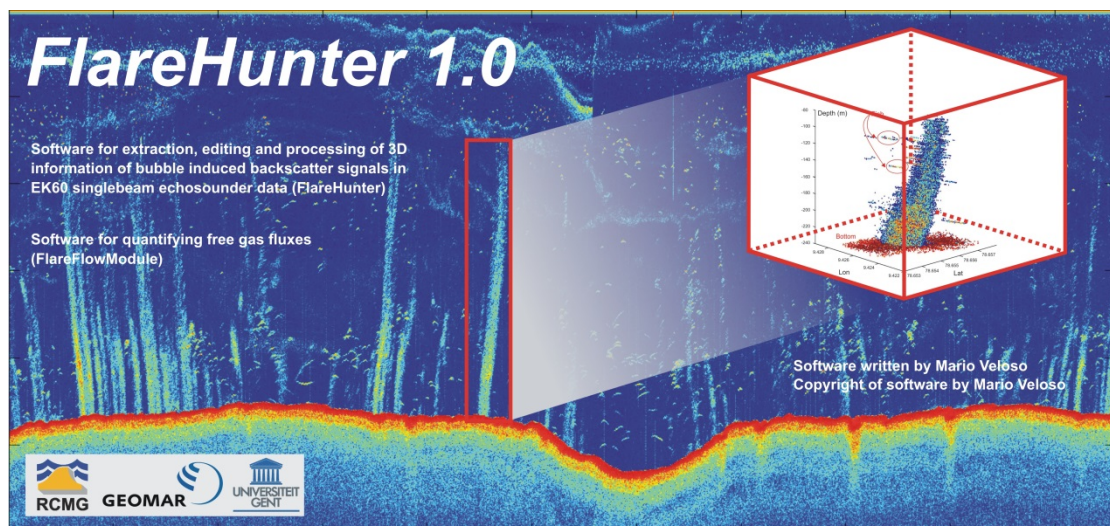


FlareHunter & FluxFlowModel

Software for extraction, editing and processing of 3D information of bubble induced backscatter signals in EK60 singlebeam echosounder data
(*FlareHunter*)

Software for quantifying free gas fluxes (*FlareFlowModule*)



Software written by Mario Veloso

Copyright of software by Mario Veloso

Software based on MATLAB; image analyses tool box needed

User manual - FlareHunter GUI

Table of Contents

1. Introduction	4
2. Capabilities	4
3. Availability	5
4. Requirements	5
5. Data reading	6
5.1 Data file format	6
5.2 Encapsulation	7
6. Quick start	8
7. Starting the GUI	8
7.1 File selection and load data	9
7.2 GPS offset input	9
7.3 Selection of available frequencies	10
7.4 Sound velocity	10
7.5 S_v, TS and TS_{COMP} calculation	11
7.6 Echograms and mechanical angles	12
7.7 Configuration display	15
7.8 Coordinates, date and time info	16
7.9 GPS track	16
7.10 Motion and heading	17
7.11 Bottom detection	18
7.12 Zoom	19
7.13 Spatial distribution of backscattering	19
7.14 3D visualization tools	22
7.15 Flare spine	23
8. FlareFlowModule	26
8.1 Loading edited flares	27
8.2 Layer thickness selection of the flare source	28
8.3 Data gridding and interpolation	28

8.4 Estimation of *K*value29

8.5 Flow rate estimation33

9. References34

1. Introduction

The Graphical User Interface '*FlareHunter*' has been designed to provide users with an easy-to-use interface to visualize, analyze and process hydroacoustic data showing bubble release in echograms. The GUI is a specialized tool to post-process backscattering of free gas (bubbles) recorded with EK60 scientific echosounder systems (split-beam) of Kongsberg. The software was named '*FlareHunter*' because hydroacoustic surveys that target methane seepage activity in a study area offshore Prins Karl Forland (Svalbard) were unofficially called '*FlareHunting*'. The scripts of the GUI were developed in the MATLAB programming platform.

