input STA (or LTA) file. This plot ensures that the user has correctly concatenated all the cruise files.
- A plot representing the time difference between each ensemble. Generally, the data in the STA (or LTA) files correspond to the mono-ping data averaged over 2 (or 5) minutes. However, each user configures the ensemble's duration as he wishes. This ensemble duration must be found on the plot. Interruptions in acquisition are also marked on the plot.
- A plot representing the difference between the ADCP acquisition PC's time and navigation time. It should be noted that it is preferable to have a correct acquisition PC time, even if an offset of a few seconds does not alter the processing carried out by VmDas.
- A plot representing the number of pings averaged by ensemble.



Ship's track

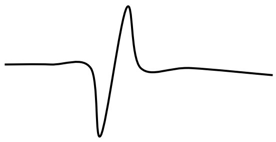


Time ensemble difference

The "Figure 100" shown above represents the time difference between the start of each ensemble according to the ensemble number (incremented sequentially from 1). Here, we can see that globally, there is a difference of 120 seconds, or 2 minutes between each ensemble. The data of an ensemble correspond to the mono-ping data averaged over 2 minutes. The user makes sure that this is consistent with the VmDas configuration. This figure also helps to detect possible clock problems:

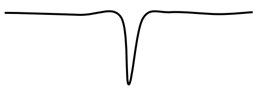
- Positive peaks indicate a ADCP stop.
- If **negative values** appear on this plot, it indicates a large clock jump (a step backwards). By reading the following points, you can determine whether it is a one-time jump followed by a clock reset or an offset that is reflected in subsequent data.

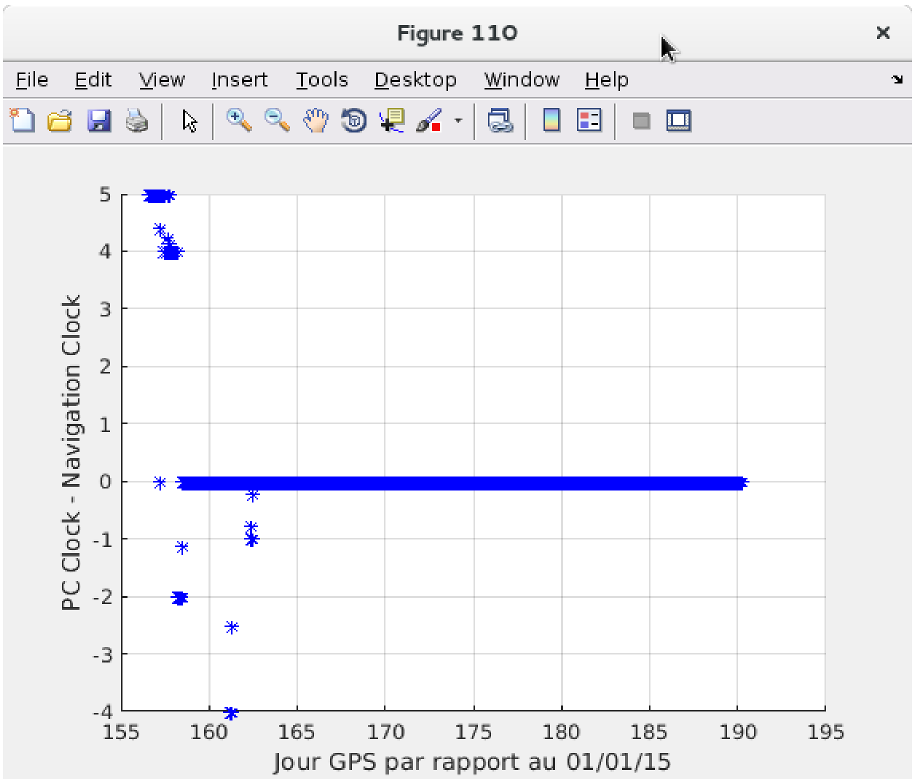
In the case of clock offset catch-up, a positive peak should follow the negative value.



Note that an error of this type is automatically corrected by CASCADE if it concerns only one point (date set in Fill_Value or modified in the case of a change of day) and will not appear on the figure.

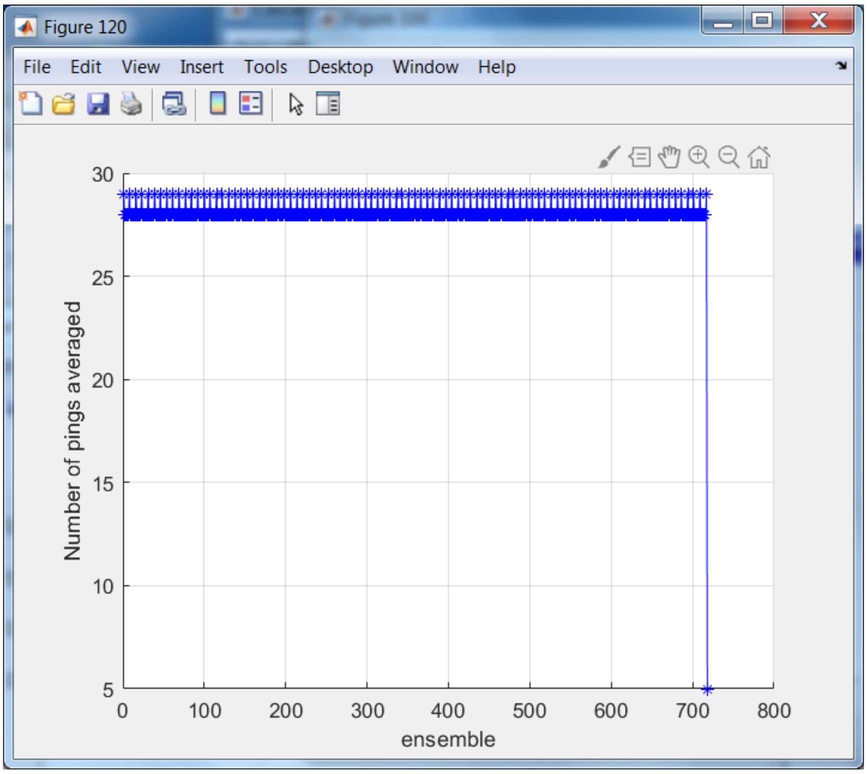
In the case of a backward movement of the clock without resetting it, the negative value is not followed by the catch-up (positive peak); the user is then invited to check the data in order to determine the reason for this backward movement in time (STA files 's order, time change, ...).





Time difference between acquisition PC and navigation

Figure “110” above shows any resetting of the acquisition PC time and ensures that it is on time in relation to the navigation time (which, on board, is UTC time). Here, we can see that at the beginning of the cruise, the acquisition PC time was 5 seconds ahead of UTC time. It was then set on time and the difference between the acquisition PC time and UTC time is then equal to 0. On board, it is advisable to work gradually, resuming the analysis of the data in full every day, concatenating new STA (or LTA) data files as and when required. Thus, if the user notices a problem with the clock, he can quickly remedy it with the help of these last 2 plots.

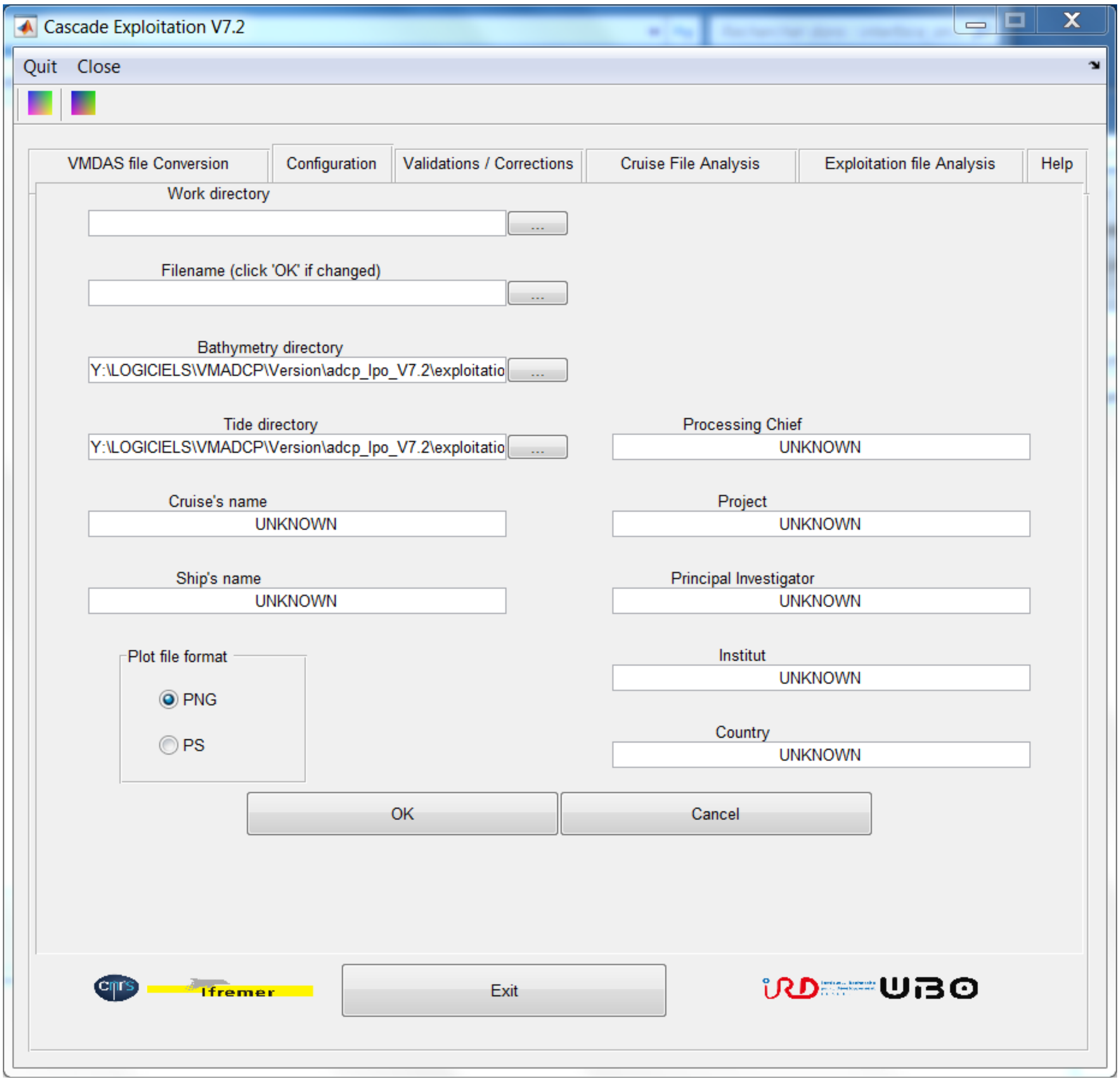


Number of pings averaged by ensemble

3.3.2 Tab “Configuration”

This step configures the user's working environment. The user must specify:

- the working directory. All the files generated by CASCADE will be saved in it, in subdirectories.
- the working file.
  The filename is automatically written in the interface, so that the user can ensure that he uses the good file.
- The tide and bathymetry directories
- cruise's name
- Different informations about the cruise
- ship's name
- ADCP data responsable's name
- cruise responsable's name with the associated organism
- plot format (PNG or postscript).

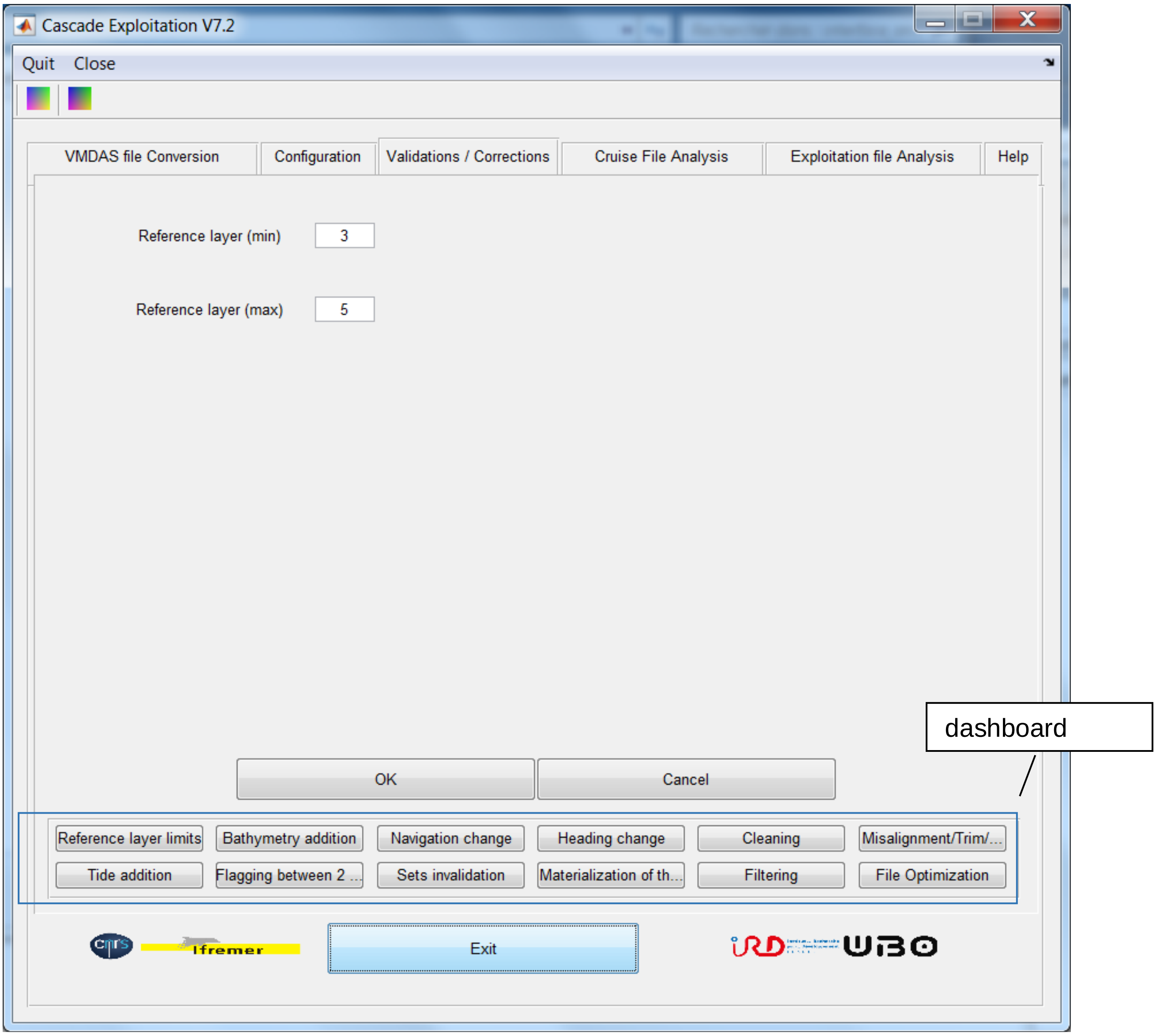


Interface

Once all the fields have been filled in, the user must click on "validate" for these choices to be taken into account. The software then automatically switches to the next tab "validation/corrections".
Note: some fields don't accept all the characters, like space.

Important note: once validated, the working file's name is always indicated at the bottom of the interface.

3.3.3 Tab “Validation/Correction”



Interface

By default, the "reference layer definition" window is selected.

The following paragraphs describe in detail the different choices available in the dashboard for validating and correcting data. To quickly identify the current module in the dashboard, the button for the relevant operation is displayed in red in the interface.

Only the "Reference Layer", " Bathymetry addition" and "Cleaning" modules are required to qualify the data. The other modules are only necessary if the user is not satisfied with the results's analysis (next tab). The addition of the tide must have been done before creating the Station and Section files (cf. 2.1.6 and 2.1.7).

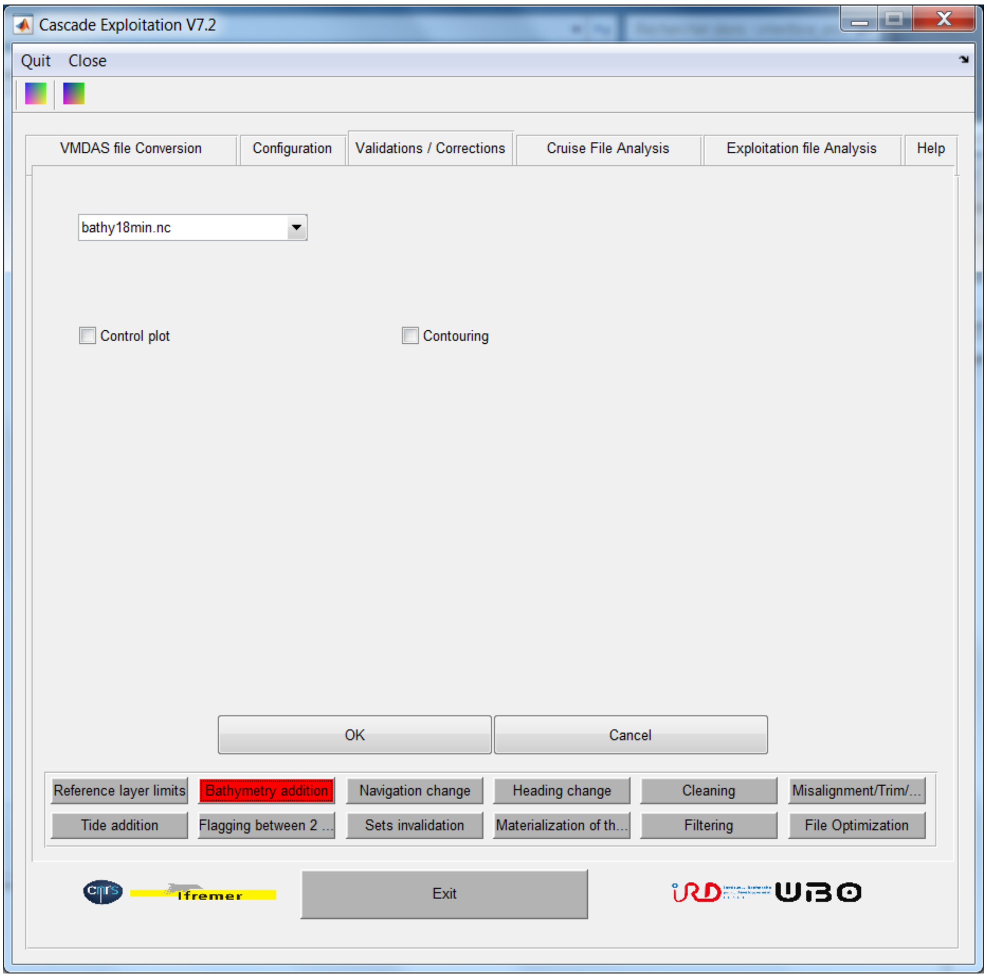
3.3.3.1 - Defining the reference layer limits

The reference layer is used for different criteria's visual verification (cleaning, on the way-station comparison). In addition, for a given profile, if more than half of the data in the reference layer is qualified as bad by cleaning, the whole profile will be qualified as doubtful (see 3.3.3.5).
The reference layer is defined by the cell numbers ("bin") MIN and MAX between which the data are considered as good and consistent with each other.

It is advisable not to take into account the surface layer, which can be disturbed by possible bubbles or ringing, nor the deepest cells where the data are not necessarily homogeneous. Cells 3 to 5, proposed by default, are generally appropriate.

The user enters the 2 values and validates.

3.3.3.2 - Adding bathymetry



Interface

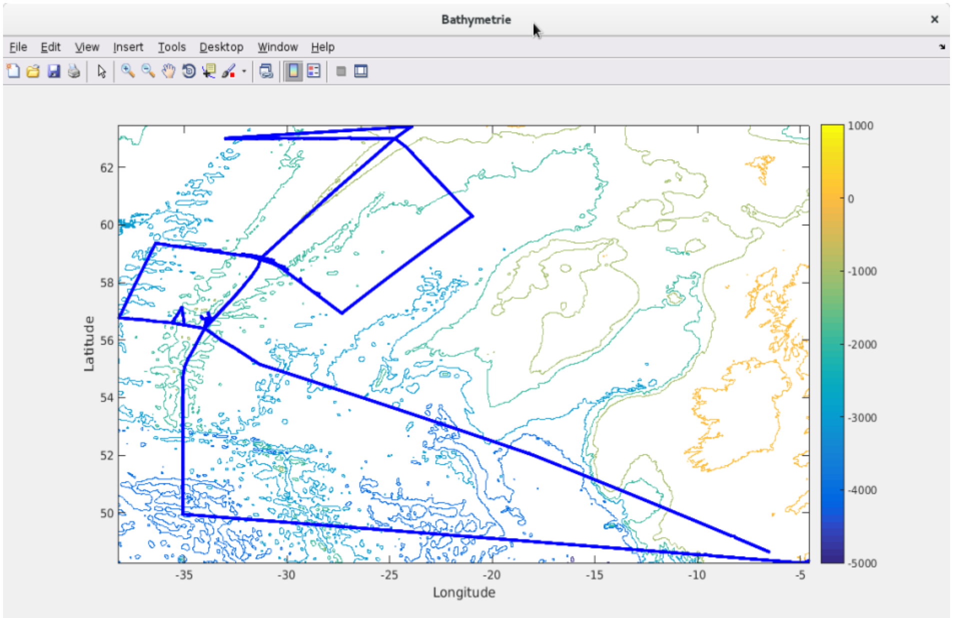
This step combines bathymetry with the cruise trajectory. A BATHY variable is added in the working file at each ADCP measurement point (= ADCP ensemble). The bathymetry of each point is determined from a **2D interpolation** in latitude and longitude of a defined bathymetry.

The interface allows you to choose between different bathymetries via a drop-down list.
By default, the following bathymetries are proposed:

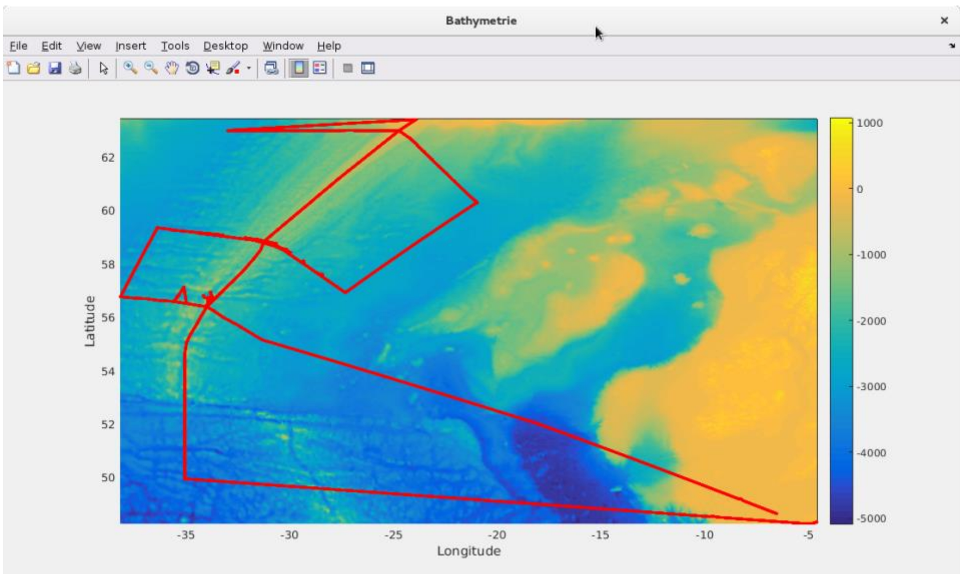
- Gebco (1 minute resolution)
- Etopo1 (1 minute resolution)
- Etopo2 (2 minutes resolution)
- Bathy6min (6 minutes resolution)
- Bathy18min (18 minutes resolution)
- the user can add his own NetCDF bathymetry files and place them in the bathymetry directory. These new files will then automatically appear in the drop-down list.

We advise to use Etopo1 bathymetry, which proves to be very satisfactory in most cases.

If the 'Contouring' option is checked, the bathymetry is represented on the map by isobathimetric lines (otherwise solid contours).

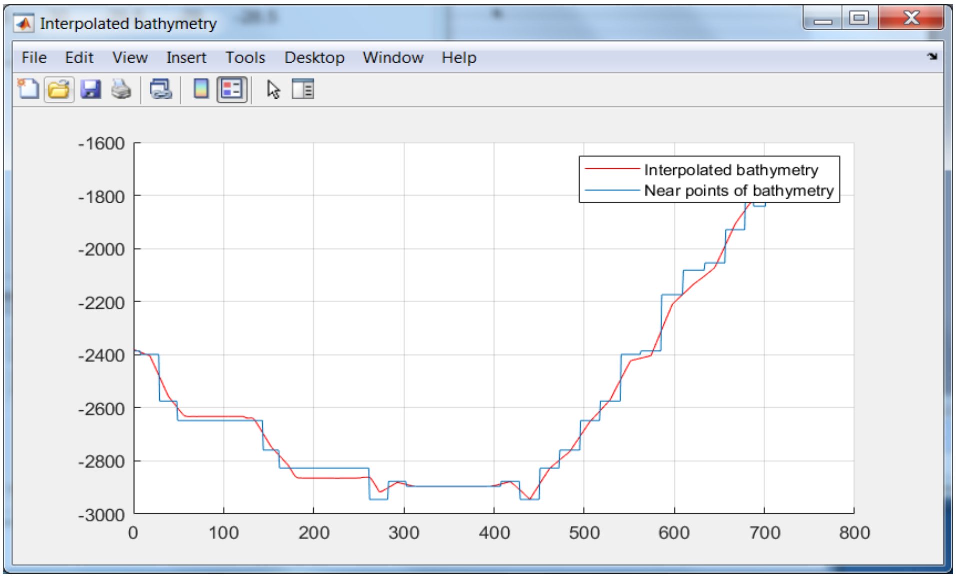


Cruise track with bathymetry isobaths



Cruise track with bathymetry represented by a coloured surface

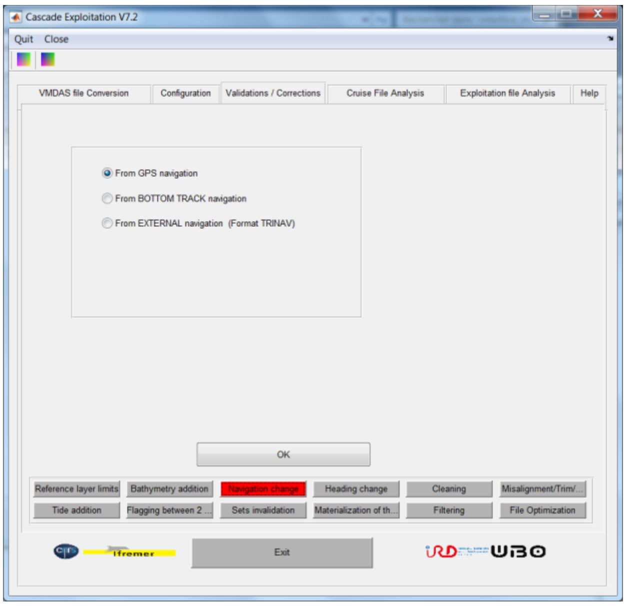
The 'Control Plot' option allows the operator to view the bathymetry on a geographical map plot as well as a comparison between the interpolated bathymetry and that of the nearest grid points (with ensemble number for the X-axis). It helps to check the resolution of the bathymetry.



Comparison between the bathymetry associated with the cruise track (in red) and the bathymetry of the nearest point (in blue)

3.3.3.3 - Navigation change

This interface allows you to recalculate the absolute current velocities from another navigation than the one taken by default by CASCADE. By default, CASCADE calculates the absolute current velocities during the conversion of the STA or LTA file, by adding to the relative current velocities the ship's speed determined from the positions and start and end dates of each ensemble (named JULD_j1 and JULD_j2 in the working file). This original navigation is noted 'GPS Navigation'.



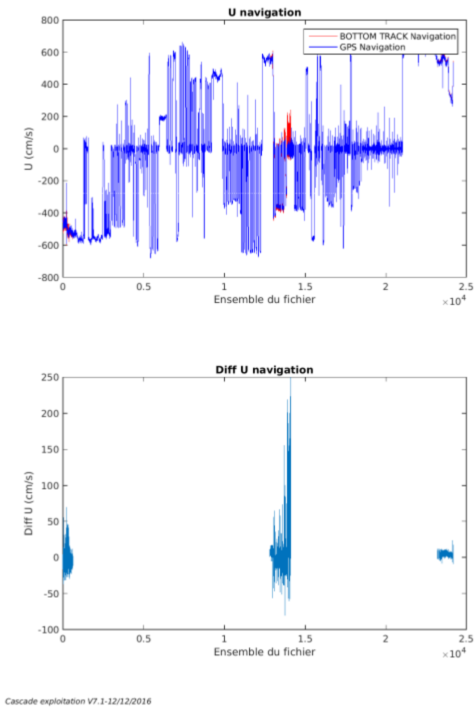
Navigation change

If needed, the user can recalculate the absolute current velocities from another ship's speed (absolute_speed = relative_speed + new_navigation_speed). This other ship's speed can either be derived from:

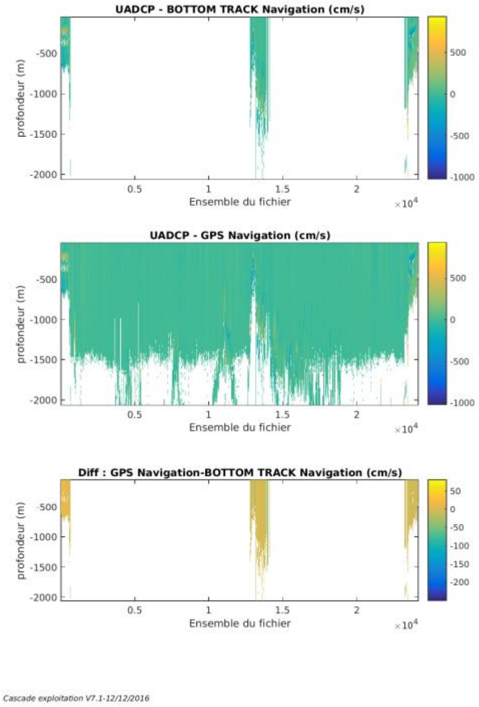
- ADCP speeds from bottom-tracking. In the case of bottom-tracking, the speed measured by the ADCP corresponds to the speed of the bottom relative to the vessel. Since the bottom does not move, the bottom-ping data actually measures the opposite of the ship's speed. The Bottom-Tracking data are available in the NetCDF file resulting from the conversion of the VmDas file, if the Bottom-Tracking option was activated during the acquisition. Note that the 'Bottom-Tracking' data may not be available for the whole cruise (if the option has not been activated for all STA (or LTA) file or if the seafloor is out of range). In this case, this navigation should not be taken into account because the absolute current data will then only be available when the Bottom-Tracking data are also available.
- an external file in TRINAV format (see appendix 5.9) that the user can create from other navigation sources at his disposal. In this case, the variables U_EXTERNAL and V_EXTERNAL are then added to the working file.

Select "From external navigation..." to display the input field for the file name in TRINAV format.

The absolute current velocities before and after taking into account the navigation change are then displayed.



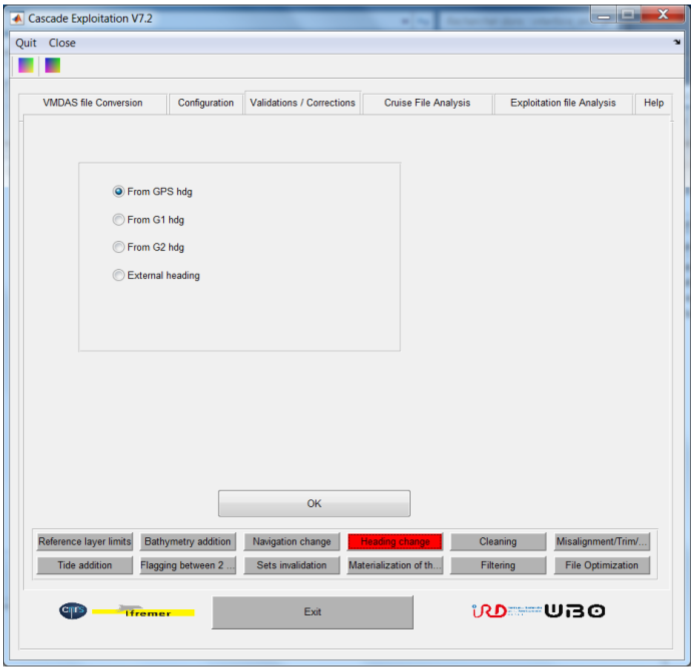
ship's speed comparison



Absolute current velocities comparison: Before and after navigation change

3.3.3.4 - Heading change

At this stage, it is possible to recalculate the absolute speed from a heading other than the one used in real time by the VmDas acquisition software, noted GPS heading (even if it is a hybrid heading). The user can recalculate the speeds from headings G1 and G2 of the working file if these are available. He also has the possibility to recalculate them from an external heading file (cf. 5.8) that he has to create from the diffrent heading sources he can use. The HDG_EXTERNAL variable is then added to the working file.



Interface

The absolute current velocities corrected for heading error are calculated:

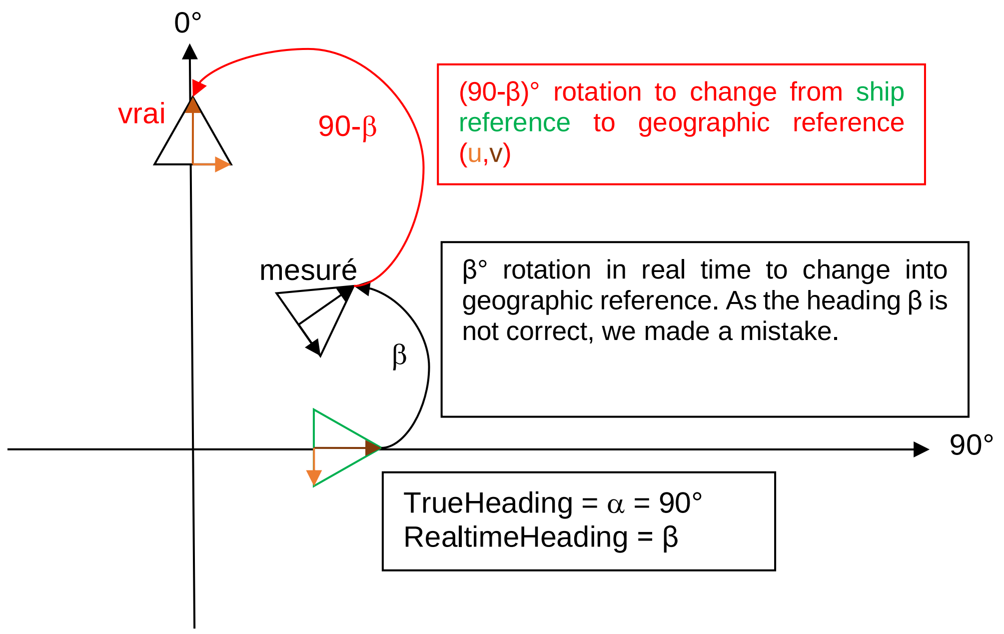
- by applying a rotation of an angle equal to the difference between the new and old heading on **the relative current speeds**.
- then adding the ships's speed to the relative current velocities.

The absolute current velocities before and after taking this change into account are displayed. A comparison plot between the external heading and the external heading interpolated on the ADCP dates is also plotted.

Explanation:
The ship is heading to α while the heading provided in real time is β. In real time, to move from the ship's fix to the North/South geographic fix, a rotation of β was applied to the relative current speeds. To apply the true heading α, therefore, one must:

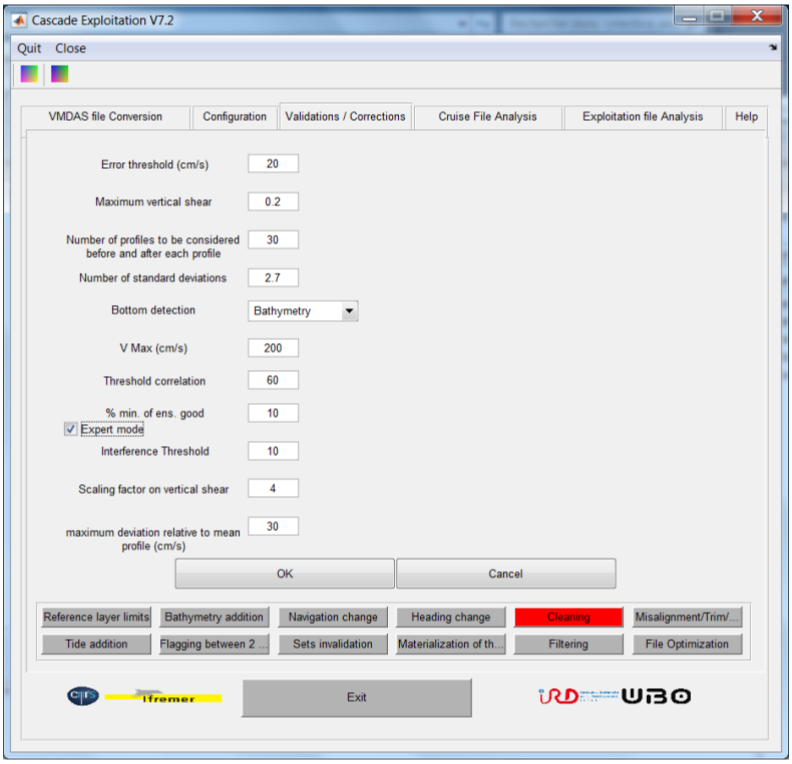
- return to the ship's reference by applying a rotation of -β to the relative current velocities
- change the relative current velocities from the ship's reference to the North/South geographic fix by applying a rotation of α.

In the end, to correct the relative current velocities of a heading error, it is therefore necessary to rotate theses speeds by an angle (α-β).



3.3.3.5 - Cleaning

The cleaning allows to associate a quality flag to the absolute current speed data according to different criterias. This flag is saved in the working file. At the start of the cleanup, all data is assumed to be good, so all flags are set to 1, so **any new cleanup invalidates the previous cleanup** (including invalidated data). The flags are set according to different criterias.



Interface

Notes:

- The default values proposed by CASCADE can be used for a first cleaning. Depending on the result of this first cleaning, the user can adapt them to his or her geographical region.
- The expert mode is only to be used by specialists. The default values proposed for the associated flagging criteria are suitable in most cases.

Data flagging is performed as follows:

- **Flag 8**: data under the bottom

  The user can locate data with a depth cell greater than bottom. To do so, the user chooses to determine the bottom by:
  - using bathymetry data if bathymetry has been added to the working file
  - using bottom detected by bottom-ping.

    Bottom-ping may not be configured for all the cruise. Depending on the ADCP's range, it may also be that the ADCP does not detect the bottom. Also, when bottom-ping bottom data is not available, it is supplemented by bathymetry data if the latter has been added to the NetCDF cruise file.

Note that the ADCP emits directional beams in one main direction. Nevertheless, secondary emissions (side lobes) outside these beams can also hit the bottom and disturb the backscattering of the main beam by the suspended particles, which can then generate aberrant velocity measurements. To avoid data being disturbed by the side lobes, when we eliminate the data under a bottom F, in the end, we eliminate the data under the bottom F' = F*cos(beam_angle).

Any cell whose depth P is such that (P+cell_size/2) is greater than F' is flagged to 8.

*Example: The bottom, resulting from bathymetry or bottom-ping, is located at a depth of 1000m. The cell size is 24m, the beam angle is 30°. All the cells with depth P greater than 854m (1000*cos(30)-(24/2)) are considered below the bottom.*

- **Flag 7:** Cells for which there are no ADCP measurements (so-called 'missing data') are flagged to 7. This can be the case, for example, when the user has set too many cells, incompatible with ADCP. The last cells may then contain no data at all. Note that a preliminary data cleanup is already performed in VmDas, sometimes resulting in no ADCP data in the STA files (see VmDAS User's Guide, 2016).
- **Flag 6:** Data are flagged to 6 when:
  - absolute current velocities are greater than a user-specified value (V max (cm/s) in the interface, 2m/s by default).
  - correlation is lower than a user-specified value (Correlation threshold in the interface, 60 by default). Correlation ensures that the signal received by the ADCP is the echo reflected by water particles from a signal previously transmitted by the ADCP.
    The correlation is a value between 0 and 255. For an OceanSurveyor type ADCP, in BroadBand mode, the value 255 corresponds to a perfect correlation. In NarrowBand, normal values are between 170 and 190. Lower values mean a reduced signal-to-noise ratio.
  - pings have been disturbed by another acoustic device. Depending on the ADCP's frequency, some instruments, depending on their own frequency, may be 'heard' by the ADCP and thus cause erroneous data. In order to use this criterion, the user must check the 'expert mode' and enter the **interference threshold** in the interface which defaults to 10. A graphical figure is provided to visualize interferences.
- **Flag 5:** Data are flagged to 5 when:
  - vertical velocity error (EVEL_ADCP) is greater than a user-specified value (**vertical velocity error threshold** (in cm/s) in the interface, 20 cm/s by default). Remember that the error is computed in VmDAS as the difference in the
  - good data percentage is less than a user-specified value (**min. % good overall** in the interface, 10% by default).
    - For NetCDF cruise files resulting from the conversion of a VmDas file by CASCADE, the good data percentage taken into account is the data percentage calculated on 4 beams within the current ensemble.
    - For NetCDF cruise files from CASCADE TREATMENT, this is the data percentage calculated with 3 or 4 beams within the current ensemble.
- **Flag 4:** Data with a peak in the vertical shear are flagged to 4. Vertical shear is the difference in absolute current velocities over the vertical. The threshold set by the user in the interface (**maximum vertical shear**, 0.2 by default) is the maximum allowed for the vertical shear difference between two adjacent vertical cells (in s^{-1}). This allows peaks' detection in vertical shear without eliminating strong shears that may be physical. The vertical shear histogram is displayed to help the user in the choice of this parameter, and the value is usually set towards the Gaussian's tail.
- **Flag 3:** Data that do not pass the following median deviation test are flagged to 3:
  - Each absolute current velocity is compared with neighbouring velocities at the same depth level.
    - median calculation (Med) from absolute current velocities between the data, of index i, and the Nb ensembles neighbouring on both sides. Nb is set by the user (**Number of profiles to be considered before and after each profile** in the interface, 30 by default).
      Med = median(absolute_velocities(i-Nb: i+Nb))
    - median deviation calculation (EC) from this median Med on these same values
      EC = median(abs((absolute_velocities(i-Nb: i+Nb) – Med))
    - If the current speed deviates from the median (Med) by more than Nb2 times the median deviation (EC), then the current speed is flagged to 3.
      Nb2 is set by the user (**No. of deviations from the mean** in the interface, 2.7 by default).

Nb2*EC is forced to be between [7 30] in order, on the one hand, not to eliminate correct data (if EC small) and, on the other hand, not to keep incorrect data (if EC large).

if abs(absolute_velocity(i) – Med) > Nb2*EC then
absolute_velocity(i) is flagged to 3

Notes:

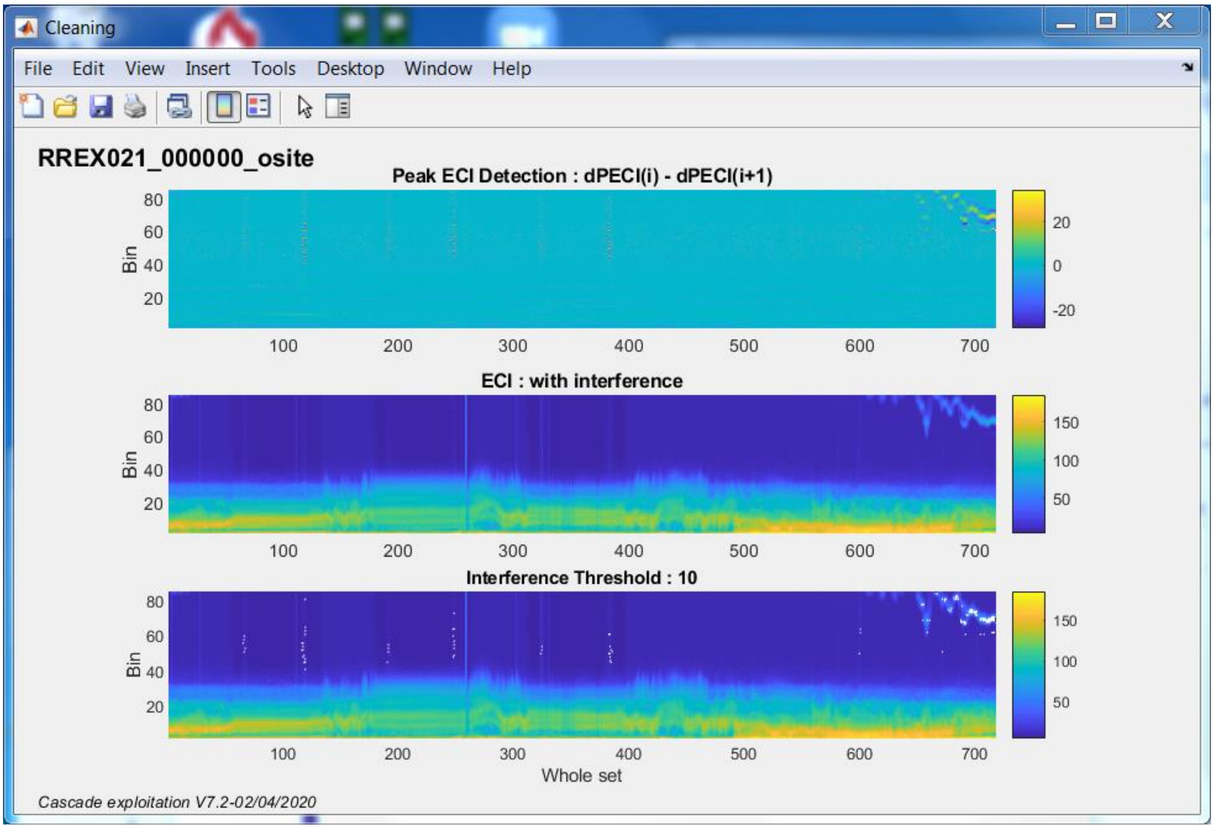
- The test is done for U velocity and then for V velocity.
- If the median is calculated with less than 0.2*velocities number(i-Nb, i+Nb), then velocity(i) is flagged to 3.

- **Flag 2:** Are flagged to 2 (doubtful data) the data not flagged by the previous tests (flag3 to 8) which do not pass the 2 following flagging criteria:
    - **Deviation test between the absolute current velocities profile and the average profile of absolute velocities determined from the 5 neighbouring profiles on either side**
        - average profile calculation on profiles(i-5, i+5).
        - Calculation of the average profile smoothed vertically on 5 cells of the 11 profiles above
        - maximum vertical shear calculation of the smoothed average profile
        - If the profile i deviates from the smoothed average profile by more than V, then it is considered doubtful. V = max([val1*maximum_shear val2]) with val1 and val2 fixed by the user (**shear tolerance factor**, 4 by default, and **maximum deviation from the average profile** (in cm/s), 30 by default, accessible in expert mode in the interface).
    - **singlets and doublets detection: If one or even 2 data are correct but isolated on the vertical (all data above and below are flagged between 2 and 8), then they are assumed to be doubtful and flagged to 2 as well**.
    - **reference layer test:** If more than 50% of the reference layer data has a flag greater than 1, all profile data flagged to 1 under the first doubtful or bad point is flagged to 2**.**
- **Flag 1:** Data assumed to be valid, and which have not been flagged by any of the above criteria, are flagged to 1.

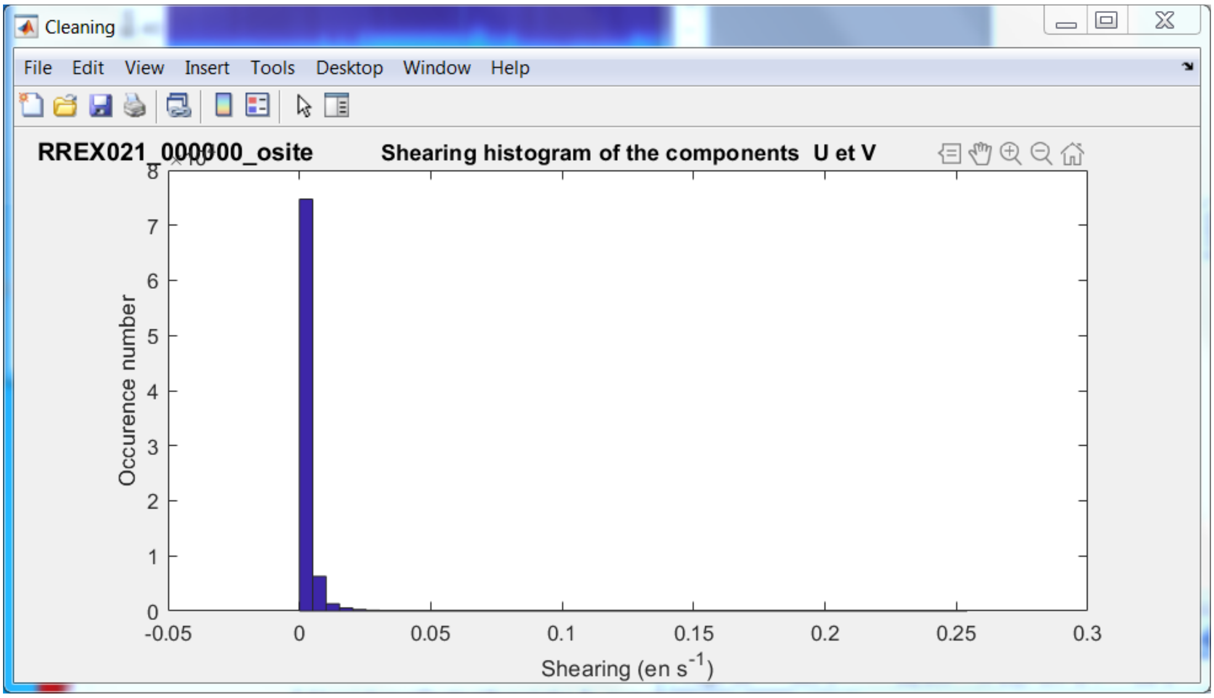
The cleaning is done in three steps:

1. A first function sets the flags of the worst data (from flag 4 to flag 8).
2. The data assumed to be valid after this first cleaning are then compared, level by level, with the `Nb` neighbouring data on both sides. A criterion based on the median deviation (`EC`) from the median (`Med`) is applied: data outside [`Med-Nb2*EC, Med+Nb2*EC`] are flagged to 3
3. The last part identifies the doubtful points (flags 2).

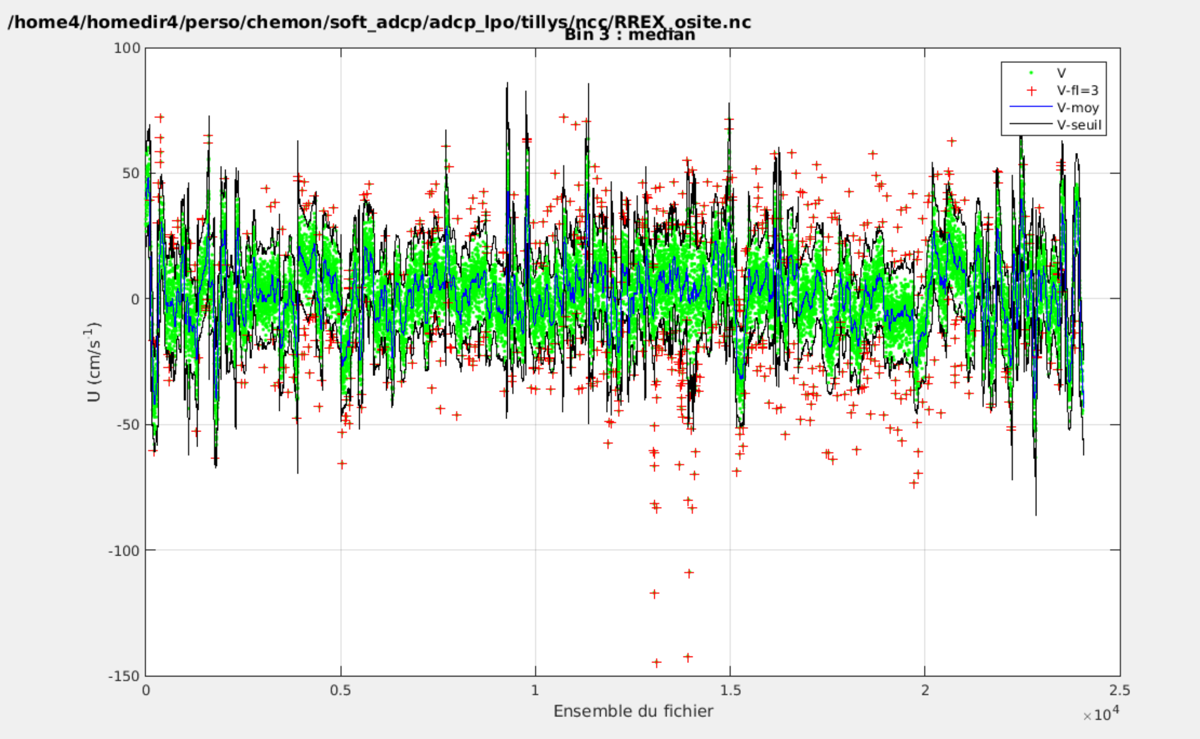
During cleaning, the following graphs are displayed:



Interference test result



current velocities vertical shear histogram



U-velocity plot per bin. In red, data flagged to 3. In blue, median speed. In black, threshold values. X-axis: ensemble ; Y-axis: Velocity (cm/s)

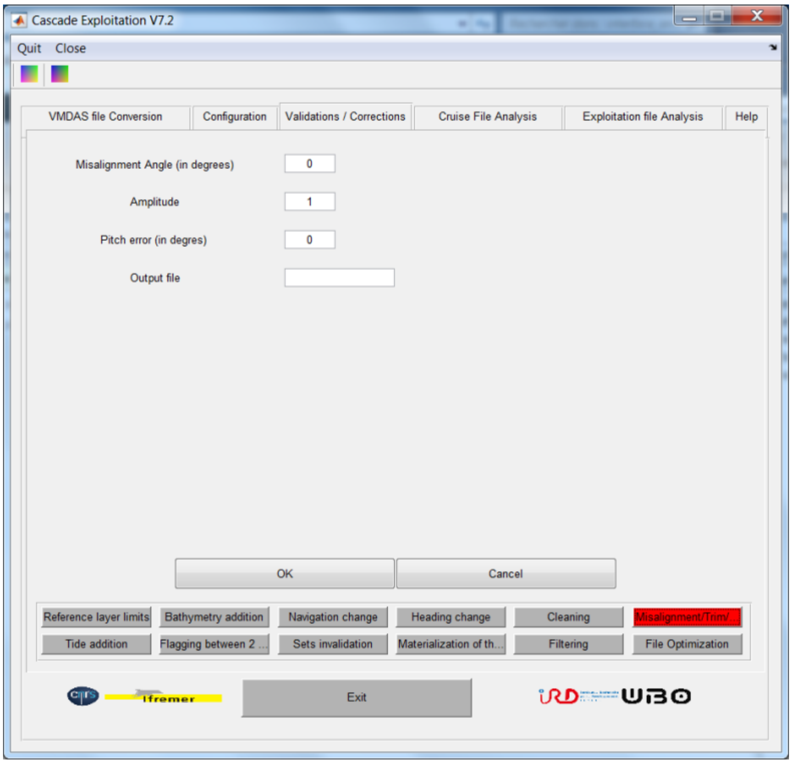
These graphs are automatically saved in the plot subdirectory of the working directory.

When cleaning is complete, a report is displayed in the MATLAB window. In this report, check that the number of errors detected seems realistic. These values are saved and can be viewed by going to the menu: Help → cruise info.

After cleaning, the user can plot the flags and absolute current speeds via 'Cruise File Analysis/2D Plot'. These plots allow the user to check whether certain data has been incorrectly flagged (visibly good data with a flag higher than 2 or, on the contrary, visibly incorrect data with a flag of 1 or 2). The user can then redo a cleanup after changing the parameters of the different flagging criteria. If, however, some data is visibly incorrect but the cleanup fails to flag it correctly, the user can then invalidate this data manually (see 3.3.3.8 invalidating dates and 3.3.3.9 invalidating sets). A quality flag of 9 will then be assigned to them.

3.3.3.6 - Misalignment/trim/Amplitude Correction

When current's velocities seems to correlate with ship's speed (see 3.3.4.4.1), and/or when the average vertical speed calculated on all good data is too high (>1cm/s), it may be necessary to go through this step. This step corrects the current velocities for an error in the ship's trim (pitch), a possible error in the ADCP's angle relative to the ship's axis and/or an error in the amplitude factor (called amplitude). Several methods for calculating these coefficients are developed in the "Analysis of cruise files" tab (3.3.4). Here, the aim is to apply them to the cruise file data.

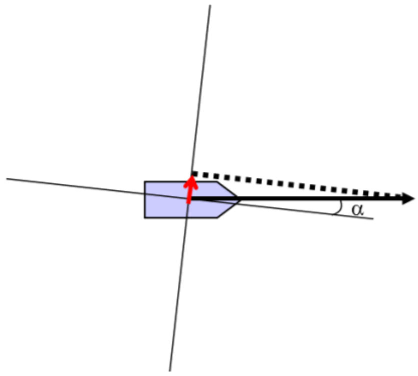


Interface

By default, the misalignment angle and pitch error are set to 0 and the amplitude to 1, corresponding to the case where no correction is to be made.
To estimate the misalignment and the amplitude, the user can refer to the values proposed by comparing the ship's speed with the bottom speeds when Bottom Ping is active (see 3.3.4.4)). He can also refer to the values proposed by comparison between the ship's speed and the relative speed of the current (cf. 3.3.4.8). It should be noted that the values resulting from the comparison of the ship's speeds with the Bottom-Ping are much more reliable. **This is why it is strongly recommended, if possible, during the cruise, to make a cross-travel on the shelf, with Bottom-Ping acquisition. These data will allow a better estimation of a possible error in the ADCP's alignment or/and in the ADCP's amplitude factor.**

When validated, the correction is launched and the new value of the average vertical velocity is given. The correction is made profile by profile for misalignment and pitch. This is done by passing the ADCP relative speeds from the geographical fix to the ship fix, which requires taking into account the ship's heading and pitch, which are specific to each profile. The misalignment and pitch correction is applied to ensure that the vessel is correctly in the fix. The geographic fix is retaken with the ship's heading taken into account. The amplitude correction is then performed on these corrected relative speeds.
A new cruise file is created in the ncc directory: <new_filename>.nc; **this file must now be the working file. It must be selected in the Configuration tab and validated**.

Notes:
- *This step can take several minutes if the cruise file is large.*
- *The flags are not modified: it is advisable to redo a cleanup on the new file.*
- *Misalignment affects perpendicular current velocity. A misalignment of 1° causes a bias on the absolute current speeds of 10 cm/s when the vessel is moving forward at 10 knots.*



- *The amplitude affects the parallel current velocity.*
- *The pitch error acts on the vertical current velocity, and marginally on the horizontal velocity. Note that for Ocean Surveyors, vertical velocity is less accurate because the error associated with the sound's speed is not corrected on this component, but this inaccuracy is generally small compared to the errors due to the ship's heave.*
- *The misalignment/amplitude/tilt correction can only be performed on NetCDF cruise files that have not already been corrected, i.e. only on files for which the amplitude is 1 and the misalignment/tilt is 0. The corrected speeds are corrected using the formula:*
  - *Ucorr = (MG*MP*MH/MG*Uori)/amplitude where:*
    MG: Geographical rotation matrix for moving from ship to land reference
    MP: rotation matrix for pitch correction
    MH: rotation matrix for misalignment correction

Since the product of matrices is not commutative, the cancellation of the correction of a misalignment of an angle A, a pitch of an angle B and an amplitude C is not cancelled by a single correction of a misalignment of an angle -A, a pitch of an angle -B and an amplitude 1/C. It is cancelled out by a pitch correction of angle -B followed by a second correction of angle -A and amplitude 1/C.

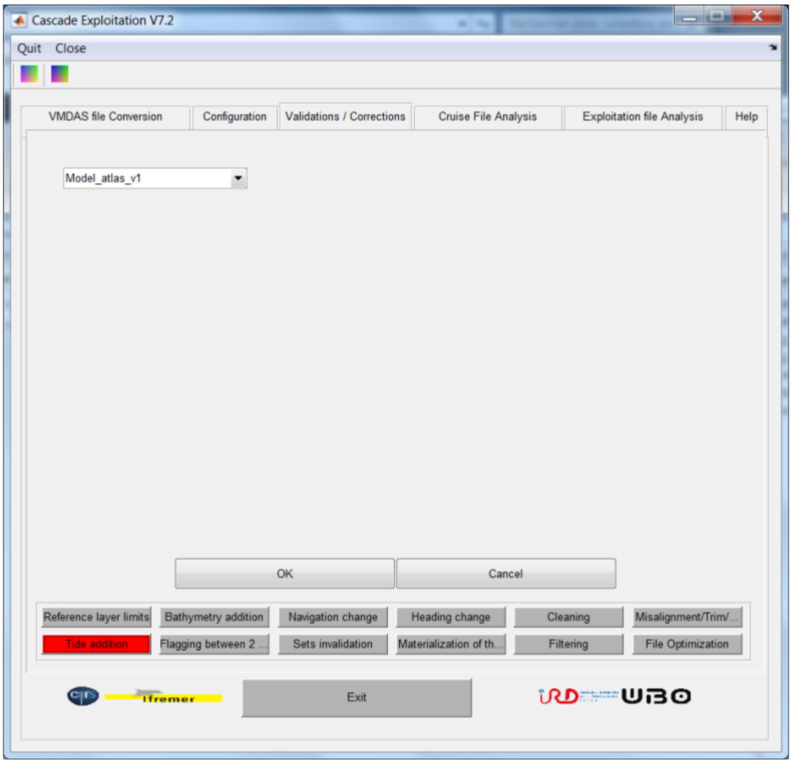
*Ucorr2 = ((MG*MP(0°)*MH(-A)/MG) * (MG*MP(-B)*MH(0°)/MG*Ucorr)/(1/C)*
*= (MG *MH(-A)/MG)*(MG*MP(-B)/MG)*(MG*MP(B)*MH(A)/MG)*Uori/C/1/C*
*= (MG*MH(-A)*MP(-B)*MP(B)*MH(A)/MG)*Uori*
= Uori

In order to prevent the user from making mistakes by chaining amplitude/misalignment/trim corrections, this is therefore only allowed on original data files that have not undergone any corrections.
Note that for experienced users, an external function to CASCADE *(f_VGcor_reinit.m) is provided in the tools subdirectory and allows you to re-correct an already corrected file. This function should be handled with care. It is only useful when a user retrieves a* CASCADE*-corrected NetCDF cruise file and wishes to return to the original ADCP speed data.*

3.3.3.7 - Adding tide

This step allows to add, in the working file, the variables corresponding to the components of the absolute current speed, corrected for the barotropic tide.
In the NetCDF cruise file, are added the variables:

- horizontal tidal velocities (U_TIDE et V_TIDE)
- tidal transports (TU_TIDE et TV_TIDE)
- tide-corrected current velocities (UVEL_ADCP_CORTIDE and VVEL_ADCP_CORTIDE)



Interface

The calculation programs implemented in CASCADE were retrieved from the Oregon State University site and converted from fortran90 to MATLAB. The tide files, binary data files (tpxo.tar.Z; do not take the NetCDF file), as well as references to cite are available on the Internet:
http://volkov.oce.orst.edu/tides/global.html

The user has to retrieve the tide model files himself and to drop them in the directory of his choice. This directory must then be indicated in the configuration interface as the tide directory. By default, CASCADE offers the user only the tide models located in this directory.
Tides are provided as harmonics: 8 associated with the elevation of the sea surface relative to land (M2, S2, N2, K2, K1, O1, P1, Q1), 2 long periods (Mf, Mm) and 3 non-linear (M4, MS4, MN4), on a full resolution global grid 1440x721 at ¼. Tide prediction uses harmonic decomposition. Knowing the amplitude and phase of the most important tidal components, it is then possible to reconstruct the tidal signal very accurately and at any time, past or future. **Note that in CASCADE, by default, only the first 11 harmonics are used**.
Each latest model version is normally of better quality than previous versions, since:

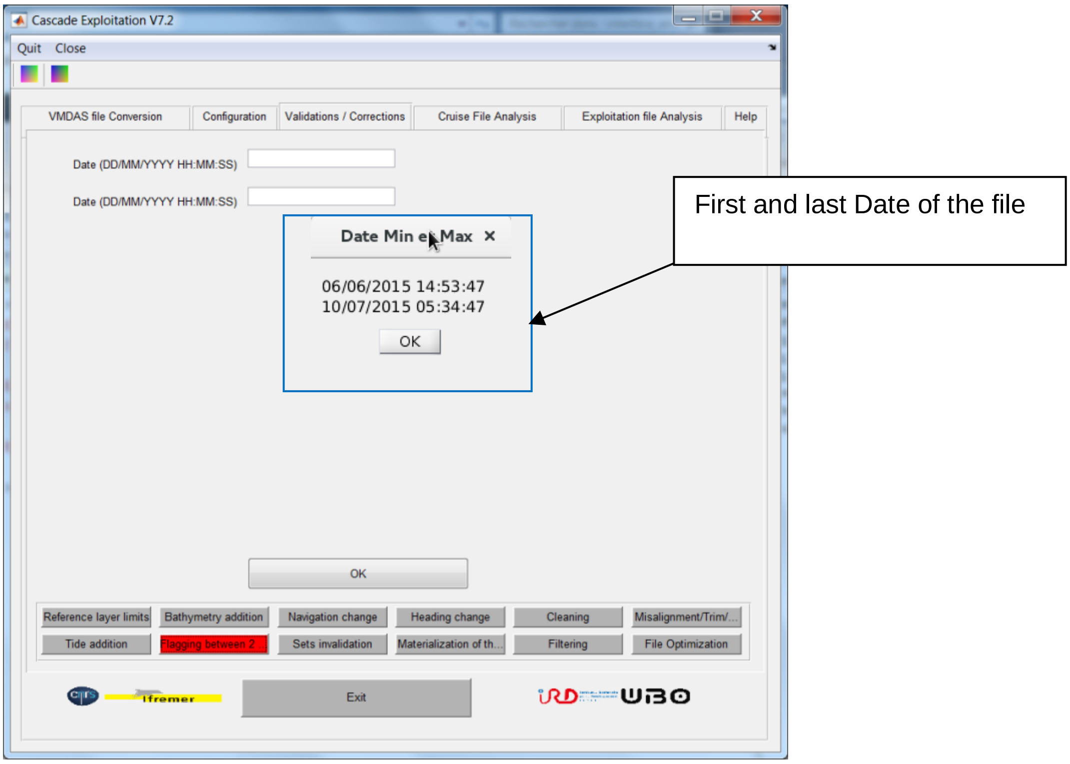
- It assimilates longer time series,
- The bathymetry improves from version to version,
- The global and local grids's resolution improves from version to version.

Be careful: Once these models are retrieved, the user must change the path of the files in the Model_tpxoX.X file.

Note: Caution should be exercised when assessing the tide near the coast (in shallow areas). This is because the tidal model is based on transport and determines tidal velocities from the bathymetry of the tidal model, which is not sufficiently resolved for these areas.

3.3.3.8 - Data invalidation between 2 dates

In some cases, one or more measurement periods may be eliminated, either because they are not appropriate for the scientific purpose or because the data have not been satisfactorily qualified by the criteria available in CASCADE. This step will invalidate these data.



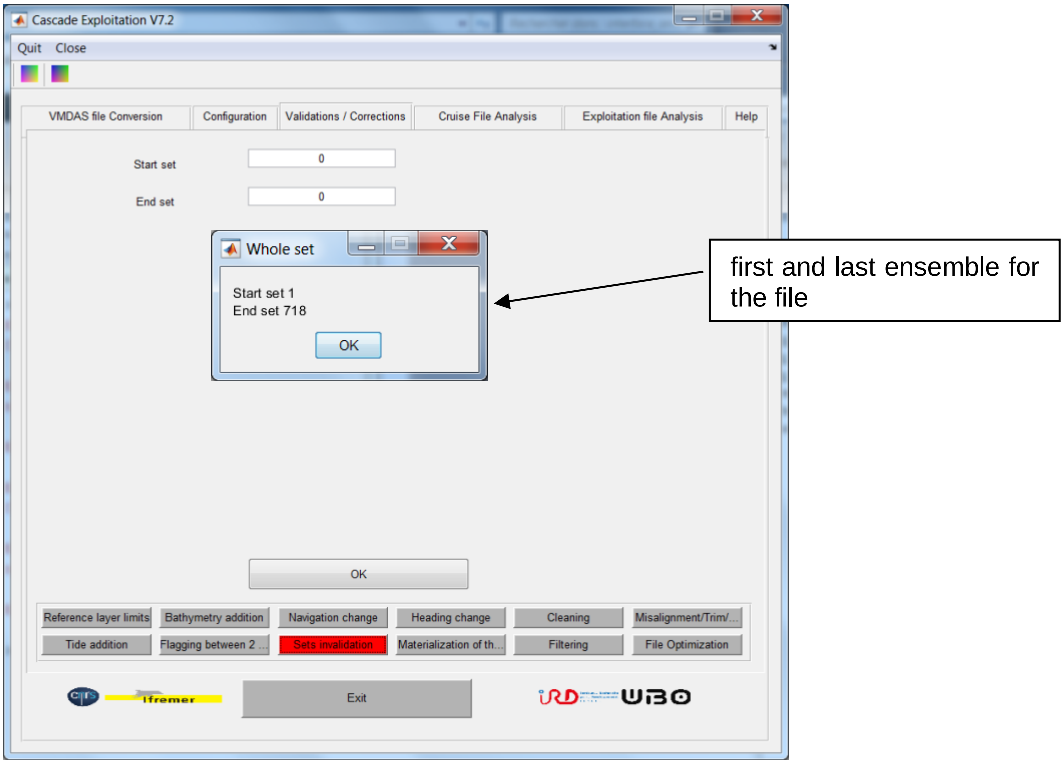
Interface

To invalidate the data (and assign a quality flag set to 9), proceed as follows:

- ○ The graph of horizontal current velocities (associated with flags 1 and 2) versus ensemble numbers is displayed. By placing the mouse pointer on the plot, the dates and time corresponding to the selected ensemble are displayed on the figure. It is possible to zoom in until you can click on the first ensemble to be invalidated. In this case, the date is no longer displayed on the plot; to make it reappear, simply click on the "Data Cursor Mode" button in the figure's menu bar.
- ○ When the first ensemble to be invalidated is chosen, position yourself in the main CASCADE window and enter the ensemble's date and time in the correct format (dd/mm/yyyy hh:mm:ss).
- ○ The two previous operations must be repeated to fill in the last date's ensemble date to be invalidated.

- After checking these dates, click on Validate to invalidate the data between these 2 dates; a message confirming the invalidation appears. **Be careful, if by mistake, the user has invalidated ensembles that had to be kept, the only way to revalidate them is to start again the data cleaning**. Indeed, at each cleaning, the first thing done is to reset the flags to 1. Flags >1 are set according to the different cleanup criteria. To exit without invalidating data, select another operation in the dashboard or change tabs according to the desired step.
- The operation must be repeated if several non-consecutive datasets are to be invalidated.

3.3.3.9 - Data invalidation between 2 ensembles

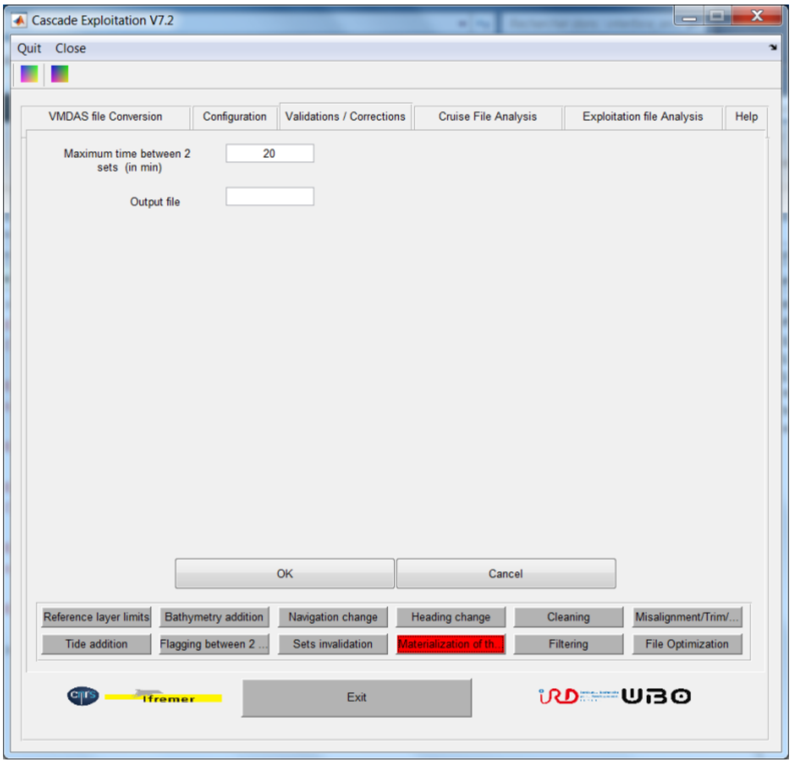


Interface

This step is similar to the previous one except that the horizontal speeds are not displayed and the user must know in advance which ensembles he wants to disable. The user indicates the beginning and end ensembles of the period he wants to invalidate, then he clicks on Validate. A invalidation confirmation message appears. The data associated with this period have a quality flag set to 9. **Be careful, if by mistake, the user has invalidated ensembles that should have been kept, the only way to revalidate them is to start a data cleaning again**. To exit without invalidating data, select another operation in the dashboard or change tabs according to the desired step. The invalidation operation must be repeated if several non-consecutive data sets are to be invalidated.

3.3.3.10 - Materialization of periods without measurement

This is to deal with the case where data acquisition has been stopped for a certain period of time during the cruise. This module will insert 3 empty profiles (containing NaN in the data) in the working file, in order to visualize by a white profile the periods too important without data on the plots.

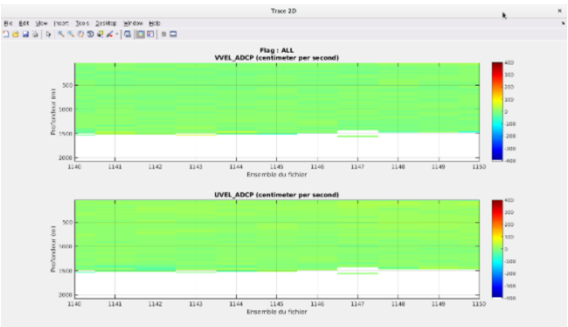


Interface

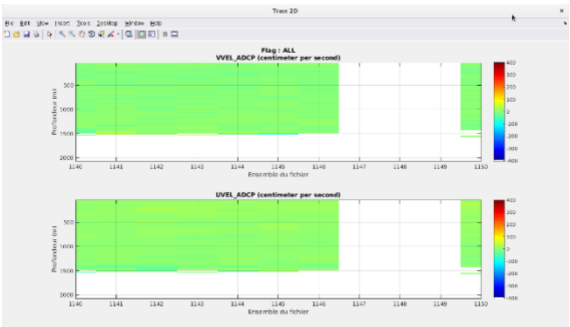
The user indicates:

- the maximum time, in minutes, between 2 data that can be pasted on the plots. 2 consecutive data in the file being spaced by more than this duration will be materialized, on the tracks, by an empty vertical line (white). The indicated duration must imperatively exceed the ensemble's duration. That is why we recommend a value of at least 15 minutes
- the output filename. Warning: If the user wishes to work on this file afterwards, he has to go through the configuration interface again to indicate the new working file.

The following plots show the effect of this materialization of period without measurement. In this example, there are 21 minutes between set 1146 and 1147. The user wants a materialization of periods without measurement from 20 minutes onwards. The program inserts 3 empty records between set 1146 and 1147 (The number 3 is explained by the fact that the filtering (see 3.3.3.11) can only fill a maximum of 2 gaps. Thus, even after filtering, this absence of data of more than 20min will remain visible on the data plots).



Before



After

3.3.3.11 - Filtering

This option allows the user to filter the absolute current velocity data (u,v,w) from the working file, after cleaning, to reduce noise and fill isolated holes. If tide-corrected velocities (u,v) exist, they are recalculated from the filtered velocities.
The user can choose between 3 filterings:

- Horizontal, i.e. time-dependent filtering
- Vertical, i.e. depth-dependent filtering
- Horizontal and vertical filtering

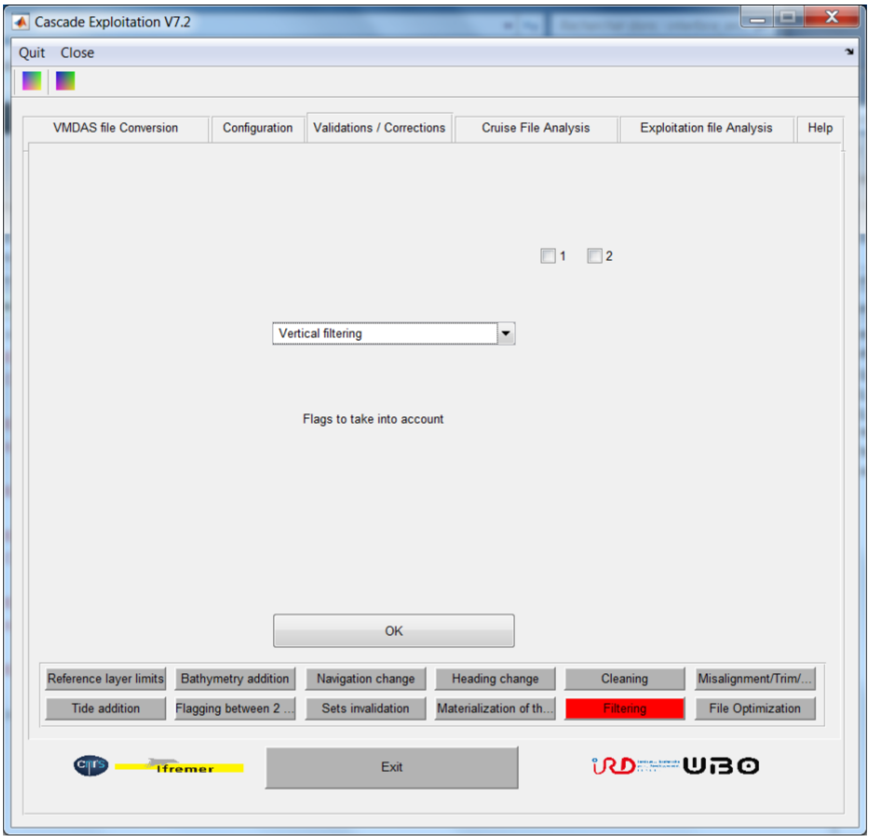
Only good data and/or doubt data (flag 1 and/or 2) can be taken into account for filtering. Flags are filtered in the same way as speeds, except that they are filtered at the end of the program:

- the flag is rounded up to the next whole number.
- a flag 7 is affected to missing data. This corresponds, in the original file, to missing data and bad data (flag > 1 or 2) not filled by the filtering.
- a flag 2 is affected to the data generated by the filtering. This corresponds, in the original file, to missing data and bad data (flag > 1 or 2) which have been filled by the filtering.
- a flag 8 is affected to the data under bottom.

So, the filtered file contains only flags 1, 2, 7 and 8.

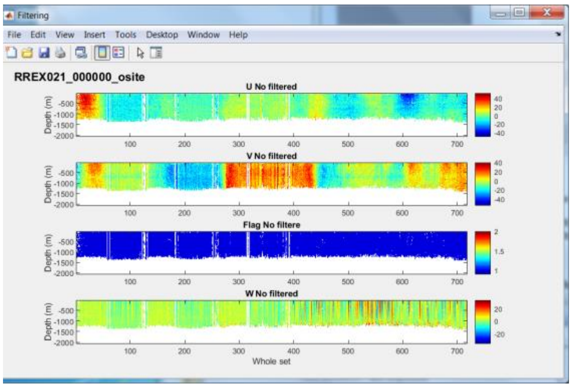
Filtering is performed on absolute current speeds and the associated flags are modified. Filtering is a sliding average over 3 points (1/4 [previous value] +1/2 [current value] +1/4 [next value]), except on the edges where the filtered value is calculated over 2 points (2/3 [current value] + 1/3 [next or previous value]). Thus, the maximum gap length filled by the filtering is 2, and the data created by the filtering is flagged to 2.

Initial values	Filtered values
A	2/3 A + 1/3B
B	0.5A + 0.5B
NaN	B – Flag to 2
NaN	NaN
NaN	C– Flag to 2
C	0.5C + 0.5D
D	0.25C + 0.5D + 0.25E
E	0.25D + 0.5E + 0.25F
F	0.5E + 0.5F
NaN	0.5F + 0.5G
G	0.5G + 0.5H
H	0.25G + 0.5H + 0.25I
I	0.25H + 0.5I + 0.25J
J	0.5I + 0.5J
NaN	J
NaN	K
K	K

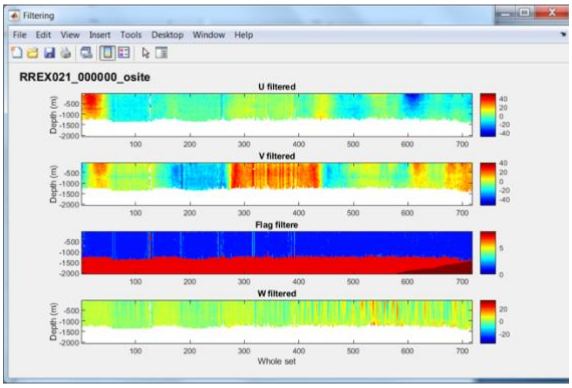


Interface

The two 2D graphs are displayed on the screen (in the example, only flags 1 have been taken into account for filtering):



Before



After

A new cruise file is created in the ncc subdirectory with the following nomenclature:

<filename>_f< filtering_type ><flag>.nc

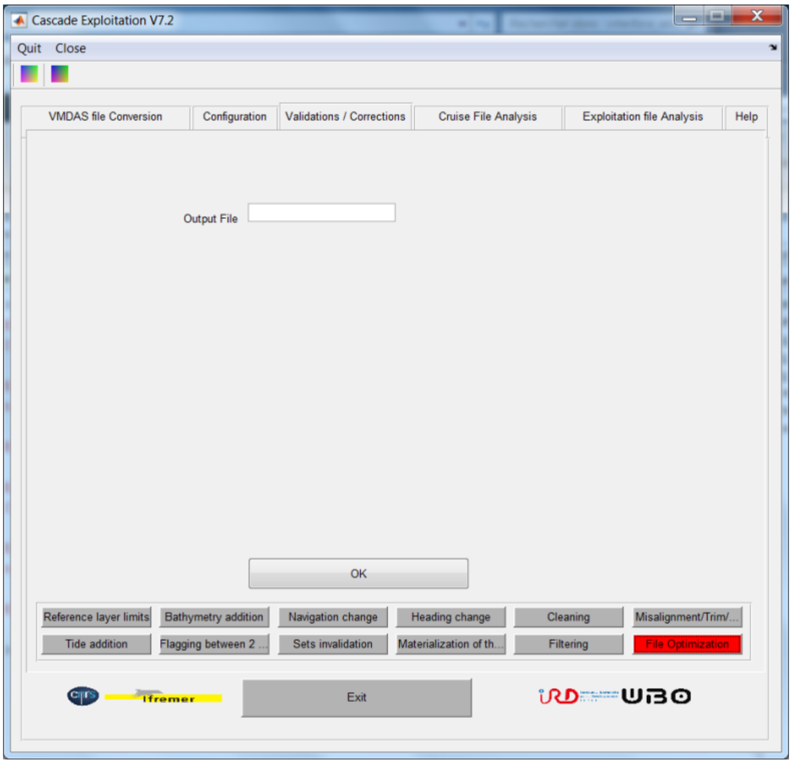
f for filtered.
the filtering type is :

- h for horizontal filtering
- v for vertical filtering
- vh for both (horizontal and vertical)

The flag indicates the list of flags taken into account for filtering. This new cruise file can then be used as a working file by selecting it in the "Configuration" tab.

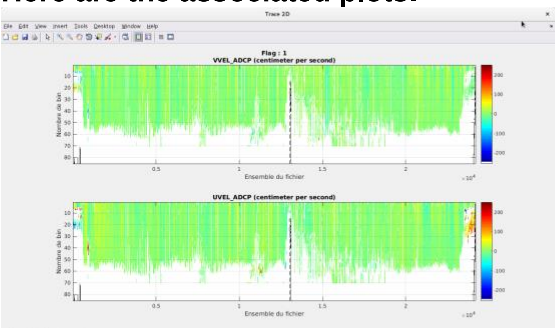
3.3.3.12 - File Optimization

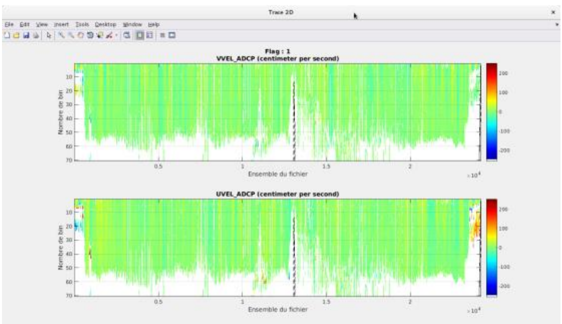
This step deletes the last data cells for which no valid absolute current velocity data exists (flag > 2). This can happen, for example, when the user has configured too many cells in relation to the ADCP frequency, so that the ADCP cannot reach the depth associated with the last cells. The last cells are then deleted.
For this step, the user specifies the output filename. This file contains the same data as the working file but only on the 'useful' cells.



Interface

Here are the associated plots.

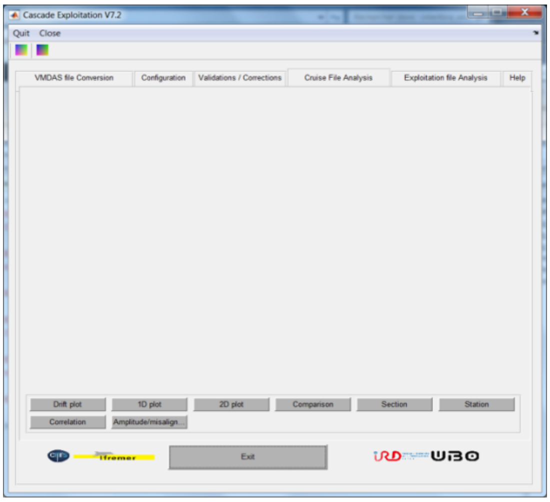




Before **After**

The initial file consists of 85 cells while the optimization file consists of 70. The file's size is thus reduced (192Mo for the file before optimization, 159Mo for the file after optimization).

3.3.4 Cruise file analysis



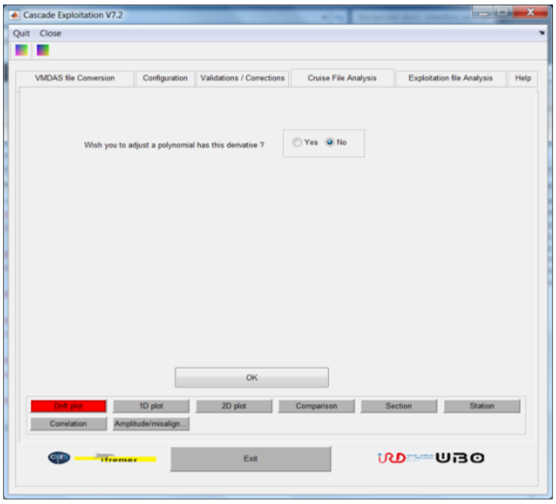
Interface

Some important points to check at this stage:

- In « 1D plot » (3.3.4.2), check that:
  - the JULD date is monotonous and increases with the ensemble's number
  - LAT and LON variables plot the expected ship's track
  - the variables HDG (heading), PTCH (pitch) and ROLL (roll) have been recorded correctly.
  - the number of pings per ensemble (NB_ENS_AVE) is the expected one
  - the bathymetry (in blue when the variable BATHY is plotted) is compatible with the bottom detected by the "bottom ping" if it has been activated (in black on the figure).
- In « 2D plot » (3.3.4.3):
  - Plot together good data percentage (PGOOD_ADCP), echo intensity average (ECI), average correlation of the reflected signal (CORR) and flags (CAS_CURRENT_FLAG); these will be displayed in a separate window. These two figures are used to analyze data quality and cleaning efficiency.
  - Next, plot the vertical speed WVEL_ADCP and the error on this speed EVEL_ADCP on flags 1. A high vertical speed associated with higher and less random errors shows that the cleanup did not remove everything. Vertical bands of high WVEL_ADCP indicate a trim problem (correction proposed in 3.3.4.8).
- To estimate an alignment or amplitude problem
  - If bottom-ping data is available (even sporadically), apply a Uship/Ubottom comparison (cf. 3.3.4.4.4)
  - Otherwise, use the estimate explained in 3.3.4.8, which will only work if the vessel has often accelerated and decelerated but will still be much less accurate than the previous method.
  - If the alignment, amplitude and pitch values are significant and credible, apply them (see 3.3.3.6), then select the new file in the Configuration tab and start the cleanup again. The same analysis steps can be performed on the new file corrected for misalignment, amplitude and pitch.
- In general, as soon as the data have been qualified, do not hesitate to compile a cruise information file (cf. 3.3.6.2) to check the analysis's progress.

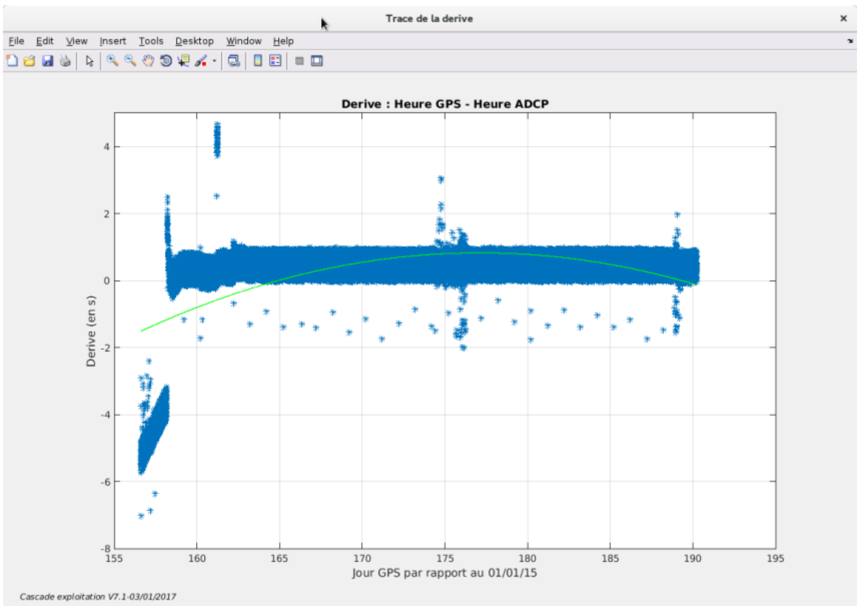
3.3.4.1 - Drift plot

This step, only available for data acquired with old onboard ADCPs and processed with processing CASCADE, allows to visualize the clock drift of the ADCP acquisition PC during the cruise in relation to the time resulting from navigation (GPS time).
Note that this drift is plotted during the conversion of the STA/LTA files for the OceanSurveyor onboard ADCP data acquired with VmDAS (cf. 3.3.1).



Interface

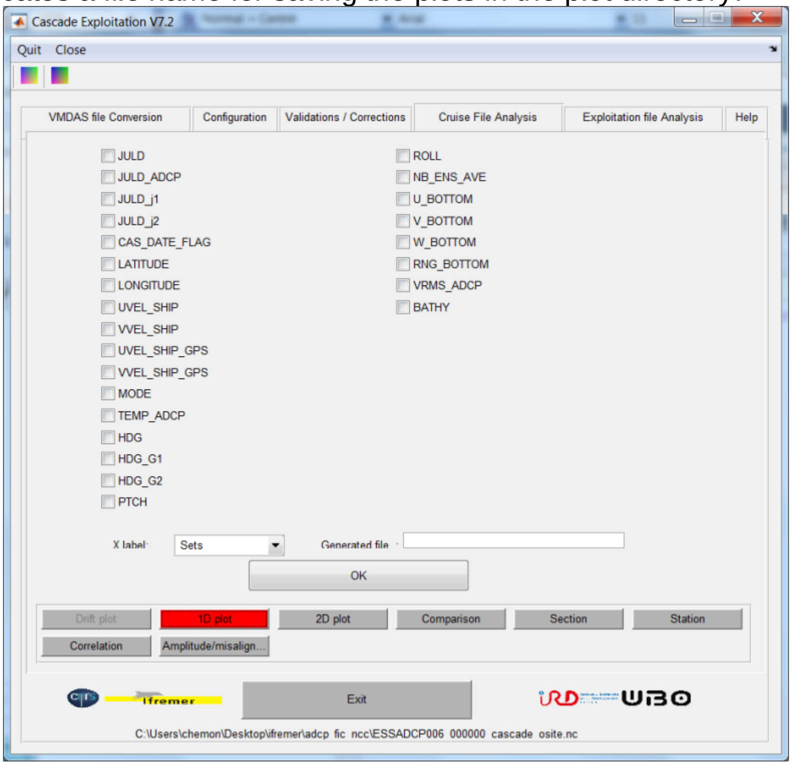
The user can, if he wishes, estimate a polynomial which passes at best of all the points of the drift. He must then specify the degree of this polynomial. This polynomial is not used afterwards. This step is just a visualization step.
Below, the plot associated with this step:



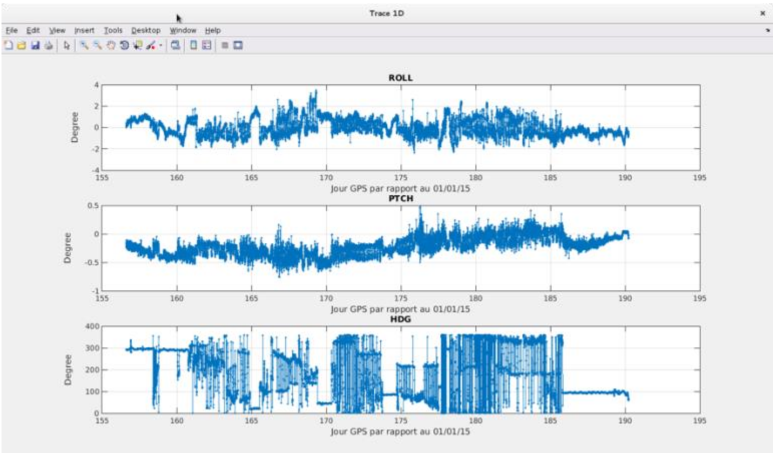
It is recommended to regularly set the acquisition PC time in relation to GPS time. One should thus generally have a quasi-constant drift around 0 (within a few seconds). If this is not the case, the VmDas acquisition software nevertheless manages to make the ADCP transmission time coincide with the navigation associated with this same time.
The working file is not modified by this step.

3.3.4.2 - 1D plot

This plot allows to plot the variables of the working file having as only dimension the time. The user can plot a maximum of 4 variables on the same plot. The X-axis can be the ensemble's number or the day in the year.
The user also indicates a file name for saving the plots in the plot directory.



Interface

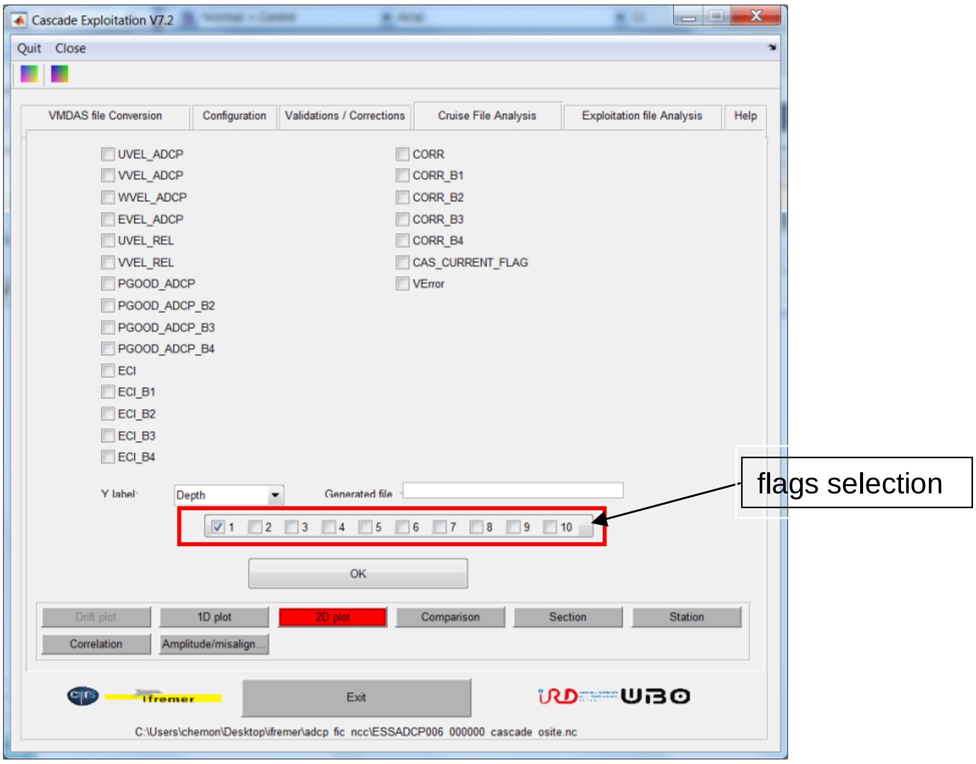


Example: roll/pitch/heading

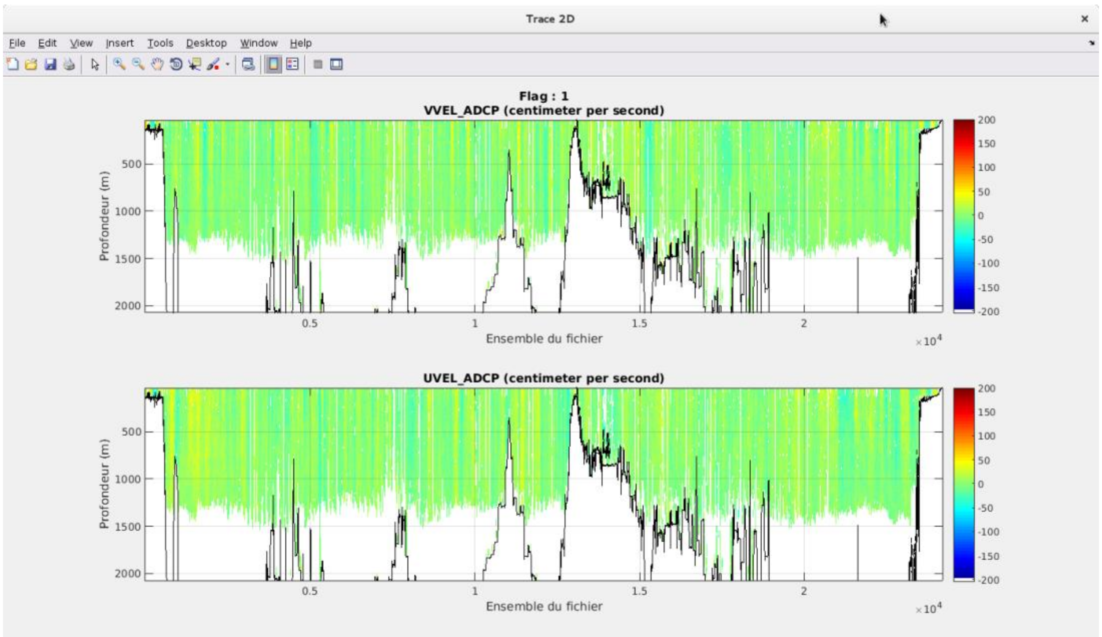
If the user plots the latitude and longitude, an additional plot showing the ship's track is automatically generated.

3.3.4.3 - 2D plot

This plot allows the user to plot the variables having 2 dimensions (1 dimension time and 1 dimension depth). On the plot, the X axis is the ensemble's number, the Y axis is either the depth, either the cells ('bins'). The user can plot a maximum of 4 variables on the same graph. He chooses whether he decides to plot all the data or only those assigned to some defined flags. He also indicates a file name for saving the plot in the plot directory.



Interface



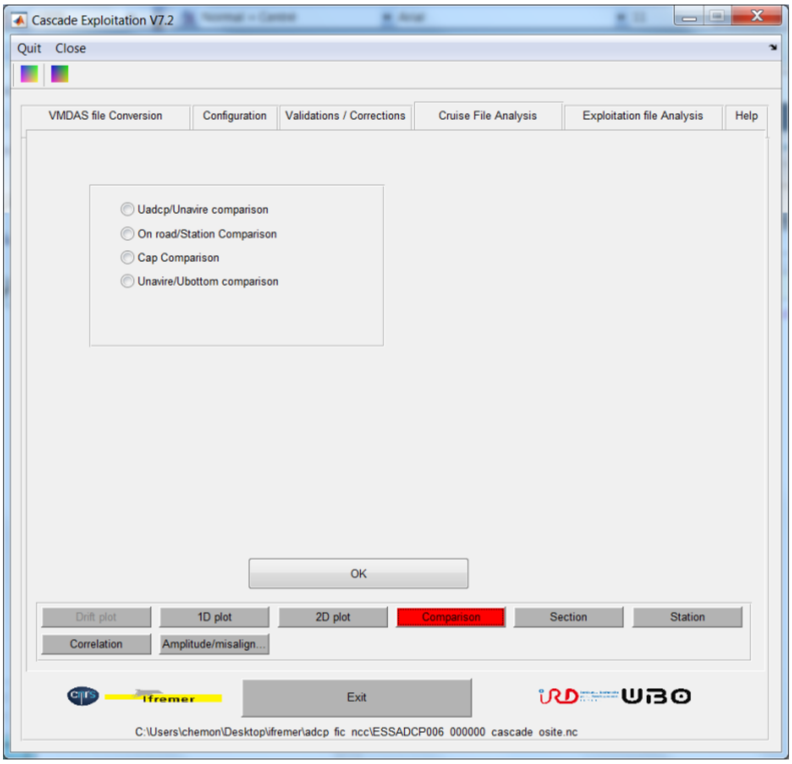
Example: absolute ADCP. X axis: ensembles, Y axis: depth

Notes:

- The user can check or uncheck all flags at once by clicking on the outline of the flags selection.
- the CAS_CURRENT_FLAG variable (the value of the flags) is displayed on a particular graph.
- The user can zoom in/out on the graphs (Zoom In and Zoom Out in the Tools menu, or directly in the graph toolbar) and change the colors or the limits of the color scale directly in the graph toolbar (Edit/colormap menu) but then it is necessary to re-save the plot manually.
- On the horizontal current velocities graph, the user may find incorrect velocities (sharp vertical lines on the background) or incorrectly invalidated. By viewing the flags in parallel, the user can decide whether to be more or less severe for the various cleaning criteria. If none of the cleanup criteria can invalidate visibly incorrect velocity profiles, then the user should invalidate them by date or ensemble.
- When the user plots one of the absolute current velocities (meridian and/or zonal), the current velocity is also presented as a modul and direction.

3.3.4.4 - Comparison

The Comparison interface allows the user to compare current velocities with ship velocities, on the way and stationary velocities, different available headings as well as bottom-pinged ship velocities and ship's velocities from navigation. It is a help to qualify the ADCP's alignment and the ADCP's amplitude factor.

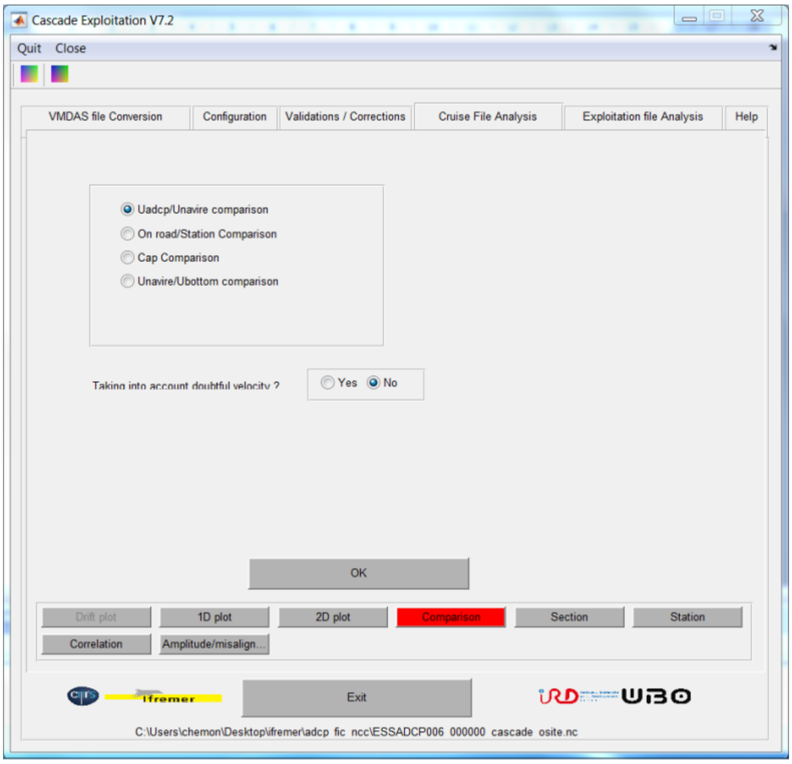


Interface

3.3.4.4.1 ADCP velocities/Ship Velocities

This step helps to check that the current's velocities and the ship's velocities are not correlated. If a clear correlation exists, there is a problem with the current measurements, probably due to a ADCP misalignment, involving a projection of the ship's speed onto the current speed. In this case, an acceleration or deceleration of the ship has an effect on the current velocity. Similarly, a change in the ship's heading may then influence a change in the direction of the current. This step allows to visualize if the changes in the ship's speed have an effect on the current's speed. If this is the case, the speeds are correlated and it is necessary to correct the ADCP misalignment (see possible estimates in 3.3.4.4 and 3.3.4.8).

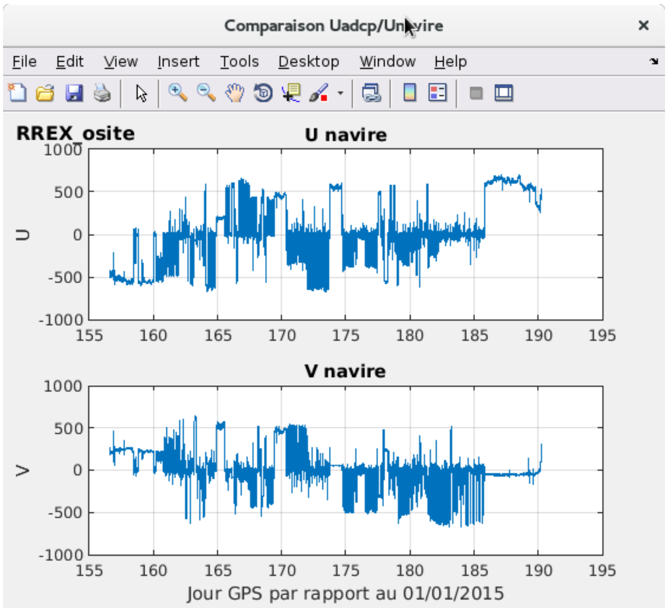
The user indicates whether he wishes to take into account the doubtful data (flag 2) or not for this check. Otherwise, only flag 1 data are considered.



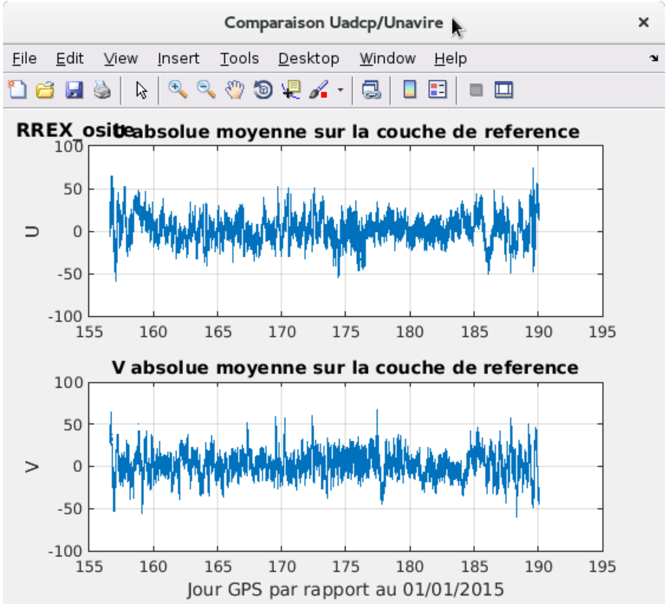
Interface

3 plots are displayed:

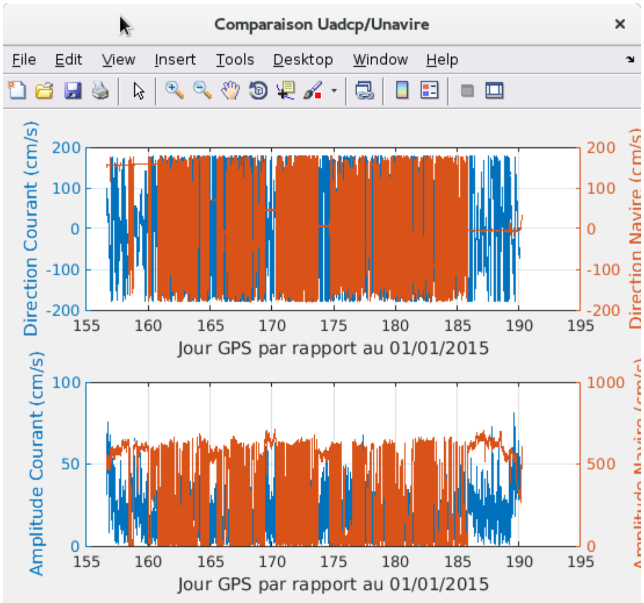
- One with ship's velocities
- One with ADCP's velocities, averaged on the reference layer
- One with the both on the same graph (with a scale factor for the ship's velocities so that the both can be compared)



Ship's velocities



ADCP velocities



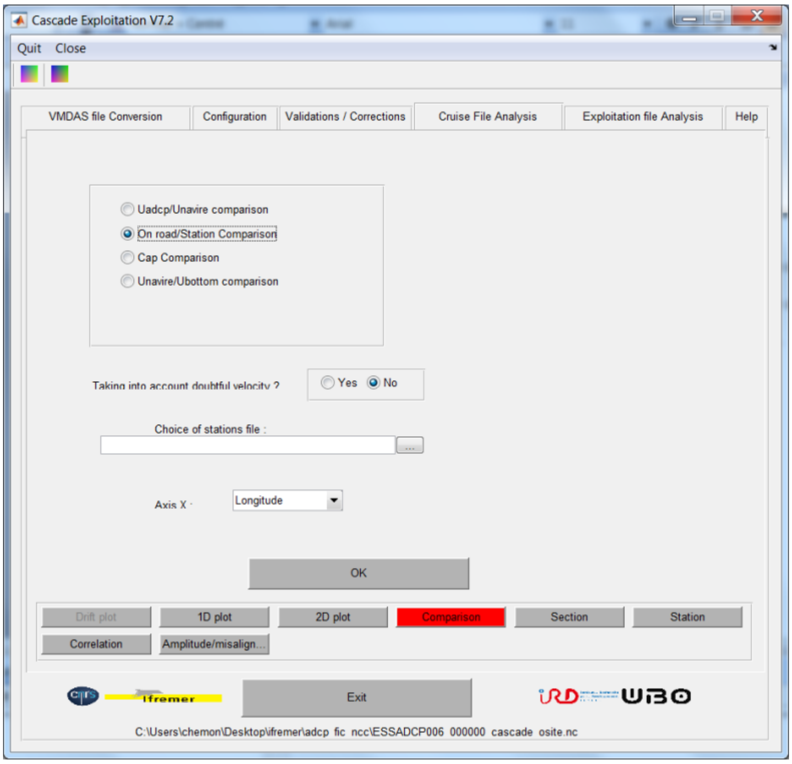
ship's velocities compared to ADCP's velocities

These graphs help to show a possible correlation between the ship's velocities and the ADCP velocities, which the user can then correct by applying a ADCP misalignment.

The plots are automatically saved in the plot subdirectory: *<filename>_cmp_Uadcp_Vnav.<.ext>.*

3.3.4.4.2 Underway/Station

The underway current velocities data are compared with the station current velocities data. The velocities are averaged over the reference layer.
Absolute velocities averaged over the station reference layer should approximate the absolute on the way velocities around the station. If this is not the case, it means that the ship's velocity influences the current velocitiy (the ship's velocity is almost zero at the station, it is non-zero on the way). In this cas, the absolute current velocities must be corrected by a misalignment or an amplitude factor.

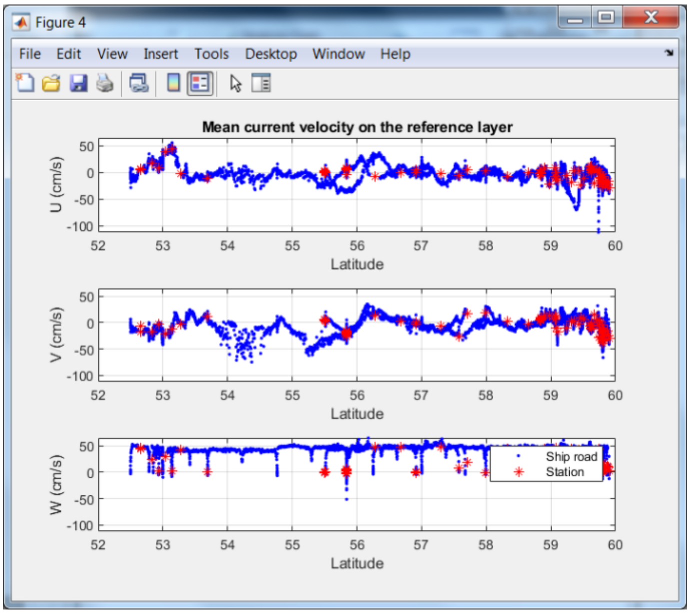


Interface

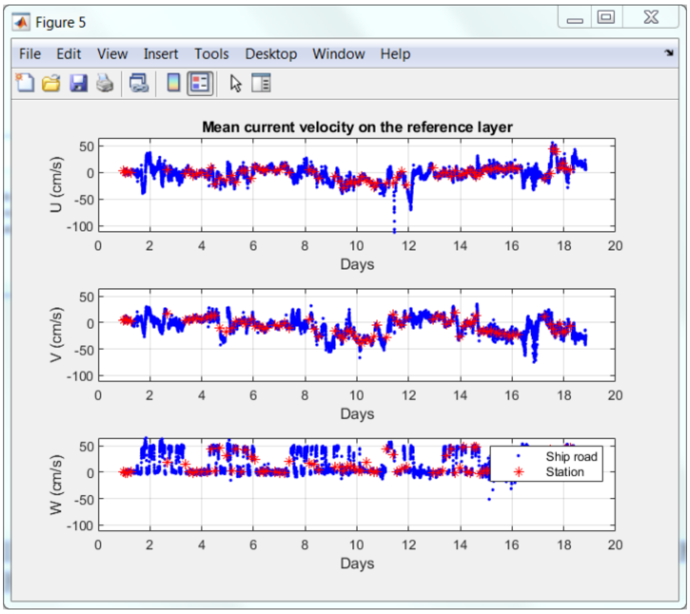
The user enters the Station List filename to be considered. He indicates if he wants to consider doubtful data or not as well as the X axis (longitude or latitude according to the ship's track).
2 plots are generated:

- 1 comparing on the way current velocities averaged over the reference layer with those at stations, with a geographical x-axis
- 1 comparing on the way current speeds averaged over the reference layer with those at stations, with time (days) on the x-axis

The plots are automatically saved in the plot subdirectory: <filename>_cmp_sta_route*.<ext>.



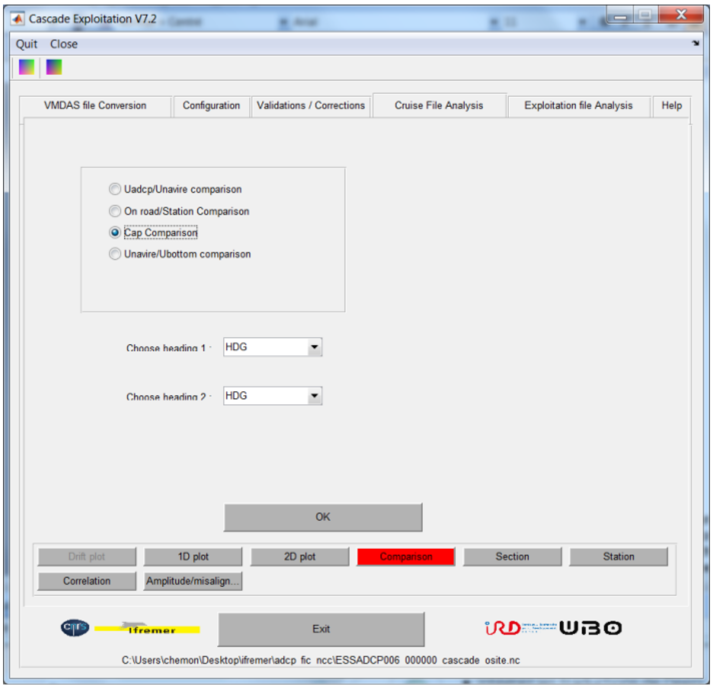
velocities comparison: Xaxis: latitude



velocities comparison: Xaxis: time

3.3.4.4.3 Heading

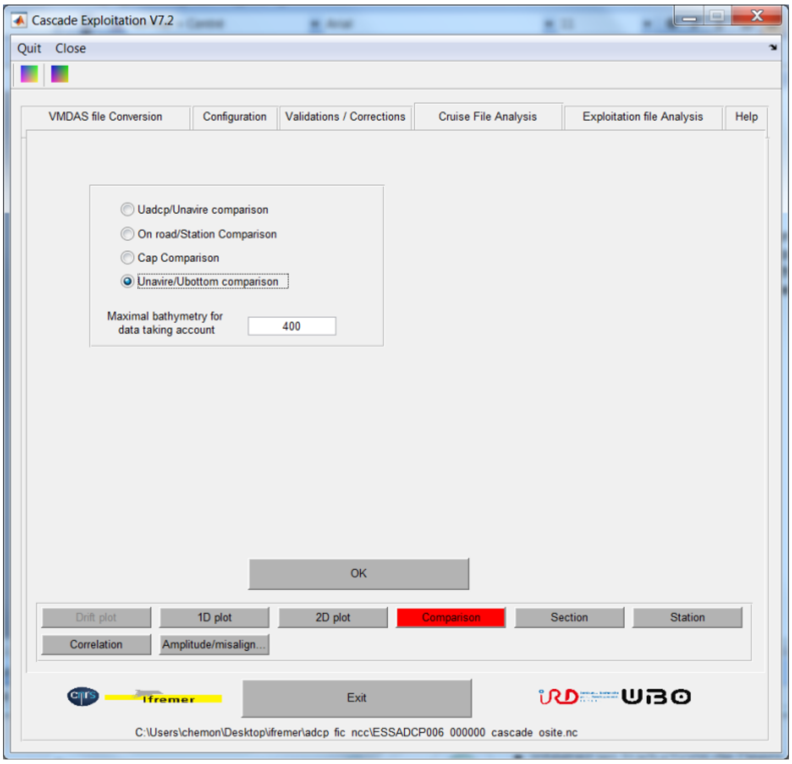
The user can compare the different heading existing in the working file. Thus, he can decide which heading is the best and recalculate the absolute current velocities accordingly (see 3.3.3.4). This comparison is not possible with STA or LTA files that contain only one heading.
The plots are automatically saved in the plot subdirectory: <filename>_comparaison_cap.<ext>.



Interface

3.3.4.4.4 Uship/Ubottom (amplitude/misalignment: bottom track method)

The ship velocities's modul and angle are compared with bottom-ping velocities's modul and angle. The bottom-ping measures the speed of the bottom relative to the ship. As the bottom does not move, it measures the opposite of the ship's speed if the amplitude factor and the alignment are correct. The modulus and angle of the ship's speed and the bottom-pinged speed should be nearly similar.

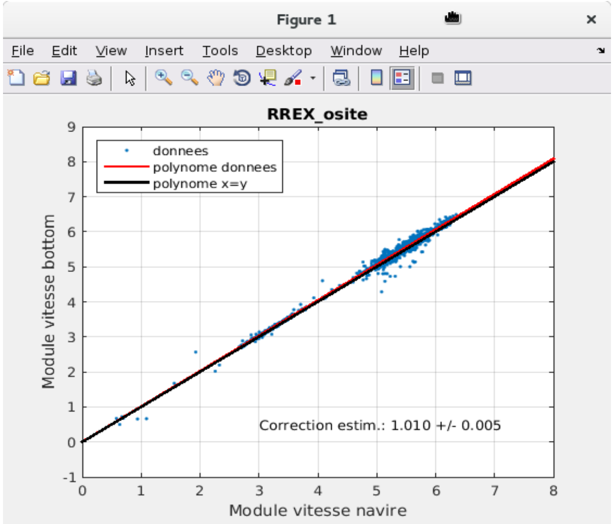


Interface

For comparison, only data associated with a bottom less than or equal to a depth indicated by the user are considered. Due to the orientation of the beams, the deeper the bottom is, the further the beams are horizontally spaced, which may generate an incorrect bottom speed estimate if the beams refer to different depths (e.g. at the edge of the continental slope). The bottom velocity estimate is better on a flat bottom, so we recommend to restrict the data associated with a shallow bottom if possible (400m).
On these speeds, information on the average of the differences between the ship's speed and the current's speed is provided in U and V, as well as the associated accuracy (accuracy = standard deviation divided by the square root of the number of points).
To estimate the amplitude, the selection criterion is restricted by keeping at the end, only the data associated with a bottom less than or equal to the depth indicated by the user and associated with a ship speed greater than 1 knot in order to increase the signal to noise ratio.
To estimate the misalignment, the selection criterion is restricted by keeping at the end only the data associated with a bottom less than or equal to the depth indicated by the user and associated with a speed greater than 2 knots and for which the difference in angle between navigation and bottom-ping does not exceed 2° (which implicitly means that the ADCP/ship angle implemented via VmDas is already relatively correct).

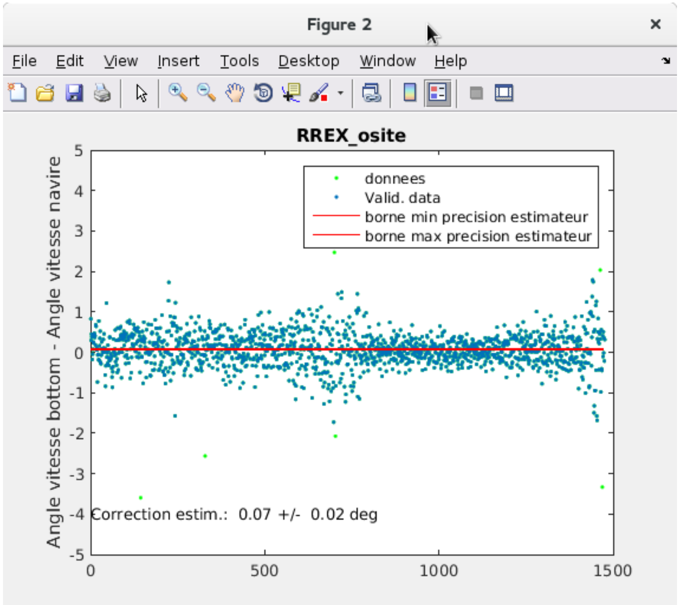
2 plots are generated:

- Figure 1 shows the modul of bottom speed relative to the vessel as a function of the vessel's speed modul. If there is no amplitude problem, then the data should cluster around the x=y line. If this is not the case, the user can correct this by applying an amplitude, <1 if the bottom speeds are smaller than the ship speeds, and vice versa. If such a difference exists, it is usually confirmed by a correlation between the parallel current velocity to the track and the vessel speed in the cruise file information (based on the calculation explained in 3.3.4.7).



The amplitude A to be applied proposed by CASCADE, with a confidence interval of 95% (2 standard deviations divided by the square root of the number of data) is estimated from P(1) with P is the polynomial: bottom_velocity = A * ship_velocity + P(2). P is determined by imposing a strong constraint so that the original ordinate is close to 0: if X data are used, X data with a ship speed and a bottom speed at 0 are artificially added, in order to constrain P(2).
In the example above, CASCADE suggests an amplitude correction of 1.01, +/- 0.005 for a 95% confidence interval. This means that the current velocity is overestimated, relative to the ship's speed, by a factor of 1.01. In the misalignment/trim/ amplitude correction, the current velocities will then be divided by the amplitude value provided by the user (i.e. 1.01 for this example in order to decrease them). Note that the +/- value is a precision value. If it is higher than (A - 1), then the correction is not reliable and it is advised not to apply it.

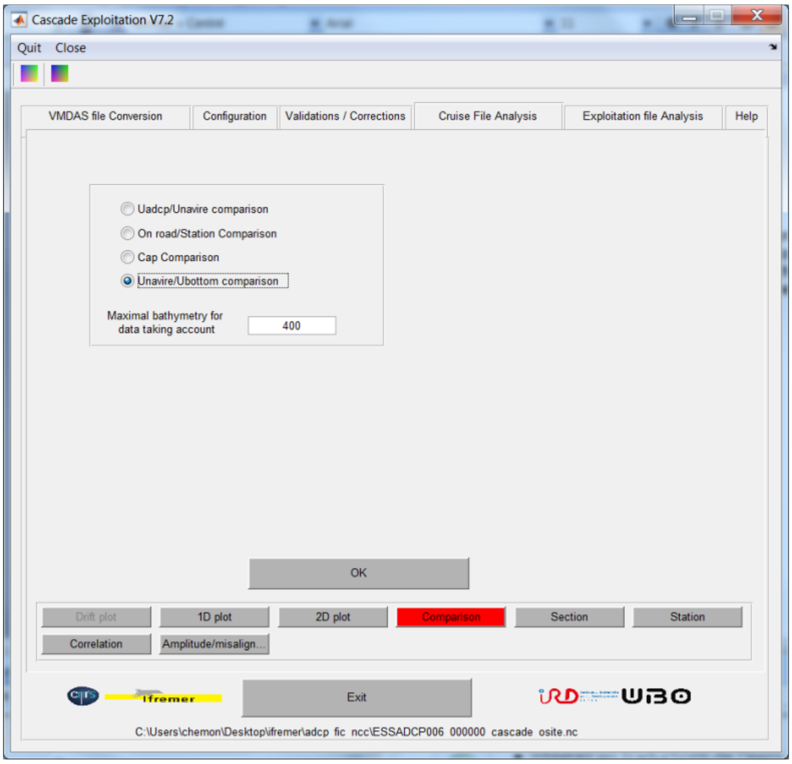
- Figure 2 shows the difference in direction between the opposite of the speed of the bottom relative to the vessel and the vessel's speed. The possible misalignment angle to be applied is estimated from the average of this difference over the data considered (as explained above) represented by a 95% confidence interval on the graph. If the average angle is significant, a correlation between the perpendicular current speed to the track and the ship's speed must be found in the information on the cruise file (according to the calculation explained in 3.3.4.7).



In the example above, CASCADE suggests a misalignment of 0.07°, +/- 0.02 for a 95% confidence interval. This means that the current speed's angle is 0.07° greater than the angle of the ship's speed. In the Misalignment/trim/Amplitude correction, the current velocities are modified by applying a misalignment value of 0.07°. Note that the +/- value is an accuracy value. If it is greater than the proposed misalignment, then the correction is unreliable and should not be applied.
Based on these graphs and the estimation of the amplitude and misalignment correction, the user can correct the absolute current velocities through the application of an amplitude and/or misalignment correction (cf.3.3.6).

3.3.4.5 - Creating a section file

This step creates a Section file. A section corresponds to a consistent ship's track. During a cruise, if there are significant changes in the ship's direction, or changes in the environment, several sections can be created to represent the data.



Interface

The user indicates:

- In the "Section File" field, the Section List filename < filename>_sec.list that the user must first create manually. This contains one line per section indicating the section number as well as the start and end dates and times of the section.
  - Section number jj/mm/yyyy hh:mm:ss jj/mm/yyyy hh:mm:ss
    - 01 12/06/2016 12:23:00 20/06/2016 14:40:00

To determine the start and end dates of a section more precisely, one can use the track plot when selecting Latitude and Longitude in Plot 1D (see 3.3.4.2): if "Data Cursor" is active in the figure's toolbar, the date is displayed when clicking on a point.

- In the "Distance (km) between each point" field, the number K of kilometers over which he wishes to average the data. The program starts from the first point (P1) of the section and then looks for the last point (P2) on the section less than K kilometers away. The data is averaged over all points P1 to P2. This average is assigned to the temporal and geographical average between P1 and P2. The program applies the same principle from the point following P2 and so on. Note that if the section has a hole in the data greater than 2*K kilometers, the program creates an empty record to visualize, graphically, this absence of data on the section plots: the program does not create data by interpolation. Thus, in the Section file, the points are not necessarily spaced K kilometers if data is missing.

- If he wishes to take into account tide-corrected or uncorrected data
- if the station data must be or not taken into account.
- In the case where station data should not be considered, the user must indicate the Station List file so that the associated data will be ignored. The Station List file is an ASCII file < filename>_sta.list with one line per station indicating the station number and the start and end dates and times of the station.
    - Station number jj/mm/yyyy hh:mm:ss jj/mm/yyyy hh:mm:ss
        - 01 12/06/2016 14:50:15 12/06/2016 17:58:03
- The flags to be considered

This step creates a Section file in the nce subdirectory. The nomenclature of the created file is as follows :

<filename>_sec_X |*m* *s*
|*x* *x* *<flag>.nc*

X: number of kilometers between each point.

|*m* tide taken into account: tide removed from the signal
|*x* tide not taken into account

|*s* stations included
|*x* station removed (not taken into account in the average over X kilometers)

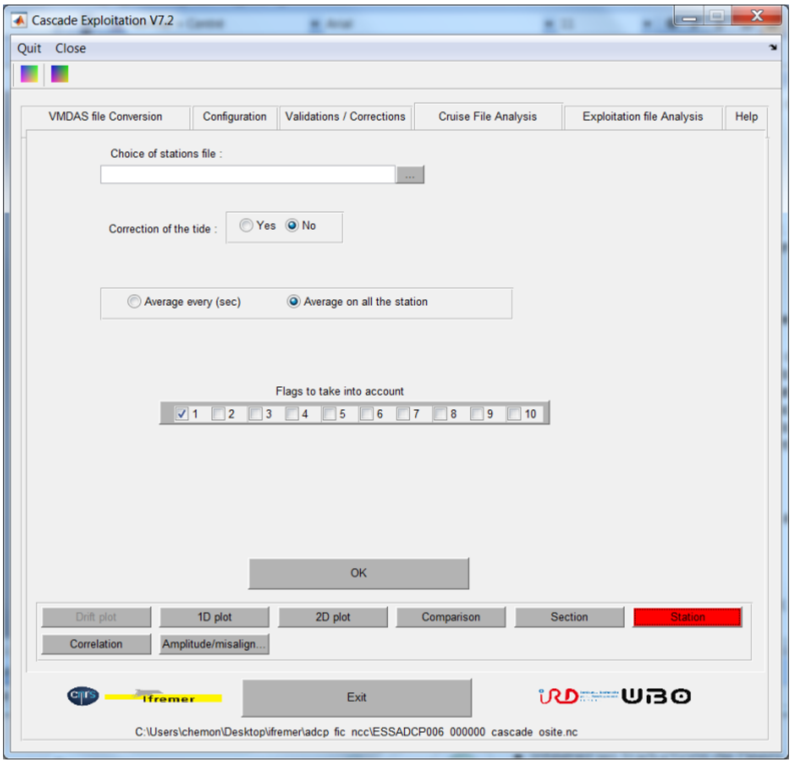
<*flag*> flags taken into account

Example*: ovide_osite_sec_02ms1.nc* for a section file averaged on 2 km including station et with removed tide. The section file has been generated with the good data (flag 1).

In the file, the sections are put end to end, and the variable INDICE indicates the beginning of the section. Thus, the data corresponding to section number N will be between INDICE(N) and INDICE(N+1)-1 for all the variables.

3.3.4.6 - Creating station file:

This step creates a Station file for the cruise. A station corresponds to a time where the ship remains on the same place (in relation to the bottom) for a particular operation (launching of instruments for example).



Interface

The user indicates:

- In the "Station File" field, the Station List filename to be created manually by the user. It must contain one line per station indicating the station number as well as the start and end dates and times of the station.
    - station number jj/mm/yyyy hh:mm:ss jj/mm/yyyy hh:mm:ss
        - 01 12/06/2016 14:50:15 12/06/2016 17:58:03
- If he wishes to take into account corrected or uncorrected tide data (if the tide has been added to the cruise file)
- If he wishes to average the data over a defined time TPS (to be specified in seconds) or over the whole station. If averaging over the whole station, the resulting NetCDF station file will include one record per station. This will include the arithmetic average of the data taken into account over the entire station. Otherwise, the result file will include, for each station, as many records as the number of TPS second slots contained over the duration of the entire station. Each record corresponds to the average of the data taken into account during TPS seconds.
- the flags to be taken into account (flag 1-2 in general)

This step creates a Station file in the nce subdirectory. The nomenclature of the created file is as follows:

<filename>_sta_X |m |x <flag>.nc

X: number of seconds used to average, if averaged over the entire station then X = 99999.

|*m* tide taken into account: tide is removed from the signal
|*x* tide not taken into account

<*flag*> indique: flags taken into account

Example*: ovide_osite_sta_00120x1.nc* for a station file where stations are averaged over 2 minutes and the tide is not removed from the signal. This file has been generated with good data (flag 1).

In the file, the stations are placed end to end, and the variable INDICE indicates the start of stations. Thus, the data corresponding to station number N will be between INDICE(N) and INDICE(N+1)-1 for all variables. If averaged over the entire duration of the stations, INDICE will be a vector from 1 to NSTA (the number of stations) in steps of 1.

3.3.4.7 - Correlation

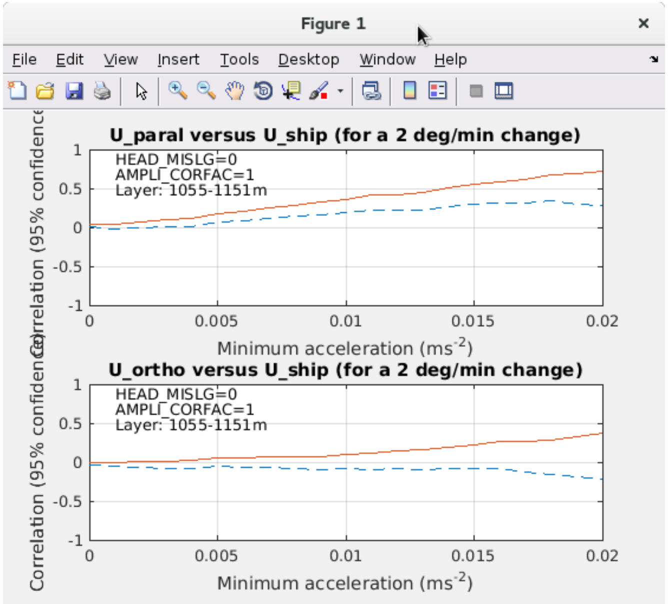
In this step, we study the correlation between ship speeds and absolute current speeds for an independent validation of the alignment and amplitude parameters estimated by the ship/bottom comparison steps (3.3.4.4) or the amplitude/misalignment calculation (3.3.4.8). This step can be performed alone with the "Correlation" button, but it will also be automatically performed when generating the cruise information (see 3.3.6.2) in order to retrieve the correlation values.
The program:

- uses only good data (flag 1) ;
- Calculates the angle of the ship's track from geographical positions ;
- Determines the depth layer of least variability in absolute current velocities ;
- Calculates the mean of the absolute current velocities over the layer of least variability for data where the layer of least variability is more than 50m away from the bottom (so that the layer of least variability is not disturbed by bathymetry) ;
- From these absolute current velocities, the parallel and orthogonal components to the ship's track are deduced ;
- Calculates the ship's track-parallel component from the ship's horizontal velocities (U and V), that means the ship speed's modul ;
- Calculates and plots the correlation with a 95% confidence interval between (i) the track-parallel component of the current's absolute speed and (ii) the track-parallel component of the ship's speed, for data associated with a maximum change of direction of 2°/min and accelerations above a threshold varying between 0 and 0.02 m/s^2.
  If the correlation interval does not include 0 for an acceleration threshold of at least 0.007 m/s^2 , then the velocities are considered to be correlated and the current velocity data must be corrected by applying an amplitude factor, usually between 0.97 and 1.03.
- Calculates and plots the correlation with a 95% confidence interval between (i) the track-orthogonal of the absolute velocity of the current and (ii) the track-parallel component of the ship's speed, for data associated with a maximum change of direction of 2°/min and accelerations above a threshold varying between 0 and 0.02 m/s^2.
  If the correlation interval does not include 0 for an acceleration threshold of at least 0.007 m/s2 , then the velocities are considered to be correlated and the current velocity data must be corrected by applying a misalignment, generally between -1° and 1°.

The accuracy of the calculation will be highly dependent on the number of data satisfying the selection criteria (acceleration and track change), which implies a trade-off between the chosen acceleration threshold and the number of data available. In the results selected for the cruise information file and displayed in the MATLAB command window, a threshold of 0.01 m/s2 is chosen. An example of the results of this step is shown below:

- `Correlation ship vel. / current vel. parallel  to the ship route: [0.18 0.36]`
- `Correlation ship vel. / current vel. orthogonal to the ship route: [-0.09 0.10]`
- `Mean velocity of the current parallel and orthogonal to the ship route, and vertical (cm/s): [-1.75 3.10  2.00] cm/s`

The correlation plot is presented funtion to the ship acceleration threshold used for data selection (example below). The minimum of the confidence interval is shown as a dotted line, the maximum as a solid line. The difference between the two increases as the number of selected data decreases. If the current velocities and the ship's speed are not correlated, the value 0 must lie between the 2 curves. In the example chosen, this is the case for the orthogonal components and not for the parallel components, which suggests an amplitude correction to be estimated via steps 3.3.4.4.4 and/or 3.3.4.8 and to be applied in the validation/correction tab (cf. 3.3.3.6).



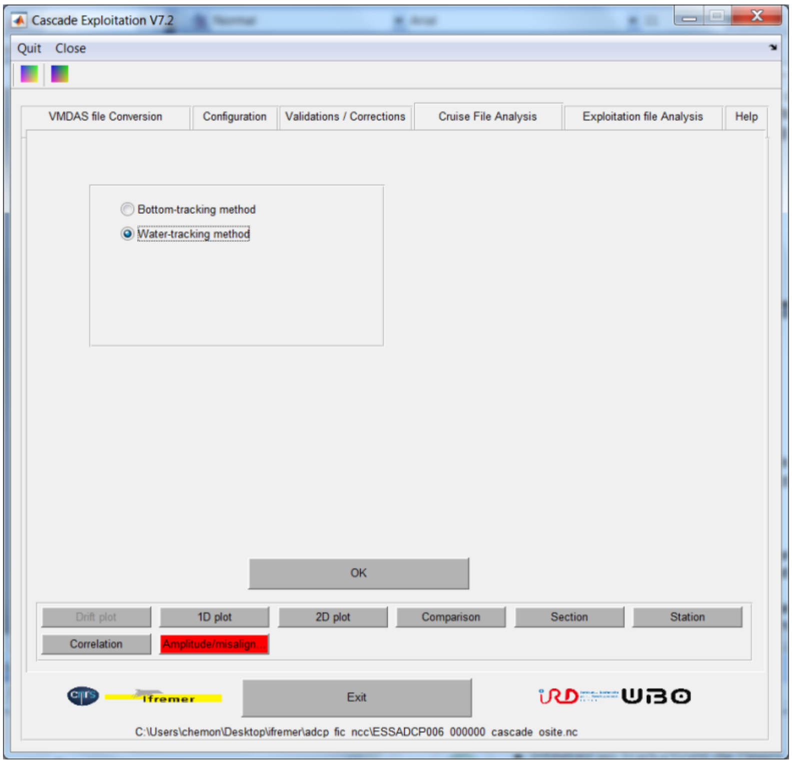
3.3.4.8 - Amplitude/alignment correction estimates

This step estimates a possible amplitude, misalignment and trim correction to be applied to the absolute current velocity data (in 3.3.3.6). Concerning amplitude and misalignment, this estimation may be necessary if a correlation between absolute current speeds and ship speed has been detected (see 3.3.4.7). In the case of trim, it makes it possible to reduce the vertical speed to an average close to 0 when the vessel is on the way.

3.3.4.8.1 Bottom-track method

An estimation of the amplitude and misalignment correction is estimated by comparing the ship's speed with the bottom-ping speed. (cf. 3.3.4.4.4). **As soon as sufficient bottom-ping data are available, it is strongly recommended to estimate corrections by using bottom-ping data, The watertrack method being much less accurate**.

3.3.4.8.2 Watertrack method



Interface

It is inspired by the calculation made in the Codas software, whose method is presented in Joyce (1989). For a α misalignment correction angle and an β amplitude correction, the basic equations are as follows:

$u_w = u_s - (1+\beta)(u_r \cos\alpha - v_r \sin\alpha)$ (1)

$v_w = v_s - (1+\beta)(u_r \sin\alpha + v_r \cos\alpha)$ (2)

where (u_w, v_w), (u_s, v_s), (u_r, v_r) are respectively the zonal and meridian components of $\mathbf{U_w}$, the current's velocity, $\mathbf{U_s}$, the ship's velocity, and $\mathbf{U_r}$, the opposite of the measured relative velocity.

We assume that during strong ship's accelerations $d\mathbf{U_s}/dt$, the current's acceleration can be neglected in the variation of the relative speed measured by the ADCP, which results in $|d\mathbf{U_w}| << |d\mathbf{U_s}|$, so $d\mathbf{U_r} \approx d\mathbf{U_s}$ if there is no misalignment or amplitude error (the time increment dt is the same for all variables and is simplified)

Using the properties of the vector product and the scalar product of $d\mathbf{U_s}$ **et** $d\mathbf{U_r}$, we can estimate the misalignment angle and amplitude errors as follows:

$d\mathbf{U_s} \wedge d\mathbf{U_r} = |d\mathbf{U_s}|.|d\mathbf{U_r}|.\sin\alpha \approx du_s dv_r - dv_s du_r$ (called duv in the program)

$d\mathbf{U_s} \cdot d\mathbf{U_r} = |d\mathbf{U_s}|.|d\mathbf{U_r}|.\cos\alpha \approx du_s du_r + dv_s dv_r$ (called uuvv in the program)

So:

- $\tan\alpha \approx (du_s dv_r - dv_s du_r) / (du_s du_r + dv_s dv_r)$
- From (1) and (2), if we neglect the current's acceleration $|d\mathbf{U_w}|$ (which is considered null) during the ship's acceleration, we deduce:
  $|d\mathbf{U_s}| = (1+\beta).|d\mathbf{U_r}|$. ==>
  $1+\beta \approx [du_s du_r + dv_s dv_r] / [(du_r^2 + dv_r^2).\cos\alpha]$

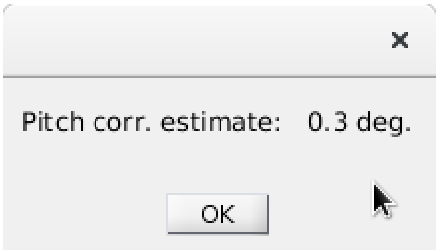
The data used for the calculation are selected as follows:

- We use only good data (flag 1).
- the depth layer with the least variability in absolute current velocities is calculated.
- We eliminate data with the above layer less than 100m from the bottom (to eliminate data that may be disturbed by bathymetry).
- The relative velocity on this layer is averaged for the remaining data.
- We look for data associated with an acceleration of the ship's speed greater than 0.008m/s^2 (0.007m/s^2 if the number of data is less than 10).
- If at least 10 accelerations are identified, the misalignment and amplitude correction is estimated using the above equations.
- Values of α et $1+\beta$ deviating by more than 2.7 standard deviations from their respective mean are eliminated and the mean is recalculated. The averages displayed are directly those to be implemented in the misalignment, trim and amplitude correction tab (3.3.3.6), only if the error calculated on the averages (based on a standard deviation, i.e. 68% confidence) allows to estimate that the corrections are significantly different from 0 for α and from 1 for $(1+\beta)$.

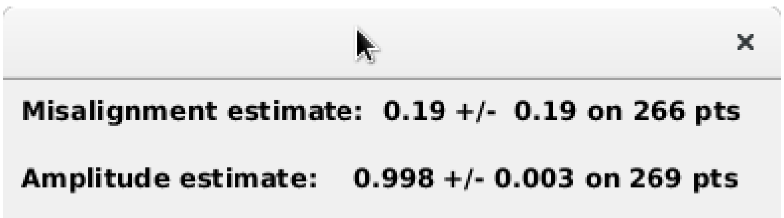
An estimate of the ADCP's trim error relative to the vessel is also calculated. To do this, it is assumed that <W>, the mean vertical current velocity, is very small, almost zero, in the layer of least variability in the horizontal current. Therefore, an average of more than 1 cm/s in absolute value may correspond to the projection of the ship's speed on the vertical due to a forward or aft inclination of the ADCP with respect to the ship's trim. To estimate this difference in trim, one moves into the layer of least variability and compares the vertical speed W to the ship's speed modul $<|\mathbf{U_s}|>$ when it exceeds 3 m/s (~6 nd) on bottoms greater than 1000m. An estimate of the ADCP trim P is thus deduced by $P \approx \sin(P) \approx <W/|\mathbf{U_s}|>$.

The information and plots generated by this step are:

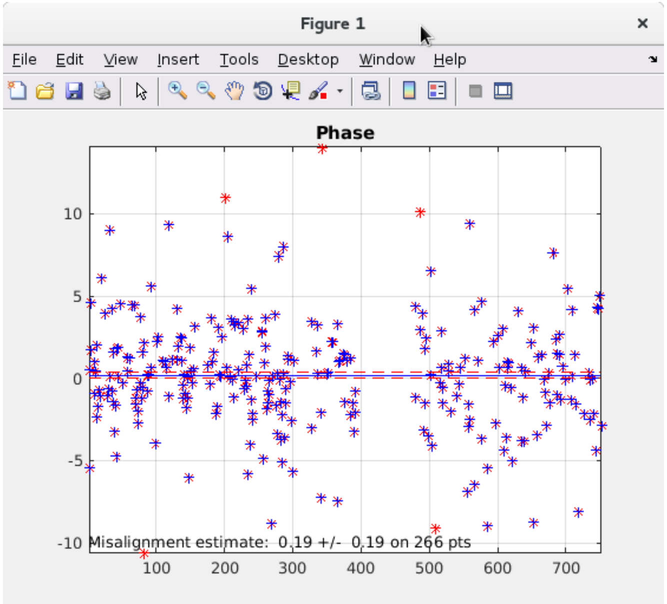
- The estimation of the ship's trim (rarely more accurate than 0.1°).
  Keep this value for later.

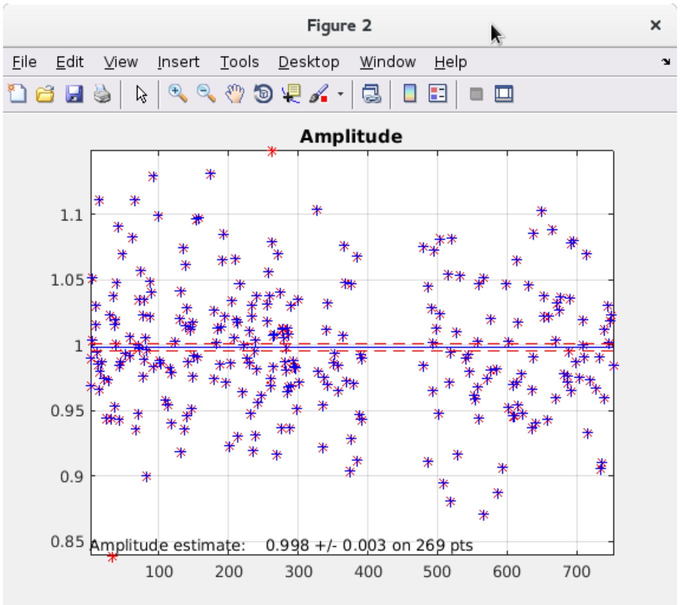


- Estimation of misalignment and amplitude, with a confidence interval of 68%. Note that if the precision provided for the misalignment is greater than the misalignment itself, the estimate is unreliable and it is recommended not to apply it.
  Similarly, if the accuracy given for the amplitude is greater than the amplitude minus 1 in absolute value, it is unreliable and should not be used.



- 2 plots are drawn, Figure 1 for the amplitude and 2 for the misalignment. For both plots:
  - In red star, all the accelerations taken into account for the estimation
  - In blue star, accelerations not eliminated by the deviation-to-average test
  - In blue solid line, the estimated misalignment/amplitude with, in red line, the associated error.

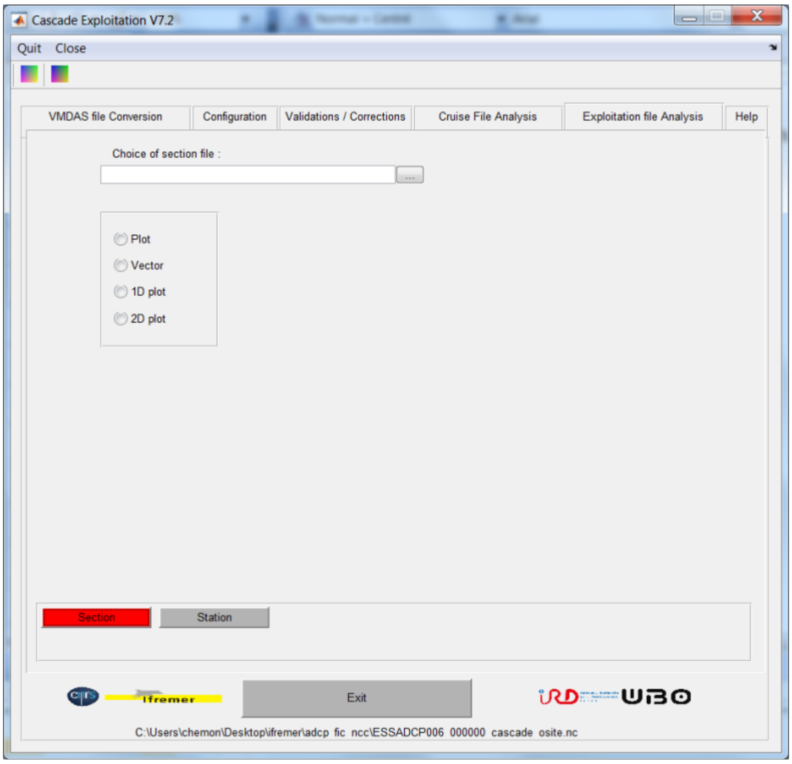




3.3.5 Analysis Operation Files

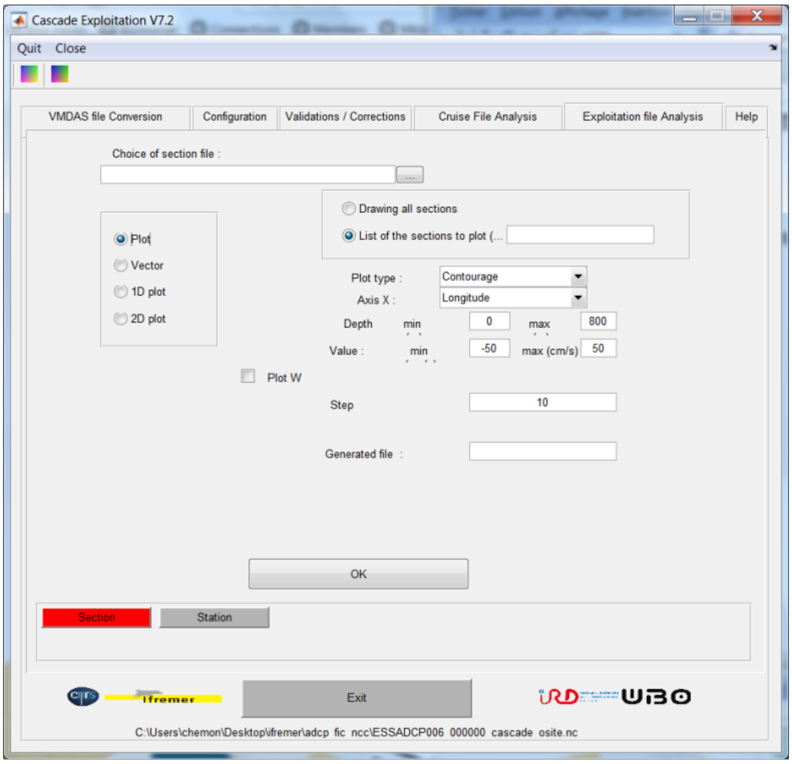
3.3.5.1 - Section

This step allows you to view the variables included in a Section file and to plot the associated current's velocities in various forms.
4 choices are possible: contouring, vector plots, 1D and 2D plots.



Interface

3.3.5.1.1 - Contouring

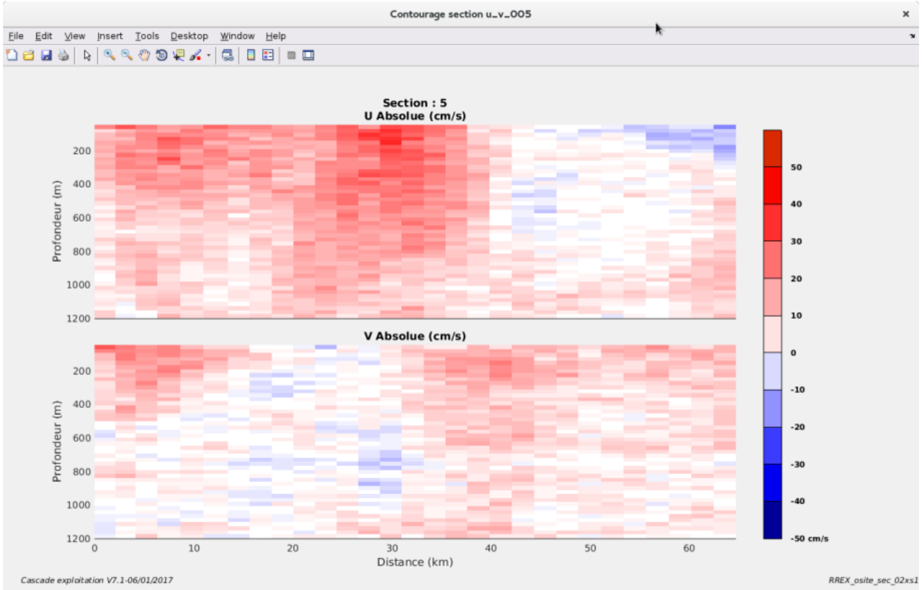


Interface

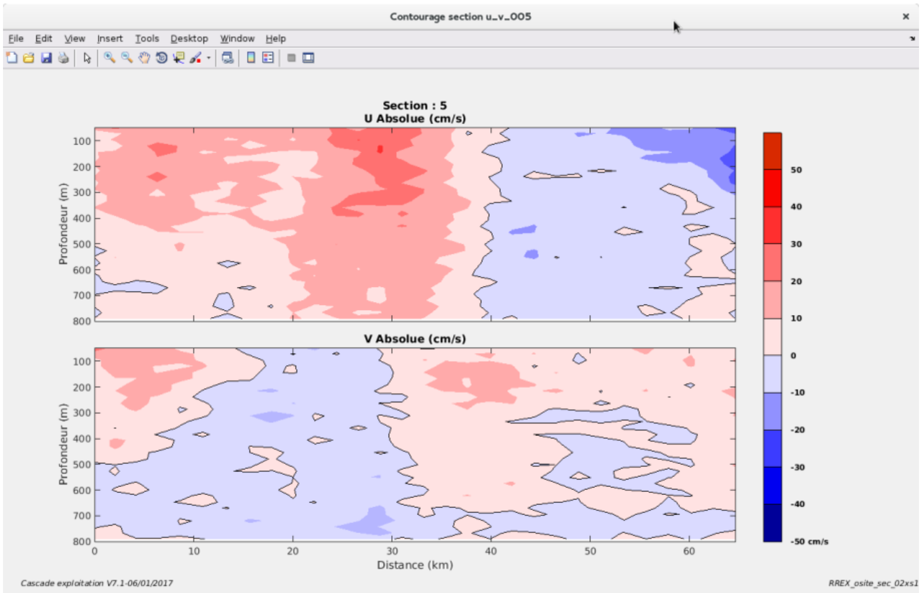
The users enters :

- the section file
- If he wishes to plot all sections or only particular sections. In this case, it provides the list of sections by separating the numbers of the sections to be taken into account with a semicolon
- If he wants a contour or image plot. The 'image' option is faster than the 'Contouring' option but it is also pixelized and the deepest cell does not appear.
- X axis:
  - Longitude
  - Latitude
  - Distance (in km)

  Note that MATLAB needs monotonous values for contouring. In this case, if the user chooses the longitude or latitude on the x-axis, the program automatically cuts out the user's sections as long as the longitude (or latitude) is not monotonous. To correct this, the user can either plot as an image, or choose the distance along the x-axis. This option assures the user that his sections are not cut out, as the cumulative distance from the first point of the section is necessarily increasing.
- Y-axis limits. Y-axis is the depth in m
- Minimum and maximum values of the data to be plotted
- Whether or not he wants to plot the vertical velocity data. In this case, he shall specify the minimum and maximum values associated.
- contouring step in cm/s
- filenames's root for saving the generated plots.



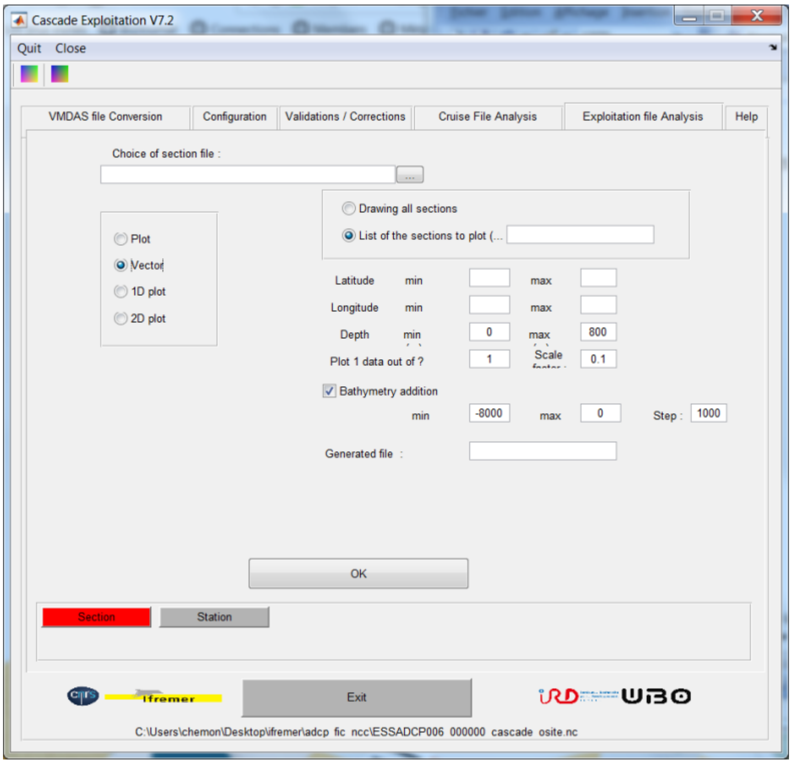
Contouring as 'image'



Contouring

An additional plot is automatically generated. It represents the contouring or image of the current velocity in modul and direction.

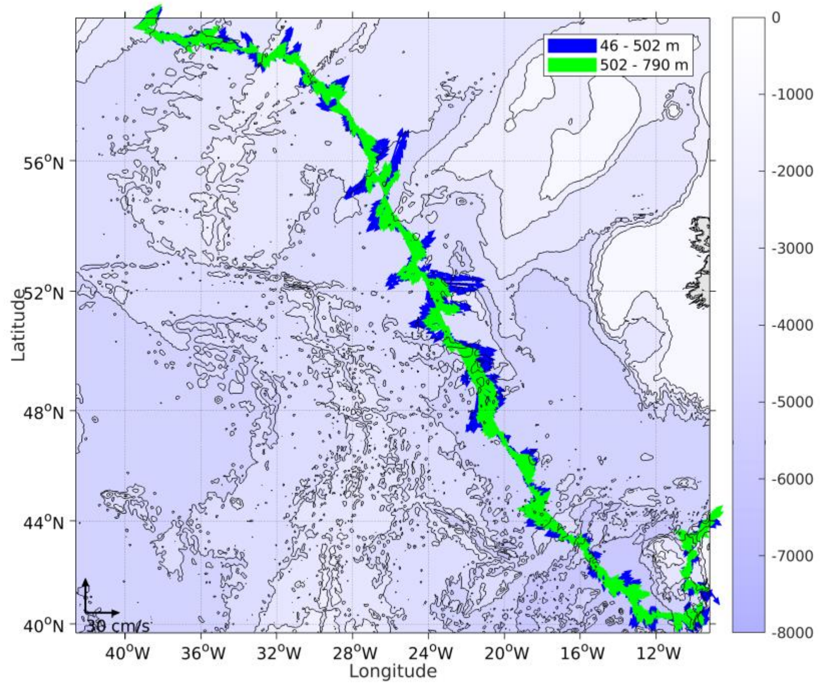
3.3.5.1.2 - Vector



Interface

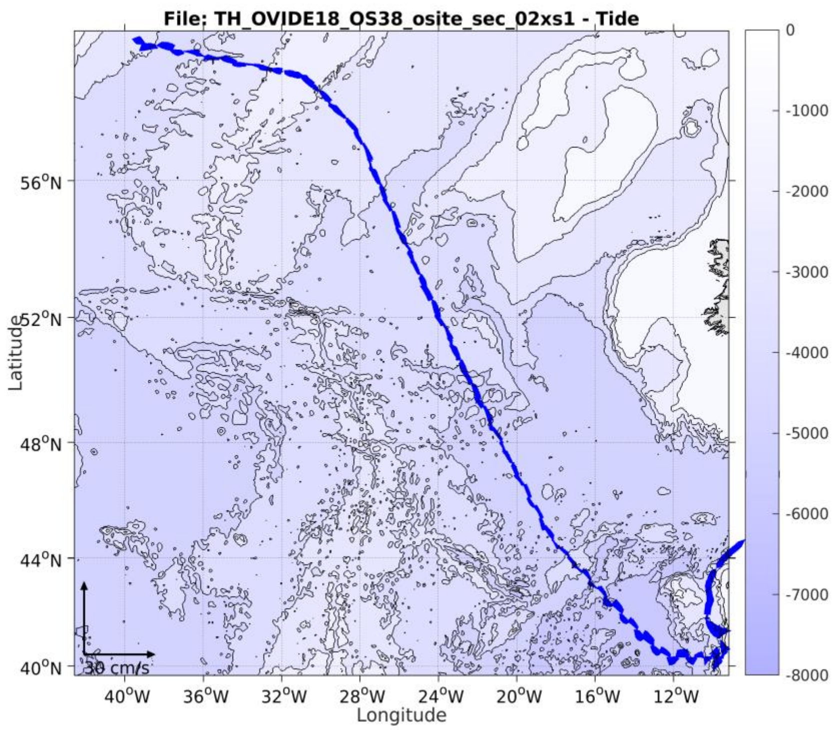
The users enters:

- The section file
- If he wishes to plot all sections or only particular sections. In this case, it provides the list of sections by separating the numbers of the sections to be taken into account with a semicolon
- the geographical limits
- Minimum and maximum depth for each of the layers to be considered. For the 2 layers [50 100] and [200 300], he enters [50 200] for minimum depth and [100 300] for maximum depth. The absolute current's velocities are averaged, for each point of the section, over each of the depth layers thus defined. The depths are to be entered with positive values.
- The percentage of data he wants to trace:
  - 1 data on 1: 1 vector for each section point is plotted
  - 1 data on X: 1 vector every X points of each section is plotted
- The scale factor that determines the size of the vectors. The larger the factor, the larger the size of the vectors
- If the bathymetry must be plotted or not. If bathymetry is desired, the user specifies the minimum and maximum depths to be plotted as well as the desired contour pitch.
  Warning: The bathymetry values to be entered are <=0 (positive bathymetries corresponding to values on the continent).
- filenames's root for saving the generated plots.



absolute current velocity vectors for the depth [47 95]m

After a first plot representing the absolute current velocity vectors, a window is displayed, asking the user if he wants an additional plot representing, at the same geographical points, the tidal velocity vectors.



Tide vector

3.3.5.1.3 - 1D plot

This step allows you to plot the one-dimensional variables of a Section file. The X-axis can be the time or ensembles' number.

The user enters:

- the section file
- The 1 dimension variables he wants to plot (4 variables maximum on a same plot)
- X axis:
  - ensemble number
  - day in the year
- filenames's root for saving the generated plots

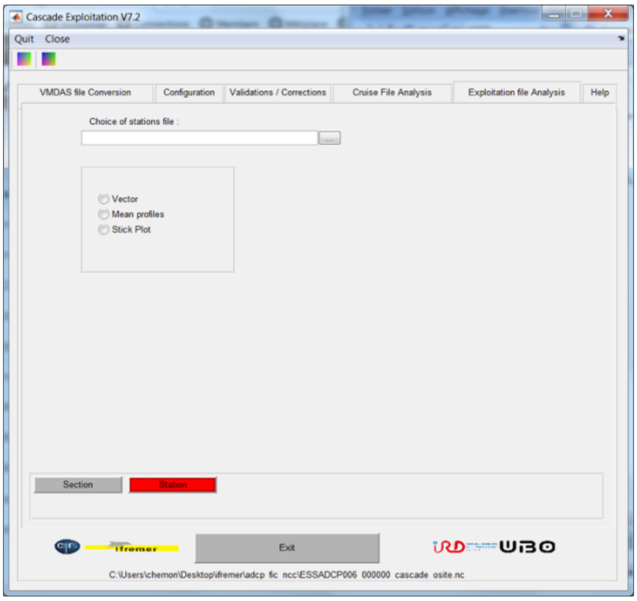
3.3.5.1.4 - 2D plot

This step allows the user to plot the 2D variables in a Section file. X-axis is ensembles number, Y-axis depth (or cell).

The user enters:

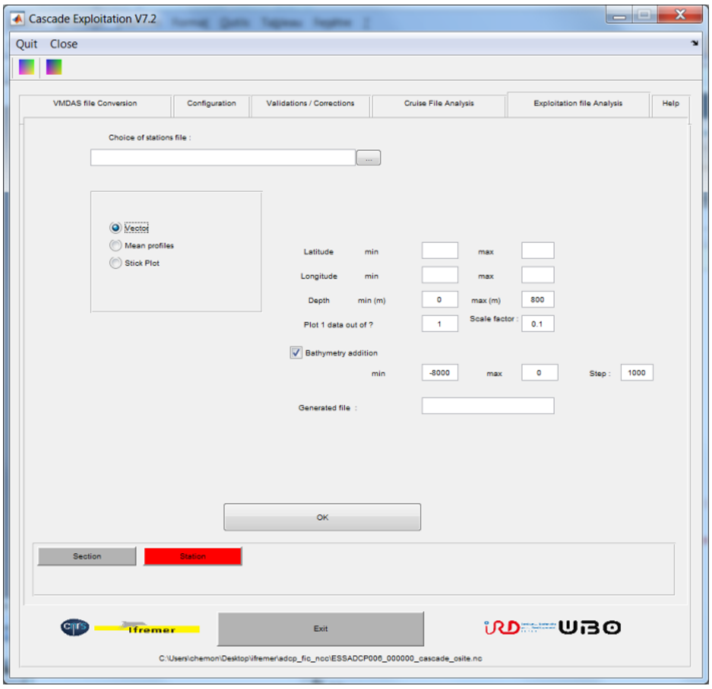
- the section file
- The variables to plot, with a maximum of 4 variables. Note that if the user visualizes one of the current speeds (meridian and/or zonal), an additional trace is automatically generated. It represents the current velocity in modul and direction.
- Y-axis:
  - cell
  - depth (m)
- filenames's root for saving the generated plots

3.3.5.2 - Station



Interface

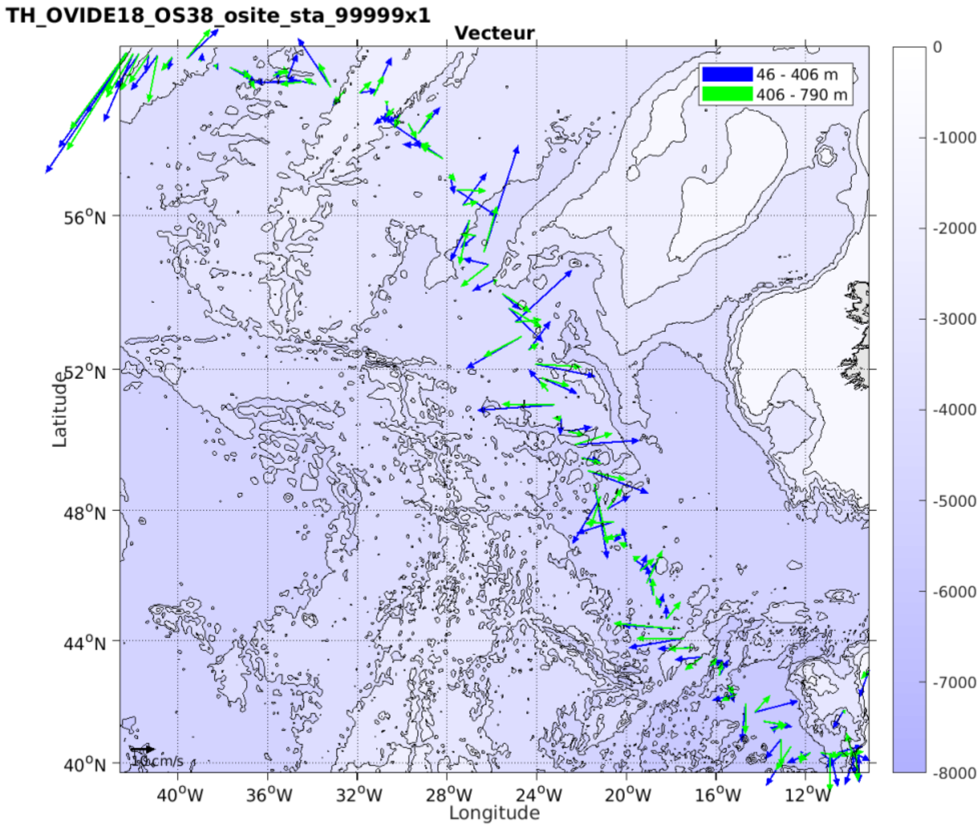
3.3.5.2.1 - Vector



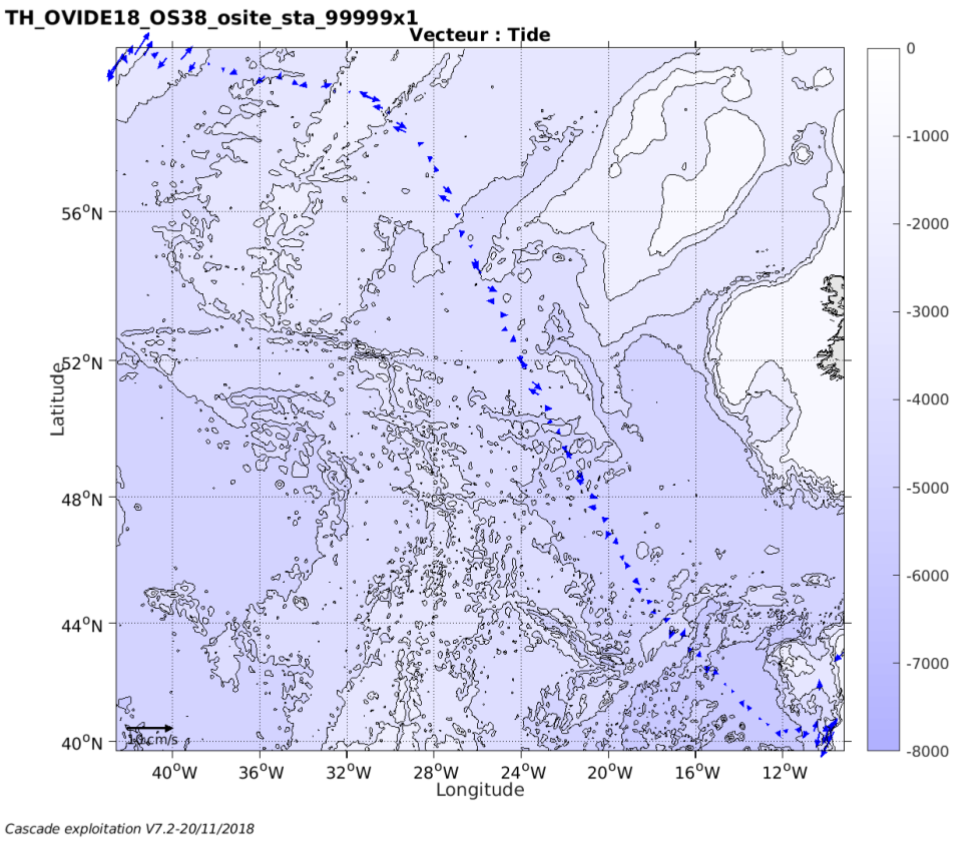
The user enters:

- The station file.
- the goegraphical limits
- Minimum and maximum depth for each of the layers to be considered. For the 2 layers [50 100] and [200 300], he enters [50 200] for minimum depth and [100 300] for maximum depth. The absolute current's velocities are averaged, for each point of the section, over each of the depth layers thus defined. The depths are to be entered with positive values.
- The percentage of data he wants to trace:
  - 1 data on 1: 1 vector for each section point is plotted
  - 1 data on X: 1 vector every X points of each section is plotted
- The scale factor that determines the size of the vectors. The larger the factor, the larger the size of the vectors
- If the bathymetry must be plotted or not. If bathymetry is desired, the user specifies the minimum and maximum depths to be plotted as well as the desired contour pitch.
  Warning: The bathymetry values to be entered are <=0 (positive bathymetries corresponding to values on the continent).
- filenames's root for saving the generated plots.

2 plots are generated ; one with absolute current velocity vectors, one the tidal velocity vectors.

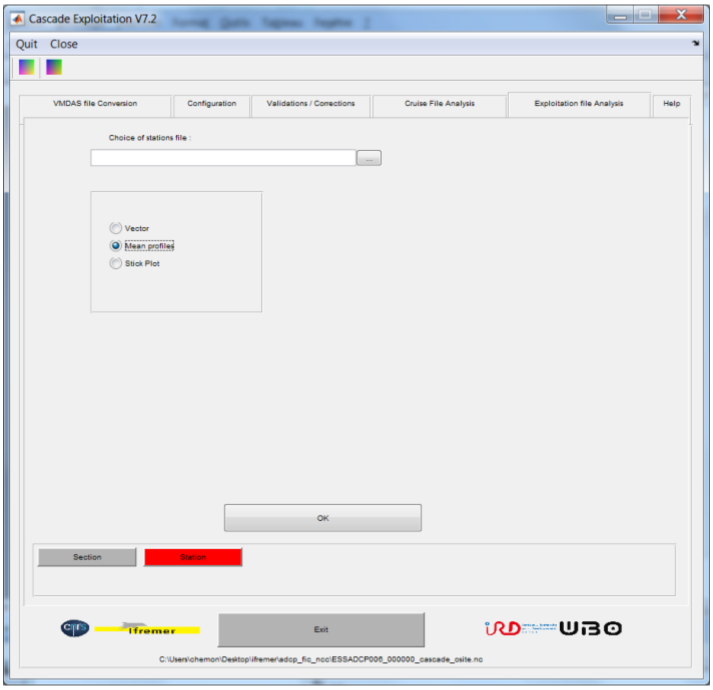


absolute current velocity vectors

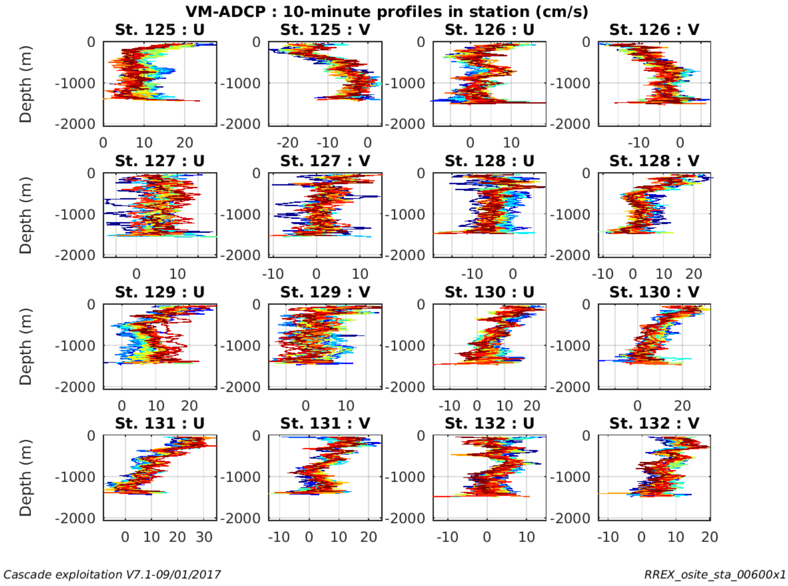


Tide vector

3.3.5.2.2 - Mean profile



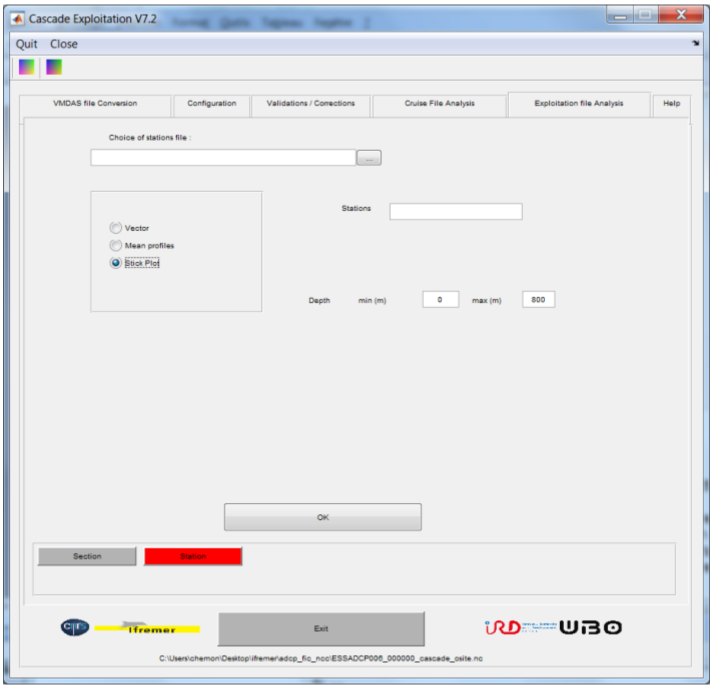
The user enters the station file. If he chooses a Station file where the data has been averaged over the entire duration of the station, the plot will present an average absolute current velocity profile per station. Otherwise, the plot will present an average absolute velocity profile for each TPS second (user-defined TPS at file creation, see 3.3.4.6) for each station.



Absolute current velocity profiles for stations 125 to 132
(1 average profile every 10 minutes)

The plots are saved in the 'plot' subdirectory:
<filename>_profils_sta_N.png
<filename> is the station file. *N* is the number of the plot.

3.3.5.2.3 Stick-plot

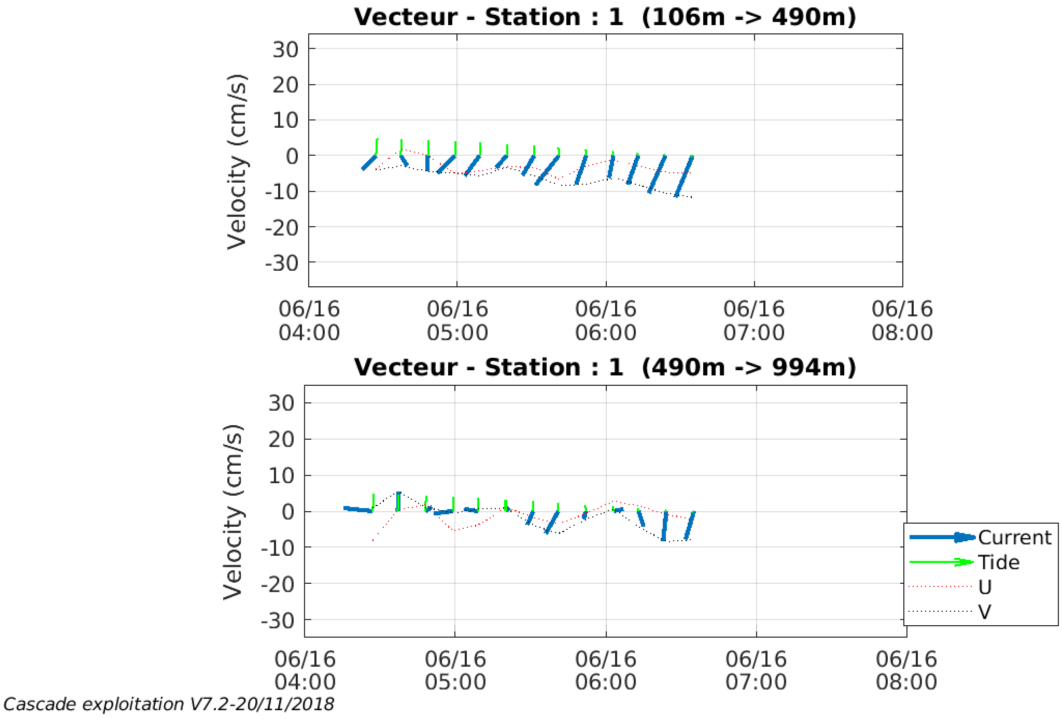


Interface

The user enters:

- the station file
- the list of stations to be taken into account for the plots (<u>ex:</u> 2:3 20 25:30 for stations [2 3 20 25 26 27 28 29 30].
- Minimum and maximum depth for each of the layers to be considered. For the 2 layers [50 100] and [200 300], he enters [50 200] for minimum depth and [100 300] for maximum depth. The absolute current's velocities are averaged, for each point of the section, over each of the depth layers thus defined. The depths are to be entered with positive values.

2 stickplot are generated; one with the absolute current velocities, one with tide velocities. These graphs help to study the temporal evolution of the absolute current velocities and the tide velocities at each station.

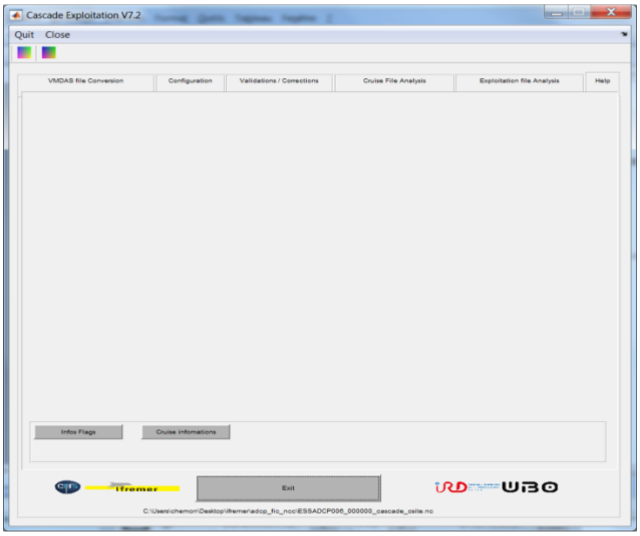


Temporal evolution (every 10 min) of the absolute current speeds for the station 1

These plots are saved in the subdirectory 'plot':
station_stick_*NNN_*xxxx.png where NNN is the station number, xxxx indicates the time (in seconds) over which the station is averaged.

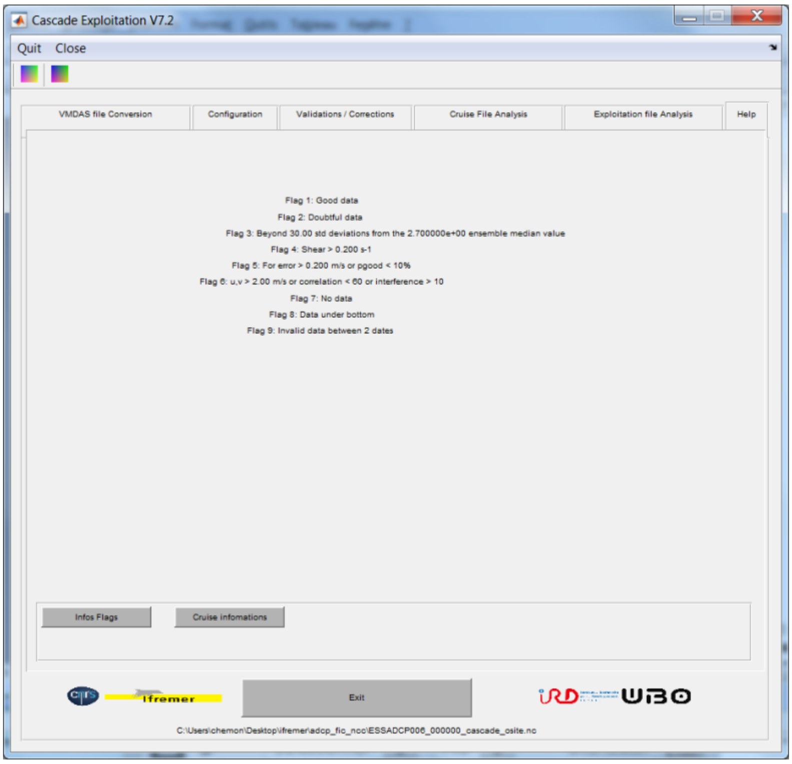
3.3.6 Help

This allows the user to have access to the flags and general information of the working file at any time.



3.3.6.1 - Flags information

This function informs the user, at any time, about the definition of the flags of the current cruise file.



Flags information

3.3.6.2 - Cruise information

Launching this command results in the creation of a file ncc/<filename>.txt summarizing all the information related to the cruise, the data and the analysis likely to appear in a report. This information, taken from the working file, is also displayed on the screen in a "Cruise Info" figure (below).

The mean vertical velocity for good data is the average, over the entire water column except for the first 2 cells, of all the correct data (flag 1).
The second indicated vertical velocity is the average, over the layer of least variability, of all the correct data (flag 1), used for the trim estimation (section 3.3.4.8). Except in special cases, if the attitude is good, both values are assumed to be less than 1 cm/s in absolute value.
Correlation information is detailed in section 3.3.4.7.
If "bottom_ping" data are present in the file, the mean difference between ship and bottom velocities is provided for each component with the associated precision (standard_deviation/sqrt(number_of_data). Means significantly different from zero indicate a residual bias.

The flags are explained and the percentage of associated data indicated. A high percentage of missing data usually indicates that the configuration had too many cells on the vertical in relation to the effective range of the ADCP, and the file can probably be optimized (cf. 3.3.3.12).

4 Références

Firing E. and Bahr F.,Caldwell P., Ranada J., Zhu W., 1995 . Processing ADCP Data with the CODAS software System, Version 3.1, JIMAR, University of Hawaii, 1000 Pope Road, Honolulu, Hawaii 96822, "unpublished manuscript".

Joyce, T.M. (1989): On in situ “calibration” of shipboard ADCPs. *J. Atmos. Oceanic Technol.*, 6 (2), 169–172.

Marchalot, C., J. P. Berthomé, J. Bertrand, A. Cressard, C. Edy, F. Gaillard, R. Le Suavé, P. Viollette: Groupe de Travail Données Navires. Rapport final de la phase 0. R.Int.TMSI/IDM/02-017

Kermabon, C. et F. Gaillard, Janvier 2001: CASCADE: logiciel de traitement des données ADCP de coque. Documentation maintenance - utilisateur (LPO-IFREMER).

Izenic Y, C. Kermabon, F . Gaillard, P, Lherminier: CASCADE 4.4: logiciel de traitement et d’anlyse des mesures ADCP de coque

Format OceanSite, oceansites-user-manual-v0.6.doc, T.Carval

5 Annexes

5.1 List of Useful Non- CASCADE Functions

5.1.1 conv_trait2exploit.m

This function converts a NetCDF cruise file generated by the PROCESSING part of CASCADE into a NetCDF cruise file in OceanSite format compatible with the OPERATION part of CASCADE. It is therefore only useful to go back to the processing of ADCP files originally acquired with the RDI Transect acquisition software.

<u>Calling the function:</u>
conv_trait2exploit(filename) where filename is a NetCDF cruise file genrated by PROCESSING part of CASCADE. In results, a filename filename_osite.nc is created. This new file can be used in the PERATION of CASCADE.

5.1.2 ncc2confexploit.m

This function generates a conf_exploit.mat file from a NetCDF cruise file in Oceansite format. This may be necessary if a user retrieves a cruise file without the associated conf_exploit.mat. If the user wishes to go back to the processing of the file, he will then be able to retrieve, by default, the parameters associated with the previous processing of this same file (cleaning parameters, cruise name, ship name,...).

<u>Calling the function:</u>
ncc2conf_exploit(filename) where filename is a cruise NetCDF file in Oceansite format. This file must be located in the subdirectory ncc, in the directory where the conf_exploit.mat file will be created.

5.1.3 ReadLogs

This function reads all the *.LOG files generated by the VmDas RDI acquisition software and creates, in the current directory, a LogCruise.cvs file including, for each *.LOG file, the associated VmDas acquisition configuration.

<u>Calling the function:</u>
ReadLogs(directoryname) with directoryname is the directory where the *.LOG files take place. The LogCruise.csv is created in the current directory.

5.1.4 f_VGcor_reinit.m

This function allows you to correct a NetCDF Oceansite cruise file for any misalignment, pitch or amplitude. Compared to the equivalent function implemented in CASCADE EXPLOITATION, this function allows to make a correction on a file already corrected for a misalignment/pitch/amplitude. Therefore, **this function should be used with care**. Because the product of rotation matrices is not switchable, making a single correction for misalignment of angle A and a pitch of angle B is not canceled by making a single correction for misalignment of angle -A and pitch -B.

Calling the function:
f_VGcor_reinit(filename,amplitude,misalignment, trim) with filename, the file to be corrected ; amplitude, misalignment and trim the corrections to apply.

5.2 Example output of the ReadLogs.m routine

```
% Start Time
% End Time
% WPnnn: Broadband mode on (1) or off (0)
% NPnnn: Narrowband mode on (1) or off (0)
% WNnnn: Number of bins in broadband mode
% NNnnn: Number of bins in narrowband mode
% WSnnn: Bin length(cm) of broadband pings
% NSnnn: Bin length(cm) of narrowband pings
% BPn: Bottom-track mode on (1) or off (0)
% BXnnnnn: Maximum bottom tracking depth (decimeters)
% CXa,b: Trigger in and out
% EA: transducer misalignment (hundredths of degrees)
% ES: ADCP bath Salinity (psu)
% EZ: use data from a manual setting/ext. sensor (cf doc)
% NFnnnn: Blank depth in narrowband mode (cm)
% WFnnnn: Blank depth in broadband mode (cm)
% TPmmssff: Time between pings (minute second hundredth)
% VmDas Version
% ADCP Type
OV18018_000000.LOG;14/06/2018 13:34:07;14/06/201815:06:02;WP0;NP001; ;
;NN061;NS2400;BP0;BX20000;CX0,1;EA04473;ES35;EZ1000001;NF1600;
;TP000200;1.49;OceanSurveyor
OV18019_000000.LOG;14/06/2018 15:06:28;14/06/2018 16:01:11;WP0;NP001; ;
;NN061;NS2400;BP0;BX20000;CX0,1;EA04473;ES35;EZ1000001;NF1600;
;TP000200;1.49;OceanSurveyor
OV18020_000000.LOG;14/06/2018 16:04:12;14/06/2018 16:53:03;WP0;NP001; ;
;NN061;NS2400;BP0;BX20000;CX0,1;EA4473 ;ES35;EZ1000001;NF1600;
;TP000200;1.49;OceanSurveyor
```

5.3 ASCII section/ASCII station example

A section file must have the extension _sec.list (example: rrex_all_sec.list) for sections.
For each section, it must contain the start date and time, and the end date and time.

Example

```
1 03/07/2015 16:30:00 06/07/2015 08:10:00
2 06/07/2015 08:20:00 08/07/2015 14:38:00
3 08/07/2015 14:38:00 13/07/2015 06:40:00
```

Start date **End date**

A station file must have the extension _sta.list (exemplerrex_all_sta.list) for stations.
For each station, it must contain the start date and time and the end date and time

Example

```
1 03/07/2015 16:30:00 03/07/2015 18:10:00
2 13/07/2015 00:38:00 13/07/2015 06:40:00
```

Start date **End date**

How to create these files ?

The user can use the latitude/longitude plot (1D plot) with the datation tool and then manually enter the desired values in a file according to the above formats.

5.4 NetCDF cruise file (Oceansite)

netcdf RREX_osite {
dimensions:
CONST1 = 1 ;
N_DATE_TIME = 24191 ; (**number of ensemble**)
N_LEVEL = 85 ; (**number of cell**)
DATE_TIME = 14 ; (Format DATE: AAAMMJJHHMMSS)
CONST2 = 2 ;
STRING32 = 32 ;
STRINGFILT = 32 ;
variables:
float **TX_FREQUENCY**(CONST1) ; (**ADCP frequency**)
TX_FREQUENCY:long_name = "ADCP transmitter frequency" ;
TX_FREQUENCY:units = "KiloHz" ;
TX_FREQUENCY:_FillValue = -999999.f ;
float **SCALE_FACTOR**(CONST1) ; (**scale factor: used in Cascace processing, to transform from beam coordinates to geographical coordinates**)
SCALE_FACTOR:long_name = "Scale factor" ;
SCALE_FACTOR:_FillValue = -999999.f ;
float **BEAM_ANGLE**(CONST1) ; (**Beam angle**)
BEAM_ANGLE:long_name = "Beam Angle/vertical" ;
BEAM_ANGLE:units = "degrees" ;
BEAM_ANGLE:_FillValue = -999999.f ;
float **ADCP_ANGLE**(CONST1) ; (**ADCP angle/ship**)
ADCP_ANGLE:long_name = "ADCP Angle/ship axis" ;
ADCP_ANGLE:units = "degrees" ;
ADCP_ANGLE:_FillValue = -999999.f ;
float **BIN_LENGTH**(CONST1) ; (**Bin length**)
BIN_LENGTH:long_name = "Bin Length" ;
BIN_LENGTH:units = "meter" ;
BIN_LENGTH:_FillValue = -999999.f ;
float **MIDDLE_BIN1_DEPTH**(CONST1) ; (**Depth of first bin center**)
MIDDLE_BIN1_DEPTH:long_name = "Depth of first bin center" ;
MIDDLE_BIN1_DEPTH:units = "meters" ;
MIDDLE_BIN1_DEPTH:_FillValue = -999999.f ;
float **HEAD_MISLG**(CONST1) ; (**ADCP misalignment)**
HEAD_MISLG:long_name = "Heading Misalignment" ;
HEAD_MISLG:units = "degrees" ;
HEAD_MISLG:_FillValue = -999999.f ;
float **PITCH_MISLG**(CONST1) ; (**trim**)
PITCH_MISLG:long_name = "Pitch Misalignment" ;
PITCH_MISLG:units = "degrees" ;
PITCH_MISLG:_FillValue = -999999.f ;
float **AMPLI_CORFAC**(CONST1) ; (**amplitude prise**)
AMPLI_CORFAC:long_name = "Correction factor on velocity amplitude" ;
AMPLI_CORFAC:_FillValue = -999999.f ;
float **XOFF**(CONST1) ; (**Transducer depth**)
XOFF:long_name = "Transducer Depth" ;
XOFF:_FillValue = -999999.f ;
float **CORR_PR**(CONST1) ; (**indicates if ADCP velocities are corrected or not from pitch/roll)**
CORR_PR:long_name = "=1: pitch/roll used, 0 otherwise" ;
CORR_PR:_FillValue = -999999.f ;
char **REFERENCE_DATE_TIME**(DATE_TIME) ; (**Date of reference julian day**)
REFERENCE_DATE_TIME:long_name = "Date of reference julian day" ;
REFERENCE_DATE_TIME:conventions = "YYYYMMDDHHMMSS" ;
REFERENCE_DATE_TIME:_FillValue = " " ;
double **JULD**(N_DATE_TIME) ; (**julian day associated to each ensemble**)
JULD:long_name = "Julian day relative to REFERENCE_DATE_TIME" ;
JULD:_FillValue = -999999. ;

double **JULD_ADCP**(N_DATE_TIME) ; (**julian day associated to the ADCP PC**)
JULD_ADCP:long_name = "ADCP Julian day relative to REFERENCE_DATE_TIME" ;
JULD_ADCP:_FillValue = -999999. ;
double **JULD_j1**(N_DATE_TIME) ; (**julian GPS day of the beginning of each ensemble**)
JULD_j1:long_name = "Begin Ensemble Julian day relative to REFERENCE_DATE_TIME" ;
JULD_j1:_FillValue = -999999. ;
double **JULD_j2**(N_DATE_TIME) ; (**julian GPS day of the end of each ensemble**)
JULD_j2:long_name = "End Ensemble Julian day relative to REFERENCE_DATE_TIME" ;
JULD_j2:_FillValue = -999999. ;
char **DATE_TIME_UTC**(N_DATE_TIME, DATE_TIME) ; (**gregorian date and time for each ensemble**)
DATE_TIME_UTC:long_name = "ASCII gregorian date and time" ;
DATE_TIME_UTC:convention = "YYYYMMDDHHMMSS" ;
DATE_TIME_UTC:_FillValue = " " ;
float **CAS_DATE_FLAG**(N_DATE_TIME) ; (**flag associated with dates. A -999999 by default, to 2 when the associated data has been invalidated by the user (invalidation by set and/or by date**).
CAS_DATE_FLAG:long_name = "Flag on date" ;
CAS_DATE_FLAG:_FillValue = -999999.f ;
float **LATITUDE**(N_DATE_TIME) ; (**latitude for each ensemble**)
LATITUDE:long_name = "Latitude of each location" ;
LATITUDE:units = "degree_north" ;
LATITUDE:valid_min = -90. ;
LATITUDE:valid_max = 90. ;
LATITUDE:_FillValue = -999999.f ;
float **LONGITUDE**(N_DATE_TIME) ; (**longitude for each ensemble**)
LONGITUDE:long_name = "Longitude of each location" ;
LONGITUDE:units = "degree_north" ;
LONGITUDE:valid_min = -180. ;
LONGITUDE:valid_max = 180. ;
LONGITUDE:_FillValue = -999999.f ;
float **UVEL_SHIP**(N_DATE_TIME) ; **(zonal ship speed associated with each ensemble**)
UVEL_SHIP:long_name = "Eastward ship velocity" ;
UVEL_SHIP:units = "meter per second" ;
UVEL_SHIP:valid_min = -20. ;
UVEL_SHIP:valid_max = 20. ;
UVEL_SHIP:_FillValue = -999999.f ;
float **VVEL_SHIP**(N_DATE_TIME) ; **(meridian ship speed associated with each ensemble**)
VVEL_SHIP:long_name = "Northward ship velocity" ;
VVEL_SHIP:units = "meter per second" ;
VVEL_SHIP:valid_min = -20. ;
VVEL_SHIP:valid_max = 20. ;
VVEL_SHIP:_FillValue = -999999.f ;
float **MODE**(N_DATE_TIME) ; (**ADCP mode: BroadBand or NarrowBand)**
MODE:long_name = "Acquisition Mode: 1 = BroadBand - 10 = NarrowBand" ;
MODE:_FillValue = -999999.f ;
float DEPH(N_LEVEL) ;
DEPH:long_name = "Depth of bin center" ;
DEPH:units = "meter" ;
DEPH:valid_min = -12000. ;
DEPH:valid_max = 0. ;
DEPH:_FillValue = -999999.f ;
float **TEMP_ADCP**(N_DATE_TIME) ; (**ADCP transducer temperature**)
TEMP_ADCP:long_name = "ADCP transducer temperature" ;
TEMP_ADCP:units = "Degree_Celsius" ;
TEMP_ADCP:valid_min = -5. ;
TEMP_ADCP:valid_max = 45. ;
TEMP_ADCP:_FillValue = -999999.f ;
float **HDG**(N_DATE_TIME) ; (**ship heading for each ensemble**)
HDG:long_name = "Ship Heading" ;
HDG:units = "Degree" ;
HDG:valid_min = -360. ;

HDG:valid_max = 360. ;
HDG:_FillValue = -999999.f ;
float **HDG_G1**(N_DATE_TIME) ; (**other heading if exists**)
HDG_G1:long_name = "Ship Heading G1" ;
HDG_G1:units = "Degree" ;
HDG_G1:valid_min = -360. ;
HDG_G1:valid_max = 360. ;
HDG_G1:_FillValue = -999999.f ;
float HDG_G2(N_DATE_TIME) ; (**other heading if exists**)
HDG_G2:long_name = "Ship Heading G2" ;
HDG_G2:units = "Degree" ;
HDG_G2:valid_min = -360. ;
HDG_G2:valid_max = 360. ;
HDG_G2:_FillValue = -999999.f ;
float **PTCH**(N_DATE_TIME) ; (**ship pitch for each ensemble**)
PTCH:long_name = "Ship Pitch" ;
PTCH:units = "Degree" ;
PTCH:valid_min = -360. ;
PTCH:valid_max = 360. ;
PTCH:_FillValue = -999999.f ;
float ROLL(N_DATE_TIME) ; (**ship roll for each ensemble**)
ROLL:long_name = "Ship Roll" ;
ROLL:units = "Degree" ;
ROLL:valid_min = -360. ;
ROLL:valid_max = 360. ;
ROLL:_FillValue = -999999.f ;
float **NB_ENS_AVE**(N_DATE_TIME) ; (**Number of averaged pings per ensemble**)
NB_ENS_AVE:long_name = "Number of averaged pings per ensemble" ;
NB_ENS_AVE:_FillValue = -999999.f ;
float **U_BOTTOM**(N_DATE_TIME) ; (**Bottom Track Eastward velocity**)
U_BOTTOM:long_name = "Bottom Track Eastward velocity" ;
U_BOTTOM:units = "meter per second" ;
U_BOTTOM:valid_min = -20. ;
U_BOTTOM:valid_max = 20. ;
U_BOTTOM:_FillValue = -999999.f ;
float **V_BOTTOM**(N_DATE_TIME) ; (**Bottom Track Northward velocity**)
V_BOTTOM:long_name = "Bottom Track Northward velocity" ;
V_BOTTOM:units = "meter per second" ;
V_BOTTOM:valid_min = -20. ;
V_BOTTOM:valid_max = 20. ;
V_BOTTOM:_FillValue = -999999.f ;
float W_BOTTOM(N_DATE_TIME) ; (**Bottom Track Vertical velocity**)
W_BOTTOM:long_name = "Bottom Track Vertical velocity" ;
W_BOTTOM:units = "meter per second" ;
W_BOTTOM:valid_min = -20. ;
W_BOTTOM:valid_max = 20. ;
W_BOTTOM:_FillValue = -999999.f ;
float **RNG_BOTTOM**(N_DATE_TIME) ; (**Bottom Range**)
RNG_BOTTOM:long_name = "Bottom Range" ;
RNG_BOTTOM:units = "meter" ;
RNG_BOTTOM:_FillValue = -999999.f ;
float **UVEL_ADCP**(N_DATE_TIME, N_LEVEL) ; (**Eastward absolute ADCP current velocity**)
UVEL_ADCP:long_name = "Eastward absolute ADCP current velocity" ;
UVEL_ADCP:units = "meter per second" ;
UVEL_ADCP:valid_min = -20. ;
UVEL_ADCP:valid_max = 20. ;
UVEL_ADCP:_FillValue = -999999.f ;
float **VVEL_ADCP**(N_DATE_TIME, N_LEVEL) ; (**Northward absolute ADCP current velocity**)
VVEL_ADCP:long_name = "Northward absolute ADCP current velocity" ;
VVEL_ADCP:units = "meter per second" ;

VVEL_ADCP:valid_min = -20. ;
VVEL_ADCP:valid_max = 20. ;
VVEL_ADCP:_FillValue = -999999.f ;
float **WVEL_ADCP**(N_DATE_TIME, N_LEVEL) ; (**Vertical absolute ADCP current velocity**)
WVEL_ADCP:long_name = "Vertical absolute ADCP current velocity" ;
WVEL_ADCP:units = "meter per second" ;
WVEL_ADCP:valid_min = -20. ;
WVEL_ADCP:valid_max = 20. ;
WVEL_ADCP:_FillValue = -999999.f ;
float **EVEL_ADCP**(N_DATE_TIME, N_LEVEL) ; (**absolute ADCP current velocity error**)
EVEL_ADCP:long_name = "absolute ADCP current velocity error" ;
EVEL_ADCP:units = "meter per second" ;
EVEL_ADCP:valid_min = -20. ;
EVEL_ADCP:valid_max = 20. ;
EVEL_ADCP:_FillValue = -999999.f ;
float **UVEL_REL**(N_DATE_TIME, N_LEVEL) ; (Eastward **relative ADCP current velocity**)
UVEL_REL:long_name = "Eastward relative ADCP current velocity" ;
UVEL_REL:units = "meter per second" ;
UVEL_REL:valid_min = -20. ;
UVEL_REL:valid_max = 20. ;
UVEL_REL:_FillValue = -999999.f ;
float **VVEL_REL**(N_DATE_TIME, N_LEVEL) ; (**Northward relative ADCP current velocity**)
VVEL_REL:long_name = "Northward relative ADCP current velocity" ;
VVEL_REL:units = "meter per second" ;
VVEL_REL:valid_min = -20. ;
VVEL_REL:valid_max = 20. ;
VVEL_REL:_FillValue = -999999.f ;
float **PGOOD_ADCP**(N_DATE_TIME, N_LEVEL) ; **(% of good data with 4 beams**)
PGOOD_ADCP:long_name = "% of good data with 4 beams" ;
PGOOD_ADCP:units = "percent" ;
PGOOD_ADCP:valid_min = 0. ;
PGOOD_ADCP:valid_max = 100. ;
PGOOD_ADCP:_FillValue = -999999.f ;
float **PGOOD_ADCP_B2**(N_DATE_TIME, N_LEVEL) ; **(% of good data with 3 beams)**
PGOOD_ADCP_B2:long_name = "% of good data with 3 beams" ;
PGOOD_ADCP_B2:units = "percent" ;
PGOOD_ADCP_B2:valid_min = 0. ;
PGOOD_ADCP_B2:valid_max = 100. ;
PGOOD_ADCP_B2:_FillValue = -999999.f ;
float **PGOOD_ADCP_B3**(N_DATE_TIME, N_LEVEL) ; **(% of good data rejected because of vertical velocity error. depends on acquisition configuration)**
PGOOD_ADCP_B3:long_name = "% of good data rejected because of EW" ;
PGOOD_ADCP_B3:units = "percent" ;
PGOOD_ADCP_B3:valid_min = 0. ;
PGOOD_ADCP_B3:valid_max = 100. ;
PGOOD_ADCP_B3:_FillValue = -999999.f ;
float **PGOOD_ADCP_B4**(N_DATE_TIME, N_LEVEL) ; **(% of data with at least 2 incorrect beam**)
PGOOD_ADCP_B4:long_name = "% of data with 2 bad beams" ;
PGOOD_ADCP_B4:units = "percent" ;
PGOOD_ADCP_B4:valid_min = 0. ;
PGOOD_ADCP_B4:valid_max = 100. ;
PGOOD_ADCP_B4:_FillValue = -999999.f ;
float **ECI**(N_DATE_TIME, N_LEVEL) ; (**Mean echo intensity (on the 4 beams)**)
ECI:long_name = "Mean echo Intensity" ;
ECI:units = "count" ;
ECI:_FillValue = -999999.f ;
float **ECI_B1**(N_DATE_TIME, N_LEVEL) ; (**Mean echo Intensity for beam 1, averaged on the all pings**)
ECI_B1:long_name = "Mean echo Intensity for beam 1" ;
ECI_B1:units = "count" ;
ECI_B1:_FillValue = -999999.f ;

float **ECI_B2**(N_DATE_TIME, N_LEVEL) ; (**Mean echo intensity for beam 2, averaged on the all pings**)
ECI_B2:long_name = "Mean echo Intensity for beam 2" ;
ECI_B2:units = "count" ;
ECI_B2:_FillValue = -999999.f ;
float **ECI_B3**(N_DATE_TIME, N_LEVEL) ; (**Mean echo intensity for beam 3, averaged on the all pings**)
ECI_B3:long_name = "Mean echo Intensity for beam 3" ;
ECI_B3:units = "count" ;
ECI_B3:_FillValue = -999999.f ;
float **ECI_B4**(N_DATE_TIME, N_LEVEL) ; (**Mean echo intensity for beam 4, averaged on the all pings**)
ECI_B4:long_name = "Mean echo Intensity for beam 4" ;
ECI_B4:units = "count" ;
ECI_B4:_FillValue = -999999.f ;
float **CORR**(N_DATE_TIME, N_LEVEL) ; (**Mean correlation (on the 4 beams)**)
CORR:long_name = "Correlation" ;
CORR:units = "count" ;
CORR:_FillValue = -999999.f ;
float **CORR_B1**(N_DATE_TIME, N_LEVEL) ; (**Mean correlation for beam 1, averaged on the all pings**)
CORR_B1:long_name = "Correlation for beam 1" ;
CORR_B1:units = "count" ;
CORR_B1:_FillValue = -999999.f ;
float **CORR_B2**(N_DATE_TIME, N_LEVEL) ; (**Mean correlation for beam 2, averaged on the all pings**)
CORR_B2:long_name = "Correlation for beam 2" ;
CORR_B2:units = "count" ;
CORR_B2:_FillValue = -999999.f ;
float **CORR_B3**(N_DATE_TIME, N_LEVEL) ; (**Mean correlation for beam 3, averaged on the all pings**)
CORR_B3:long_name = "Correlation for beam 3" ;
CORR_B3:units = "count" ;
CORR_B3:_FillValue = -999999.f ;
float **CORR_B4**(N_DATE_TIME, N_LEVEL) ; (**Mean correlation for beam 4, averaged on the all pings**)
CORR_B4:long_name = "Correlation for beam 4" ;
CORR_B4:units = "count" ;
CORR_B4:_FillValue = -999999.f ;
float CAS_CURRENT_FLAG(N_DATE_TIME, N_LEVEL) ; (**Flag on ADCP velocity (value from 1 to 9)**).
CAS_CURRENT_FLAG:long_name = "Flag on ADCP velocity" ;
CAS_CURRENT_FLAG:valid_min = 0. ;
CAS_CURRENT_FLAG:valid_max = 9. ;
CAS_CURRENT_FLAG:_FillValue = -999999.f ;
char **FILT_TYPE**(STRINGFILT) ; (**type of filtering)**)
FILT_TYPE:long_name = "Type of filtring" ;
float **VError**(N_DATE_TIME, N_LEVEL) ; (**précision des vitesses absolues de courant horizontales (std/sqrt(nb_données)))**
VError:long_name = "Horizontal Velocity precision" ;
VError:units = "m/s" ;
VError:_FillValue = -999999.f ;
float **VRMS_ADCP**(N_DATE_TIME) **; (Standard deviation of absolute current velocities over a ping (constructor's data))**
VRMS_ADCP:long_name = "Ping Horizontal Velocity precision" ;
VRMS_ADCP:units = "m/s" ;
VRMS_ADCP:_FillValue = -999999.f ;
int **REF_LAYER_ILIM**(CONST2) ; (**Reference layer**)
REF_LAYER_ILIM:long_name = "Reference Layer Limit Index" ;
REF_LAYER_ILIM:_FillValue = -999999 ;
int **FLAG3_HALF_WINDOW**(CONST1) ; (**Flag 3 half width of window**)
FLAG3_HALF_WINDOW:long_name = "Flag 3 half width of window" ;
FLAG3_HALF_WINDOW:_FillValue = -999999 ;
float **FLAG3_SCF_MED_DEV**(CONST1) ; **(Flag 3 scaling factor median deviation)**
FLAG3_SCF_MED_DEV:long_name = "Flag 3 scaling factor median deviation" ;
FLAG3_SCF_MED_DEV:_FillValue = -999999.f ;
float **FLAG5_MAX_WVEL**(CONST1) ; (**Flag 5 maximum horizontal velocity)**

FLAG5_MAX_WVEL:long_name = "Flag 5 maximum horizontal velocity" ;
FLAG5_MAX_WVEL:units = "meter per second" ;
FLAG5_MAX_WVEL:_FillValue = -999999.f ;
float **FLAG5_PGOOD_MIN**(CONST1) ; **(Flag 5 minimum percent data with 4 beam)**
FLAG5_PGOOD_MIN:long_name = "Flag 5 minimum percent data with 4 beam " ;
FLAG5_PGOOD_MIN:units = "percent" ;
FLAG5_PGOOD_MIN:_FillValue = -999999.f ;
float **FLAG4_MAX_VSHEAR**(CONST1) ; **(Flag 4 maximum vertical shear)**
FLAG4_MAX_VSHEAR:long_name = "Flag 4 maximum vertical shear" ;
FLAG4_MAX_VSHEAR:units = "second-1" ;
FLAG4_MAX_VSHEAR:_FillValue = -999999.f ;
int **FLAG8_BOTTOM**(CONST1) ; **(Flag 8 bottom detection)**
FLAG8_BOTTOM:long_name = "Flag 8 bottom detection" ;
FLAG8_BOTTOM:comments = "3 = no detection; 2 = ADCP bottom range; 1 = external bathymetry" ;
FLAG8_BOTTOM:_FillValue = -999999 ;
float **FLAG2_SCF_VSHEAR**(CONST1) ; (Flag 2 scaling factor on vertical shear)
FLAG2_SCF_VSHEAR:long_name = "Flag 2 scaling factor on vertical shear" ;
FLAG2_SCF_VSHEAR:_FillValue = -999999.f ;
float **FLAG2_MAX_DEV**(CONST1) ; **(Flag 2 maximum deviation relative to mean profile (cm/s))**
FLAG2_MAX_DEV:long_name = "Flag 2 maximum deviation relative to mean profile (cm/s)" ;
FLAG2_MAX_DEV:_FillValue = -999999.f ;
float **FLAG6_MAX_VVEL**(CONST1) ; **(flag 6 maximum vertical velocity)**
FLAG6_MAX_VVEL:long_name = "flag 6 maximum vertical velocity" ;
FLAG6_MAX_VVEL:units = "meter per second" ;
FLAG6_MAX_VVEL:_FillValue = -999999.f ;
float **FLAG6_INTERF**(CONST1) ; **(flag 6 Interference threshold)**
FLAG6_INTERF:long_name = "Interference threshold" ;
FLAG6_INTERF:_FillValue = -999999.f ;
float **FLAG6_MIN_CORR**(CONST1) ; **(flag 6 Threshold correlation)**
FLAG6_MIN_CORR:long_name = "Threshold correlation" ;
FLAG6_MIN_CORR:_FillValue = -999999.f ;
float **BATHY**(N_DATE_TIME) ; **(Bathymetry**)
BATHY:units = "meter" ;
BATHY:comment = "4.4.0" ;
BATHY:long_name = "ETOPO1_Bed_g_gmt4.grd" ;
BATHY:_FillValue = -999999.f ;
BATHY:valid_min = -12000. ;
BATHY:valid_max = 0. ;
BATHY:file = "/home/lpo5/LOGICIELS/VMADCP/Version/adcp_lpo_V7.1/exploitation/bathymetrie/bathy_etopo1.nc" ;
float **U_TIDE**(N_DATE_TIME) ; **(Eastward tide Velocity)**
U_TIDE:units = "meter per second" ;
U_TIDE:type_tide = "Model_atlas_v1" ;
U_TIDE:long_name = "Eastward tide Velocity" ;
U_TIDE:_FillValue = -999999.f ;
U_TIDE:valid_min = -20. ;
U_TIDE:valid_max = 20. ;
float **V_TIDE**(N_DATE_TIME) ; **(Northward tide Velocity)**
V_TIDE:units = "meter per second" ;
V_TIDE:type_tide = "Model_atlas_v1" ;
V_TIDE:long_name = "Northward tide Velocity" ;
V_TIDE:_FillValue = -999999.f ;
V_TIDE:valid_min = -20. ;
V_TIDE:valid_max = 20. ;
float **UVEL_ADCP**_CORTIDE(N_DATE_TIME, N_LEVEL) ; (**Eastward absolute velocity corrected for tide**)
UVEL_ADCP_CORTIDE:units = "meter per second" ;
UVEL_ADCP_CORTIDE:long_name = "Eastward absolute velocity corrected for tide" ;
UVEL_ADCP_CORTIDE:_FillValue = -999999.f ;

UVEL_ADCP_CORTIDE:valid_min = -20. ;
UVEL_ADCP_CORTIDE:valid_max = 20. ;
float **VVEL_ADCP_CORTIDE**(N_DATE_TIME, N_LEVEL) ; **(Northward absolute velocity corrected for tide)**
VVEL_ADCP_CORTIDE:units = "meter per second" ;
VVEL_ADCP_CORTIDE:long_name = "Northward absolute velocity corrected for tide" ;
VVEL_ADCP_CORTIDE:_FillValue = -999999.f ;
VVEL_ADCP_CORTIDE:valid_min = -20. ;
VVEL_ADCP_CORTIDE:valid_max = 20. ;
float **TU_TIDE**(N_DATE_TIME) ; **(Eastward tide Transport)**
TU_TIDE:units = "meter^2 per second" ;
TU_TIDE:long_name = "Eastward tide Transport" ;
TU_TIDE:_FillValue = -999999.f ;
TU_TIDE:valid_min = -20. ;
TU_TIDE:valid_max = 20. ;
float **TV_TIDE**(N_DATE_TIME) ; **(Northward tide Transport)**
TV_TIDE:units = "meter^2 per second" ;
TV_TIDE:long_name = "Northward tide Transport" ;
TV_TIDE:_FillValue = -999999.f ;
TV_TIDE:valid_min = -20. ;
TV_TIDE:valid_max = 20. ;

// **global attributes**:
:DATE_CREATION = "13-Dec-2016" ;
:SOFTWARE = "Cascade Exploitation V7.1" ;
:CPU_FIRMWARE_VERSION = 23. ;
:CPU_FIRMWARE_REVISION = 19. ;
:ADCP_FREQUENCY = 38. ;
:ADCP_BEAM_ANGLE = 30. ;
:ADCP_ANGLE_NAVIRE = 45. ;
:ADCP_FORM = "CONVEXE" ;
:ADCP_FACING = "DOWN" ;
:ADCP_CONSTRUCTOR = "RDI" ;
:ADCP_TYPE = "OceanSurveyor   " ;
:CONVENTIONS = "OceanSite dictionary" ;
:CRUISE_NAME = "RREX15" ;
:PLATFORM_NAME = "THALASSA" ;
:PROCESSING_CHIEF = "H. Mercier" ;
:DATE_UPDATE = "22-Dec-2016" ;
:DATA_TYPE = "SADCP" ;
:FORMAT_VERSION = "1.0" ;
:NAVIGATION_REFERENCE = "GPS Navigation" ;
:HEADING_REFERENCE = "GPS Heading" ;
:PLATFORM_NUMBER = "FNFP" ;
}

5.5 NetCDF section file

In this file, each data corresponds to an average over data from the original NetCDF cruise file. The data to be taken into account for the same average depends on how many kilometres the data are to be averaged over.

netcdf RREX_osite_sec_02xs1 {
dimensions:
    N_DATE_TIME = 3537 ; (**number of data**)
    N_LEVEL = 85 ; (**number of cell**)
    CONST1 = 1 ;
    DATE_TIME = 14 ;
    nbre_section = 35 ; (**number of section**)
variables:
    short **INDICE**(nbre_section) ; (**example: INDICE = [1 832 930]: the first section is associated with data 1 to 831, the second section with data 832 to 929, the last section with data 929 to tne end of the file)**
        INDICE:units = "indice" ;
        INDICE:long_name = "indice" ;
        INDICE:_FillValue = -999s ;
    char **REFERENCE_DATE_TIME**(DATE_TIME) ; (**Date of reference for julian days**)
        REFERENCE_DATE_TIME:long_name = "Date of reference for Julian days" ;
        REFERENCE_DATE_TIME:convention = "YYYYMMDDHHMISS" ;
        REFERENCE_DATE_TIME:_FillValue = " " ;
    double **JULD**(N_DATE_TIME) ; (**julian day**)
        JULD:long_name = "julian days relative to REFERENCE_DATE_TIME" ;
        JULD:_FillValue = -999999. ;
    float **SecLat**(N_DATE_TIME) ; (**latitude**)
        SecLat:units = "degree" ;
        SecLat:long_name = "Latitude" ;
        SecLat:_FillValue = -999999.f ;
    float **SecLon**(N_DATE_TIME) ; (**longitude**)
        SecLon:units = "degree" ;
        SecLon:long_name = "Longitude" ;
        SecLon:_FillValue = -999999.f ;
    float **U_TIDE**(N_DATE_TIME) ; (**Eastward tide Velocity**)
        U_TIDE:units = "meter per second" ;
        U_TIDE:long_name = "Eastward tide Velocity" ;
        U_TIDE:_FillValue = -999999.f ;
        U_TIDE:valid_min = -20. ;
        U_TIDE:valid_max = 20. ;
        U_TIDE:type_tide = "Model_tpxo7.2" ;
    float **V_TIDE**(N_DATE_TIME) ; (**Northward tide Velocity**)
        V_TIDE:units = "meter per second" ;
        V_TIDE:long_name = "Northward tide Velocity" ;
        V_TIDE:_FillValue = -999999.f ;
        V_TIDE:valid_min = -20. ;
        V_TIDE:valid_max = 20. ;
        V_TIDE:type_tide = "Model_tpxo7.2" ;
    float DEPH(N_LEVEL) ; (**Depth of bin center**)
        DEPH:units = "meter" ;
        DEPH:long_name = "Depth of bin center" ;
        DEPH:valid_min = -12000. ;
        DEPH:valid_max = 0. ;
        DEPH:_FillValue = -999999.f ;
    float **VVEL_ADCP**(N_DATE_TIME, N_LEVEL) ; (**Northward absolute ADCP current velocity**)
        VVEL_ADCP:units = "meter per second" ;
        VVEL_ADCP:long_name = "Northward absolute ADCP current velocity" ;
        VVEL_ADCP:valid_min = -20. ;
        VVEL_ADCP:valid_max = 20. ;

VVEL_ADCP:_FillValue = -999999.f ;
float **UVEL_ADCP**(N_DATE_TIME, N_LEVEL) ; **(Eastward absolute ADCP current velocity)**
UVEL_ADCP:units = "meter per second" ;
UVEL_ADCP:long_name = "Eastward absolute ADCP current velocity" ;
UVEL_ADCP:valid_min = -20. ;
UVEL_ADCP:valid_max = 20. ;
UVEL_ADCP:_FillValue = -999999.f ;
float **WVEL_ADCP**(N_DATE_TIME, N_LEVEL) ; (**Vertical absolute ADCP current velocity**)
WVEL_ADCP:units = "meter per second" ;
WVEL_ADCP:long_name = "Upward absolute ADCP current velocity" ;
WVEL_ADCP:valid_min = -20. ;
WVEL_ADCP:valid_max = 20. ;
WVEL_ADCP:_FillValue = -999999.f ;
float **BATHY**(N_DATE_TIME) ; (**Bathymetry**)
BATHY:units = "meter" ;
BATHY:comment = "" ;
BATHY:long_name = "" ;
BATHY:_FillValue = -999999.f ;
BATHY:valid_min = -12000. ;
BATHY:valid_max = 0. ;
short **NB_PTS**(N_DATE_TIME, N_LEVEL) ; (**Number of points used for averaging)**
NB_PTS:units = " " ;
NB_PTS:long_name = "Number of points used for averaging" ;
NB_PTS:valid_min = 0. ;
NB_PTS:valid_max = 50000. ;
NB_PTS:_FillValue = -999s ;
float **URMS_ADCP**(N_DATE_TIME, N_LEVEL) ; **(Root mean square Eastward velocity)**
URMS_ADCP:units = "meter per second" ;
URMS_ADCP:long_name = "Root mean square Eastward velocity" ;
URMS_ADCP:valid_min = -20. ;
URMS_ADCP:valid_max = 20. ;
URMS_ADCP:_FillValue = -999999.f ;
float **VRMS_ADCP**(N_DATE_TIME, N_LEVEL) ; **(Root mean square Northward velocity)**
VRMS_ADCP:units = "meter per second" ;
VRMS_ADCP:long_name = "Root mean square Northward velocity" ;
VRMS_ADCP:valid_min = -20. ;
VRMS_ADCP:valid_max = 20. ;
VRMS_ADCP:_FillValue = -999999.f ;
float **WRMS_ADCP**(N_DATE_TIME, N_LEVEL) ; **(Root mean square vertical velocity**)
WRMS_ADCP:units = "meter per second" ;
WRMS_ADCP:long_name = "Root mean square Upward velocity" ;
WRMS_ADCP:valid_min = -20. ;
WRMS_ADCP:valid_max = 20. ;
WRMS_ADCP:_FillValue = -999999.f ;
float **ECI**(N_DATE_TIME, N_LEVEL) ; (**Mean echo intensity**)
ECI:units = "count" ;
ECI:long_name = "Mean echo intensity" ;
ECI:_FillValue = -999999.f ;

// **global attributes**:
:DATE_CREATION = "06-Dec-2016" ;
:INST_TYPE = "RD Instruments ADCP" ;
:INST_MODEL = "narrow band" ;
:PROG_CMNT1 = "Section file" ;
:delta_distance = 2. ;
:type_vitesses = "Vitesses non corrigees de la maree" ;
:avec_ou_sans_station = "Stations prises en compte" ;
:flag_vitesses = "1" ;
}

5.6 NetCDF station file

In this file, each data corresponds to an average over data from the original NetCDF cruise file. The data to be taken into account for the same average depends on how long the data are to be averaged over the stations.

netcdf RREX_osite_sta_00600x1 {
dimensions:
  N_DATE_TIME = 1278 ; (**number of data**)
  N_LEVEL = 85 ; (**number of cell**)
  CONST1 = 1 ;
  DATE_TIME = 14 ;
  nbre_station = 136 ; (**number of station**)
variables:
  float **NUMERO**(nbre_station) ; (**station number**)
    NUMERO:long_name = "numero de station" ;
  short **INDICE**(nbre_station) ; ; (**example: INDICE = [1 10 17]: the first station is associated with data 1 to 9, the second station with data 10 to 1-, the last station with data 17 to the last data of the file)**
  INDICE:units = "indice" ;
    INDICE:long_name = "indice" ;
    INDICE:_FillValue = -999s ;
  char **REFERENCE_DATE_TIME**(DATE_TIME) ; **(Date of reference for Julian days)**
    REFERENCE_DATE_TIME:long_name = "Date of reference for Julian days" ;
    REFERENCE_DATE_TIME:convention = "YYYYMMDDHHMISS" ;
    REFERENCE_DATE_TIME:_FillValue = " " ;
  double **JULD**(N_DATE_TIME) ; (**Julian day**)
    JULD:long_name = "ulian days relative to REFERENCE_DATE_TIME" ;
    JULD:_FillValue = -999999. ;
  float **StaLat**(N_DATE_TIME) ; (**latitude**)
    StaLat:units = "degree" ;
    StaLat:long_name = "Latitude" ;
    StaLat:_FillValue = -999999.f ;
  float StaLon(N_DATE_TIME) ; (**longitude**)
    StaLon:units = "degree" ;
    StaLon:long_name = "Longitude" ;
    StaLon:_FillValue = -999999.f ;
  float **U_TIDE**(N_DATE_TIME) ; **(Eastward tide Velocity)**
    U_TIDE:units = "meter per second" ;
    U_TIDE:long_name = "Eastward tide Velocity" ;
    U_TIDE:_FillValue = -999999.f ;
    U_TIDE:valid_min = -20. ;
    U_TIDE:valid_max = 20. ;
    U_TIDE:type_tide = "Model_tpxo7.2" ;
  float **V_TIDE**(N_DATE_TIME) ; (**Northward tide Velocity)**
    V_TIDE:units = "meter per second" ;
    V_TIDE:long_name = "Northward tide Velocity" ;
    V_TIDE:_FillValue = -999999.f ;
    V_TIDE:valid_min = -20. ;
    V_TIDE:valid_max = 20. ;
    V_TIDE:type_tide = "Model_tpxo7.2" ;
  float **DEPH**(N_LEVEL) ; **(Depth of bin center)**
    DEPH:units = "meter" ;
    DEPH:long_name = "Depth of bin center" ;
    DEPH:valid_min = -12000. ;
    DEPH:valid_max = 0. ;
    DEPH:_FillValue = -999999.f ;
  float **VVEL_ADCP**(N_DATE_TIME, N_LEVEL) ; **(Northward absolute ADCP current velocity)**

```
        VVEL_ADCP:units = "meter per second" ;
        VVEL_ADCP:long_name = "Northward absolute ADCP current velocity" ;
        VVEL_ADCP:valid_min = -20. ;
        VVEL_ADCP:valid_max = 20. ;
        VVEL_ADCP:_FillValue = -999999.f ;
    float UVEL_ADCP(N_DATE_TIME, N_LEVEL) ; (Eastward absolute ADCP current velocity)
        UVEL_ADCP:units = "meter per second" ;
        UVEL_ADCP:long_name = "Eastward absolute ADCP current velocity" ;
        UVEL_ADCP:valid_min = -20. ;
        UVEL_ADCP:valid_max = 20. ;
        UVEL_ADCP:_FillValue = -999999.f ;
    float WVEL_ADCP(N_DATE_TIME, N_LEVEL) ; (Vertical absolute ADCP current velocity)
        WVEL_ADCP:units = "meter per second" ;
        WVEL_ADCP:long_name = "Upward absolute ADCP current velocity" ;
        WVEL_ADCP:valid_min = -20. ;
        WVEL_ADCP:valid_max = 20. ;
        WVEL_ADCP:_FillValue = -999999.f ;
    short NB_PTS(N_DATE_TIME, N_LEVEL) ; (Number of points used for averaging)
        NB_PTS:units = " " ;
        NB_PTS:long_name = "Number of points used for averaging" ;
        NB_PTS:valid_min = 0. ;
        NB_PTS:valid_max = 50000. ;
        NB_PTS:_FillValue = -999s ;
    float URMS_ADCP(N_DATE_TIME, N_LEVEL) ; (Root mean square Eastward velocity)
        URMS_ADCP:units = "meter per second" ;
        URMS_ADCP:long_name = "Root mean square Eastward velocity" ;
        URMS_ADCP:valid_min = -20. ;
        URMS_ADCP:valid_max = 20. ;
        URMS_ADCP:_FillValue = -999999.f ;
    float VRMS_ADCP(N_DATE_TIME, N_LEVEL) ; (Root mean square Northward velocity)
        VRMS_ADCP:units = "meter per second" ;
        VRMS_ADCP:long_name = "Root mean square Northward velocity" ;
        VRMS_ADCP:valid_min = -20. ;
        VRMS_ADCP:valid_max = 20. ;
        VRMS_ADCP:_FillValue = -999999.f ;
    float WRMS_ADCP(N_DATE_TIME, N_LEVEL) ; (Root mean square vertical velocity)
        WRMS_ADCP:units = "meter per second" ;
        WRMS_ADCP:long_name = "Root mean square Upward velocity" ;
        WRMS_ADCP:valid_min = -20. ;
        WRMS_ADCP:valid_max = 20. ;
        WRMS_ADCP:_FillValue = -999999.f ;
    float ECI(N_DATE_TIME, N_LEVEL) ; (Mean echo intensity)
        ECI:units = "count" ;
        ECI:long_name = "Mean echo intensity" ;
        ECI:_FillValue = -999999.f ;

// global attributes:
        :DATE_CREATION = "06-Dec-2016" ;
        :INST_TYPE = "RD Instruments ADCP" ;
        :INST_MODEL = "narrow band" ;
        :PROG_CMNT1 = "Station file" ;
        :duree_moyenne = 600. ;
        :type_vitesses = "Vitesses non corrigees de la maree" ;
        :flag_vitesses = "1" ;
}
```

5.7 NetCDF Bathymetry file

netcdf bathy_etopo1 {
dimensions:
    longitude = 21601 ; (**number of longitude**)
    latitude = 10801 ; (**number of latitude**)
variables:
    double **longitude**(longitude) ; (**longitude**)
        x:long_name = "Longitude" ;
        x:actual_range = -180., 180. ;
        x:units = "degrees_east" ;
    double **latitude**(latitude) ; (**latitude**)
        y:long_name = "Latitude" ;
        y:actual_range = -90., 90. ;
        y:units = "degrees_north" ;
    int **z**(latitude,longitude) ; (**bathymétrie: positive value for the continent, negative otherwise**)
        z:long_name = "z" ;
        z:_FillValue = -2147483648 ;
        z:actual_range = -10898., 8271. ;

// **global attributes**:
        :Conventions = "COARDS/CF-1.0" ;
        :title = "ETOPO1_Bed_g_gmt4.grd" ;
        :GMT_version = "4.4.0" ;
        :node_offset = 0 ;
}

5.8 External heading file

netcdf ovide_new_cap {
dimensions:
    N_DATE_TIME = 10987 ;
variables:
    double **JULD_EXT**(N_DATE_TIME) ; (**Julian day. The reference must be the same as the REFERENCE from the cruise NetCDF file)**
        JULD_EXT:long_name = "Julian days relative to REFERENCE_DATE_TIME" ;
        JULD_EXT:_FillValue = -999999. ;
    float **HDG_EXT**(N_DATE_TIME) ; (**ship heading**)
        HDG_EXT:long_name = "Ship heading" ;
        HDG_EXT:units = "degree" ;
        HDG_EXT:valid_min = -360.f ;
        HDG_EXT:valid_max = 360.f ;
        HDG_EXT:_FillValue = -999999.f ;
}

5.9 TRINAV file

netcdf ovidenavi {
dimensions:
    dim_trin = 230976 ;
    cte = 1 ;

// global attributes:
        :CREATION_DATE = "17-Mar-2005" ;
        :INST_TYPE = "TRINAV" ;

:PROG_CMNT1 = "Converted to netCDF via MATLAB by Cdf_trin_c" ;

variables:
double **T_TRIN**(dim_trin) ; (**Julia day. The reference must be the same as the REFERENCE from the cruise netCDF file)**
T_TRIN:units = "decimal days" ;
T_TRIN:long_name = "JULIAN DAYS TRINAV" ;
T_TRIN:_FillValue = -999999. ;
T_TRIN:valid_range = 2451545., 2461545. ;
float **VitU_TRIN**(dim_trin) ; (**ship's zonal velocity**)
VitU_TRIN:units = "m/s" ;
VitU_TRIN:long_name = "Vitesse navire filtree (U)" ;
VitU_TRIN:_FillValue = -999999.f ;
VitU_TRIN:valid_range = -15.f, 15.f ;
float VitV_TRIN(dim_trin) ; (**ship's meridian velocity**)
VitV_TRIN:units = "m/s" ;
VitV_TRIN:long_name = "Vitesse navire filtree (V)" ;
VitV_TRIN:_FillValue = -999999.f ;
VitV_TRIN:valid_range = -15.f, 15.f ;

}

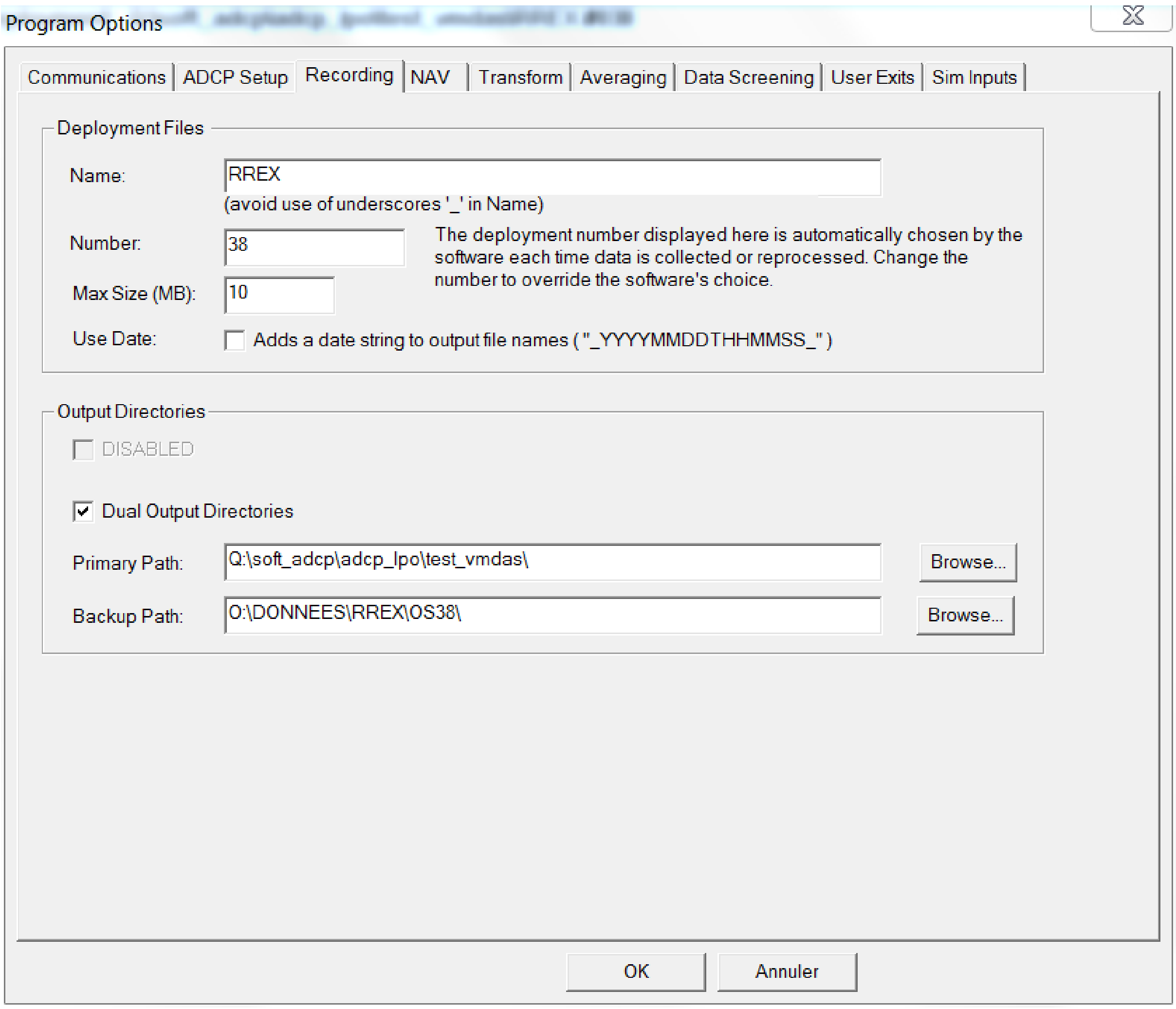
5.10 VmDas Configuration

During the ADCP acquisition on board, the user can configure several tabs of the VmDas acquisition software.

5.10.1 Recording Tab

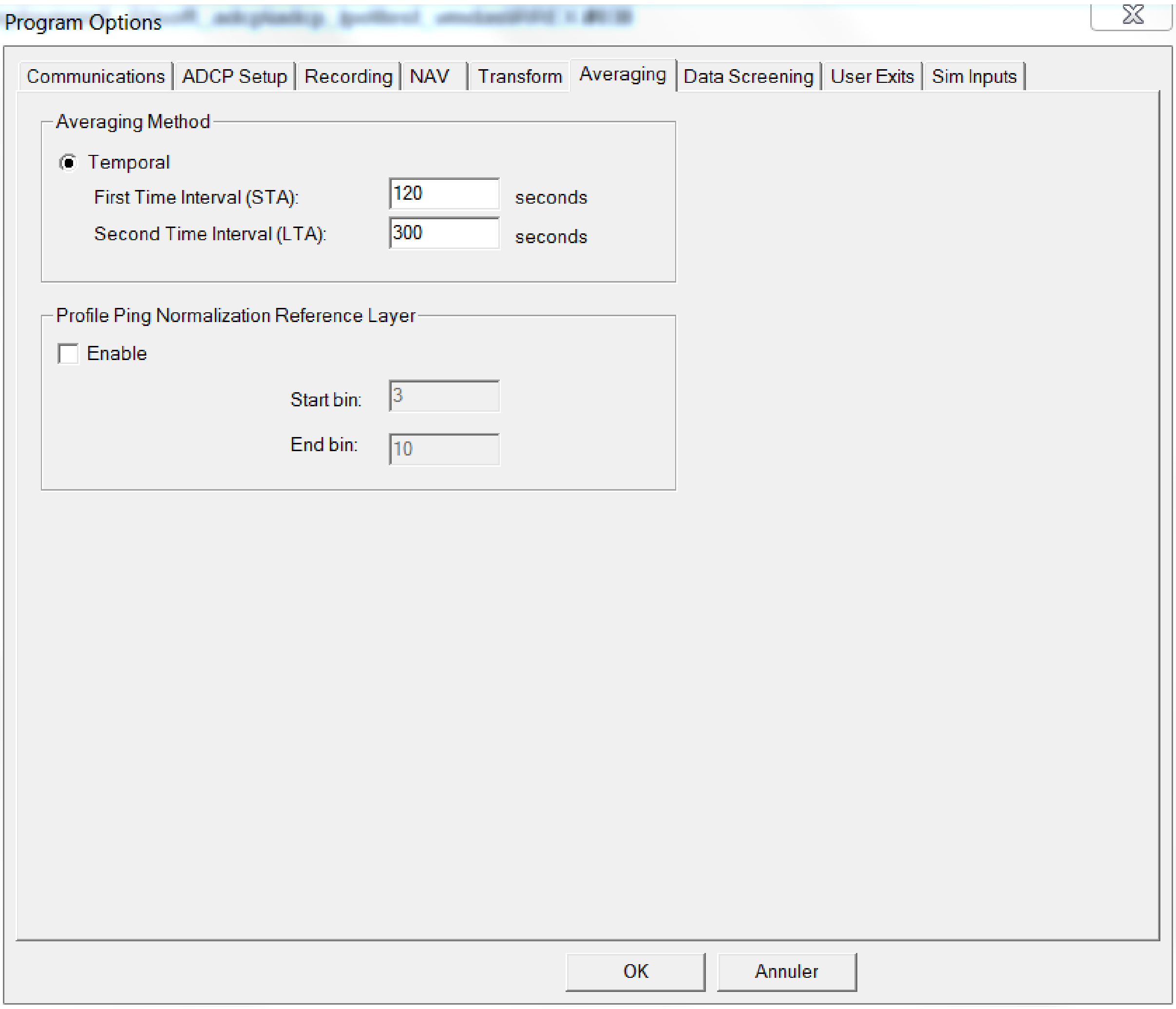
The user enters:

- The filenames's root for the STA and LTA files (‘Name’)
- The directory name for saving the STA and LTA files (‘Primary Path’ et ‘Backup Path’)



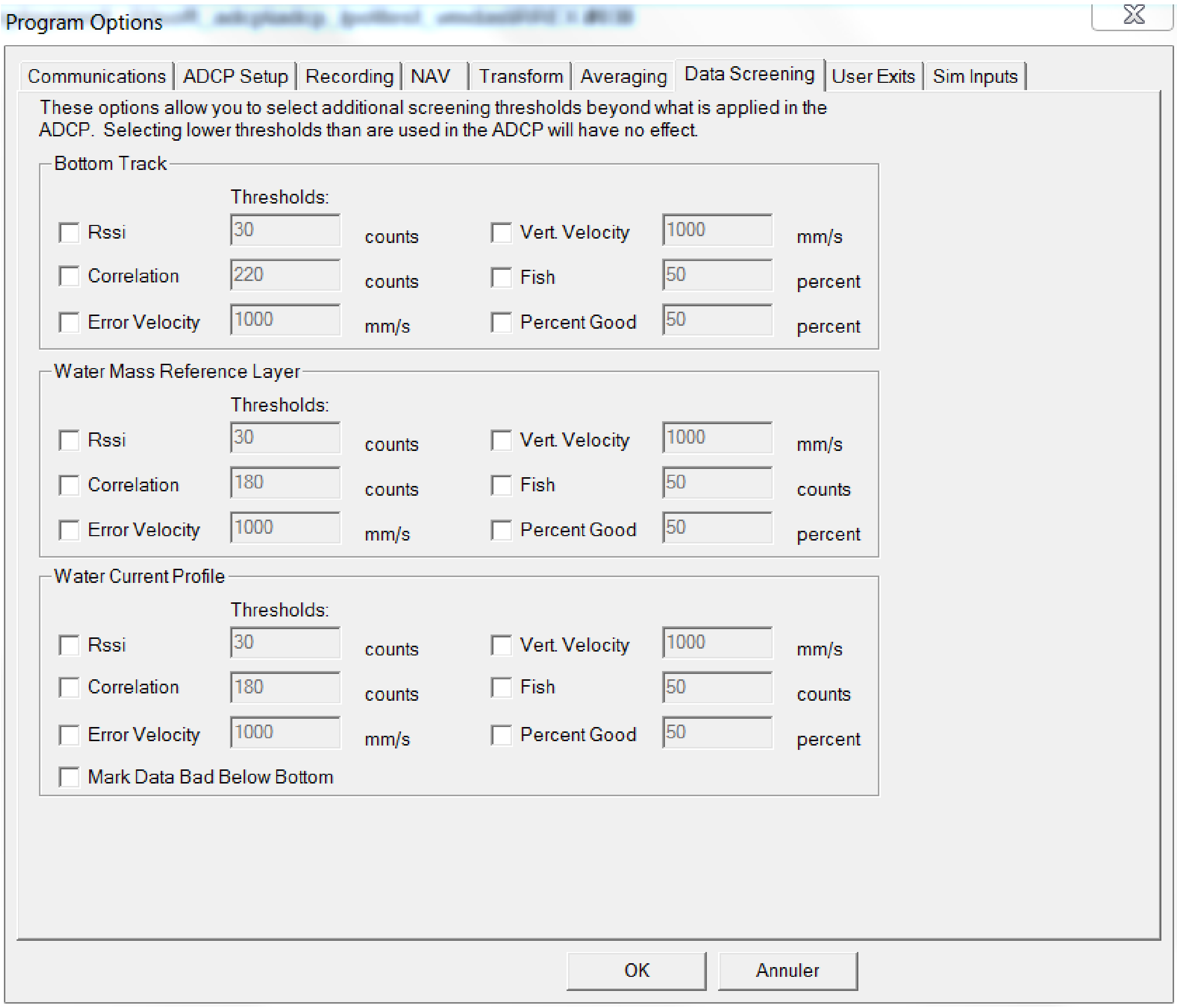
5.10.2 Averaging Tab

In this tab, the user indicates the duration of the STA and LTA files.



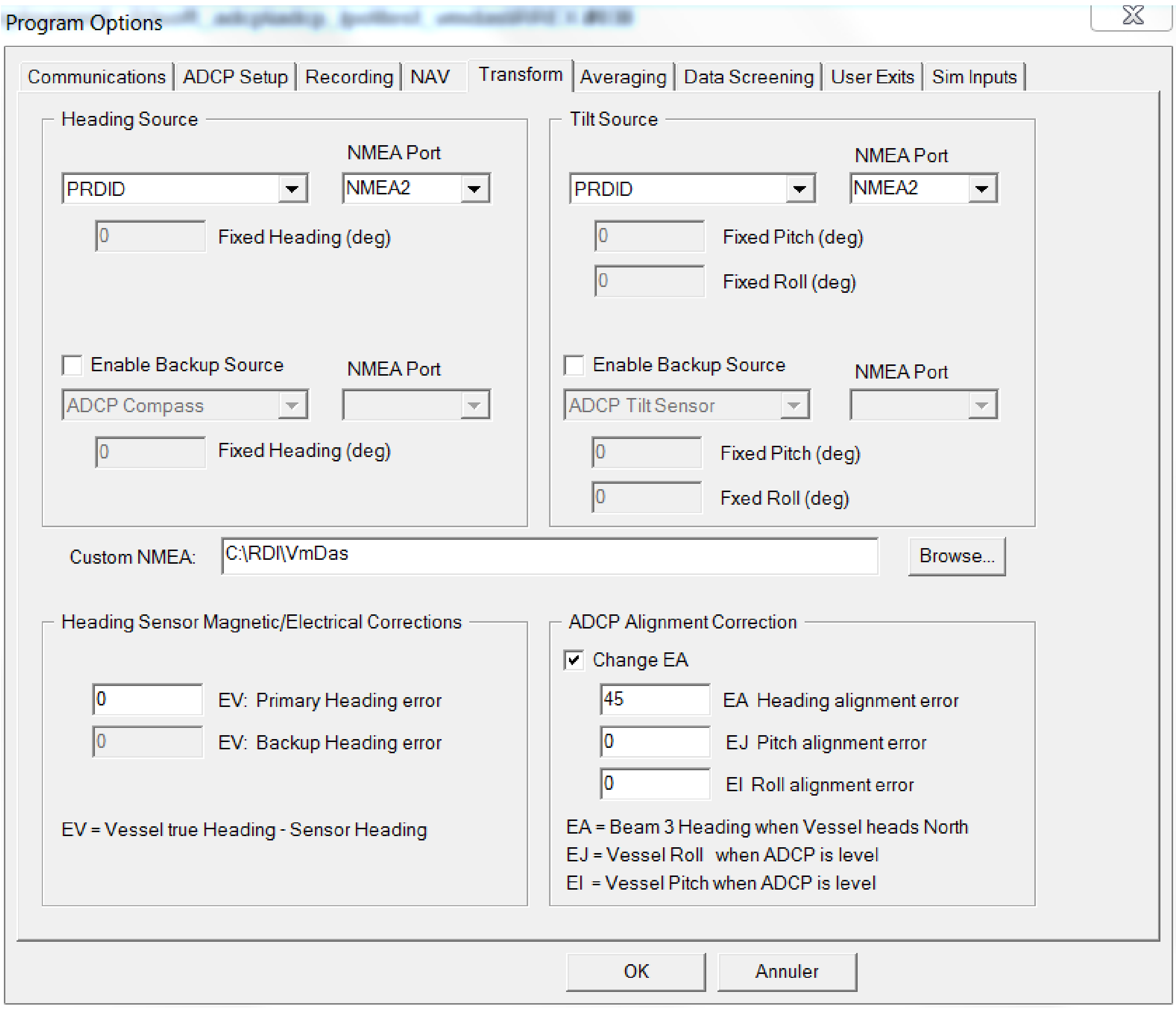
5.10.3 DataScreening Tab

In this tab, the user indicates cleaning thresholds to be applied to the data acquisition and thus to the generation of STA and LTA files. Note that these thresholds are currently ineffective when the acquisition is in NarrowBand mode.
It is recommended not to activate any threshold during acquisition and to prefer the qualification of the data during processing with CASCADE (3.3.3.5).



5.10.4 Transform Tab

On Genavir ships, this tab is managed by the ship's electronics engineer. It can nevertheless be verified that the angle of alignment of the ADCP in relation to the vessel ('EA') is close to 45° and is identical to the same angle also defined in the configuration file in text format.



5.10.5 Example configuration file

5.10.5.1 Example for low-bottomed areas

;---\
; ADCP Command File for use with VmDas software.
;
; ADCP type: 38 Khz Ocean Surveyor THALASSA
;
; ADCP 38 TL PETIT FOND TYPE
; VERSION 1.0
; LAST UPDATE 28 JUILLET 2011
;---/

; Restore factory default settings in the ADCP
cr1

; set the data collection baud rate to 38400 bps,
; avec la nouvelle version de VmDAS la commande cb811 qui
;met le baud rate à 115200 bps est lancée automatiquement

cb611

; Set for broadband single-ping profile mode (WP),
; WPnnn: number of ping for broadband mode per ensemble
; WP1 = monoping

WP1

; WNnnn: number of cell for broad band mode(de 001 à 128)
; WN070 = 70 cellules broad band

WN085

; WSnnnn: depth cell for boad band mode (cm)

WS2400

; WFnnnn: blank length (cm)
; WF1600 = 16m

WF1600

; WVnnn: threshold on radial velocity (cm/s) By default, for broadband, 390 cm/s

WV390

; Set for narrowband single-ping profile mode (NP)
;NPnnn: number of poing per ensemble in narraowband mode
;NP000 = no narrowband ping (=broad band configuration)

NP000

; NNnnn: number of bins for narrow band mode (de 001 à 128)

NN080

; NSnnnn: cell length for narrow band mode (cm)
; NS1600 = 16m

NS1600

; NFnnnn: blank length (cm)
; NF1600 = 16m

NF1600

CW0

; SYNCHRO externe
; CXa,b définit les trigger in et out
; a = 0 trigger in off, master mode or autonomous
; a = 1 trigger in on on rising edge, slave mode
; b = 0 trigger output off, slave mode or autonomous
; b = 1 trigger output, master mode
; cx0,1: adcp master mode or autonomous

cx0,1

; Enable single-ping bottom track (BP),
; "bottom ping" emission (1 = Yes, 0 = No) :
; BPn: number of bottom-ping per ensemble
; BP1 = 1 ping de bottom-track par ensemble

BP01

```
; BXnnnn maximum bottom search depth (dm)
; BX20000 = 2000 m
BX20000


; output velocity, correlation, echo intensity, percent good
; WDadcde0000

WD111100000

; TPmmssff: time between pings  (minute seconde centième)
; TP000200 = 2s

TP000300

; TEhhmmssff: time between ensembme (heure minute seconde centieme)
; TE00000400 = 4s

TE00000600

; Set to calculate speed-of-sound, no depth sensor, external synchro heading

EZ10000010

; Output beam data (rotations are done in software)
EX00000

; Set transducer misalignment (hundredths of degrees)
;EA04500 = 45.00°
; OCTANS 1 45,0°  EA04500 / OCTANS 2 44.9° EA04490

EA04500

; Set transducer depth to (decimeters)
; ED00060 = base à 6,0m de profondeur

ED00060

; Set Salinity (ppt)
ES35

; save this setup to non-volatile memory in the ADCP
CK
```

5.10.5.2 Example for deep-sea areas

```
;------------------------------------------------------------------------------\
; ADCP Command File for use with VmDas software.
;
; ADCP type:     38 Khz Ocean Surveyor THALASSA
;
; ADCP 38 TL PETIT FOND TYPE
; VERSION 1.0
; LAST UPDATE 28 JUILLET 2011
;------------------------------------------------------------------------------/

; Restore factory default settings in the ADCP
cr1

; set the data collection baud rate to 38400 bps,
; avec la nouvelle version de VmDAS la commande cb811 qui
;met le baud rate à 115200 bps est lancée automatiquement

cb611
```

```
; Set for broadband single-ping profile mode (WP),
WP0
WN055


WS2400
WF1600
WV390
NP001
NN085
NS2400
NF1600
CW0
cx0,1
BP00
BX20000
WD111100000
TP000300
TE00000300
EZ10000010
EX00000
EA04500
ED00060
ES35
CK
```

5.11 Abbreviations et acronyms

BB: Broad band
CASCADE: Chaine Automatisée de Suivi des Courantomètres Acoustiques Doppler Embarqués
LOPS: Laboratoire d'Océanographie Physique et Spatiale (Ifremer, Plouzané, France)
LTA: Long-term averaged
NetCDF: Network Common Data Form
NB: Narrow Band (mode d'émission en bande étroite)
OS: Ocean Surveyor
SADCP: Ship Mounted Acoustic Doppler Current Profiler (aussi appelé VM-ADCP)
STA: Short-term averaged
VM-ADCP: Vessel Mounted Acoustic Doppler Current Profiler (aussi appelé SADCP)
VmDas: Vessel-mount Data Acquisition System