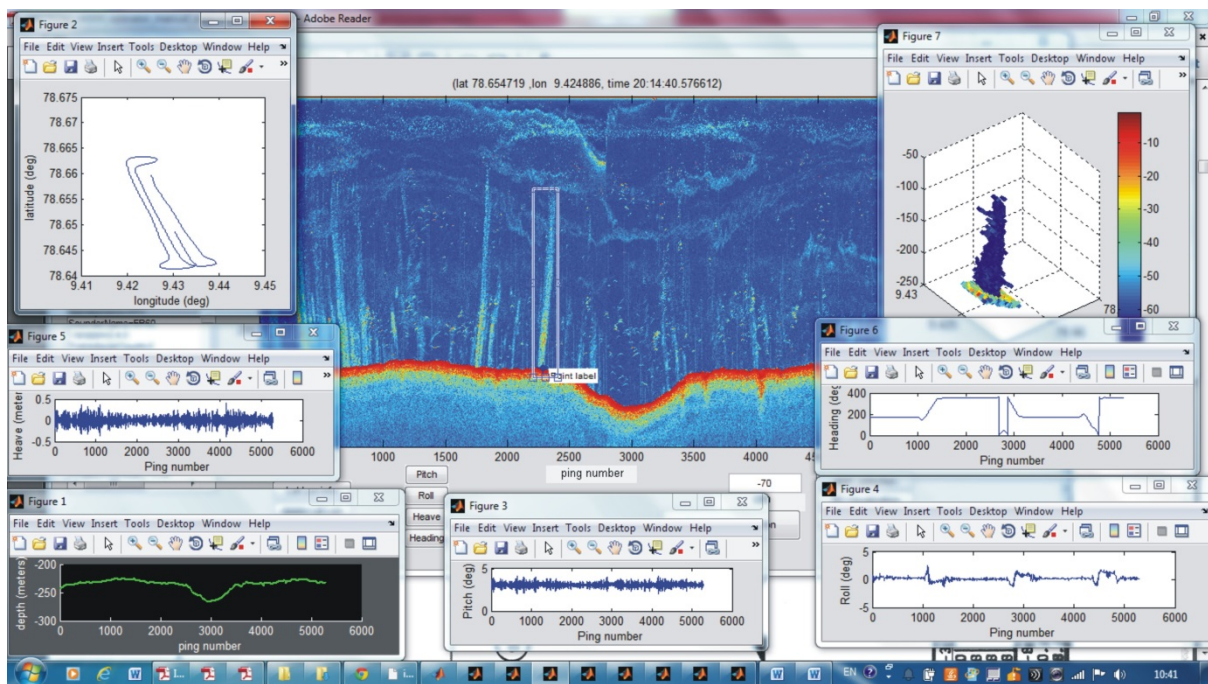


Figure 1. Screenshot of *FlareHunter* GUI.

2. Capabilities

The current version of *FlareHunter* GUI has the following capabilities:

- Reading EK60 data in the .raw format.
- Display of echograms of volume backscattering strength (S_V), target strength (TS) and target strength including beam compensation (TS_{COMP}).
- Display of images containing information of the mechanical angles (alongship and athwarship angles)
- Bottom detection
- Three dimensional visualization of S_V positions by accounting motion compensation and static offset corrections
- Manual editing of backscatter values in 3D and saving of these edited data.
- Geometrical averaging of scatterer positions to source location at the seafloor and observation of water current effects.

- *FlareFlowModule* to calculate the methane flow rate of single edited flares

3. Availability

MATLAB scripts of the *FlareHunter* and *FlareFlowModule* GUIs can be downloaded as compressed folder from this ftp site:

ftpaddress

The complete list of supplied MATLAB scripts is:

Table 1. List of scripts of *Flarehunter* and *FlareFlowModule*

FlareHunter folder	FlareFlowModule Functions folder
backscatter_visualization.m	BRS_selection.m
bottom_detection.m	BSD_selection.m
cell2str.m	buoyvel.m
circle.m	calculates_K.m
Convert_W_NT_time_format.m	caps.m
deg2utm.m	data_gridding.m
FlareHunter.fig	distance_footprint.m
FlareHunter.m	flares_file_selection.m
FlareFlowModule.fig	flow_estimation.m
FlareFlowModule.m	gausjoin.m
load_file.m	models_BRS.m
nangeomean.m	physcon.m
savingdata.m	polynom_fit_BRS.m
savingspine.m	polynom_fit_BSD.m
spine.m	viscosity.m
struct_config_text.m	
Sv_TS_calculation.m	
utm2degx.m	
viewer.m	
xyz_backst.m	

For questions please contact:

Mario Veloso (MarioEnrique.VelosoAlarcon@Ugent.com) or

Jens Greinert (jgreinert@geomar.de)

4. Software requirements

FlareHunter and the *FlareFlowModule* have been tested for MATLAB 7.8 (R2009a) and MATLAB 8.1 (R2013a) and MATLAB 8.3 (2014a)

!!To run the *FlareHunterGUI*, the **MATLAB Image Processing Toolbox** is required.

!!To run the *FlareFlowModuleGUI*, the **Generic Mapping Tools (GMT)** is required and needs to be installed.

5. Data Input

Data input is based on the file structure of .raw files recorded with the ER60 software of Kongsberg (Simrad, 2012). A brief description of the data file format is given below. For more detail see the ER60 manual.

5.1 Data file format

Simrad EK60 system saves the information in binary files (*.raw files). The files are composed of datagrams written in different data type formats. Each data type format can be described using common C structures (Table 1):

Table 2. Size of the used 'C' types (ER60 reference manual, 2008)

Char	8-bit integer
WORD	16-bit unsigned integer
Short	16-bit integer
Int	32-bit integer
Long	32-bit integer
Float	32-bit floating point (IEEE 754)
Double	64-bit floating point (IEEE 754)
DWORDLONG	64-bit integer

!!The following datagram types need to be part of the binary .raw files:

- **Configuration**

This datagram contains the information of the configuration of the echosounder system (transducer frequency, beam angle, etc.). The information is at the beginning of the *.raw file.

- **NMEA**

This datagram contains original NMEA 0183 input sentences. Here information like position (NMEA: GGA, GLL, RMC), ship motion (e.g. NMEA: VTG), ship orientation (NMEA: HDG, HDT) can be found. These sentences are written in ASCII.

Position time does not coincide with the ping time (obtained from the *Sample* datagram) and therefore positions are not specified for each ping. To obtain the ping positions, the position information is linearly interpolated according to the number of pings to obtain the corresponding position for each ping. Because the transducers

ping simultaneously and are relatively close to each other, they all share the same position

- **Annotation**

This datagram contains comment text. The sentences are written in ASCII.

- **Sample**

The sample datagram contains data from just one transducer channel. It can contain power sample data, or both power and angle sample data (split-beam electrical angles). **Motion information (pitch, roll, heading and heave) are also part of the sample datagram.** Each *.raw file contains information for a certain number of pings. In case there was more than one transducer connected to the EK60 echosounder, each transducer/frequency will have its own sample datagram.

5.2 Encapsulation

All data files use the same encapsulation structure. Datagramms are preceded by a four byte number which gives the datagram length (also in bytes) and an identical length tag is appended at the end of each datagram. The structure looks as follows:

```
long Length;
struct DatagramHeader
{
    long DatagramType;
    struct {
        long LowDateTime;
        long HighDateTime;
    } DateTime;
};
--
<datagram content
--
long Length;
};
```

Datagramms use the same header. The datagram type field is used to identify the type of datagram. These fields are ASCII quadruples; the first three characters are used to identify the datagram type and the last one the version of the datagram.

6. *FlareHunter* workflow

As mentioned above, *FlareHunter* has been designed to read and process hydroacoustic data that contain backscatter information coming from free gas/bubbles in the water column (flares; see Veloso et al. 2014; Limnology and Oceanography Methods). The order of reading data and processing is summarized in the flow chart in Figure 2. To successfully reading and processing the data, it is recommended to follow the order shown in the flow chart.

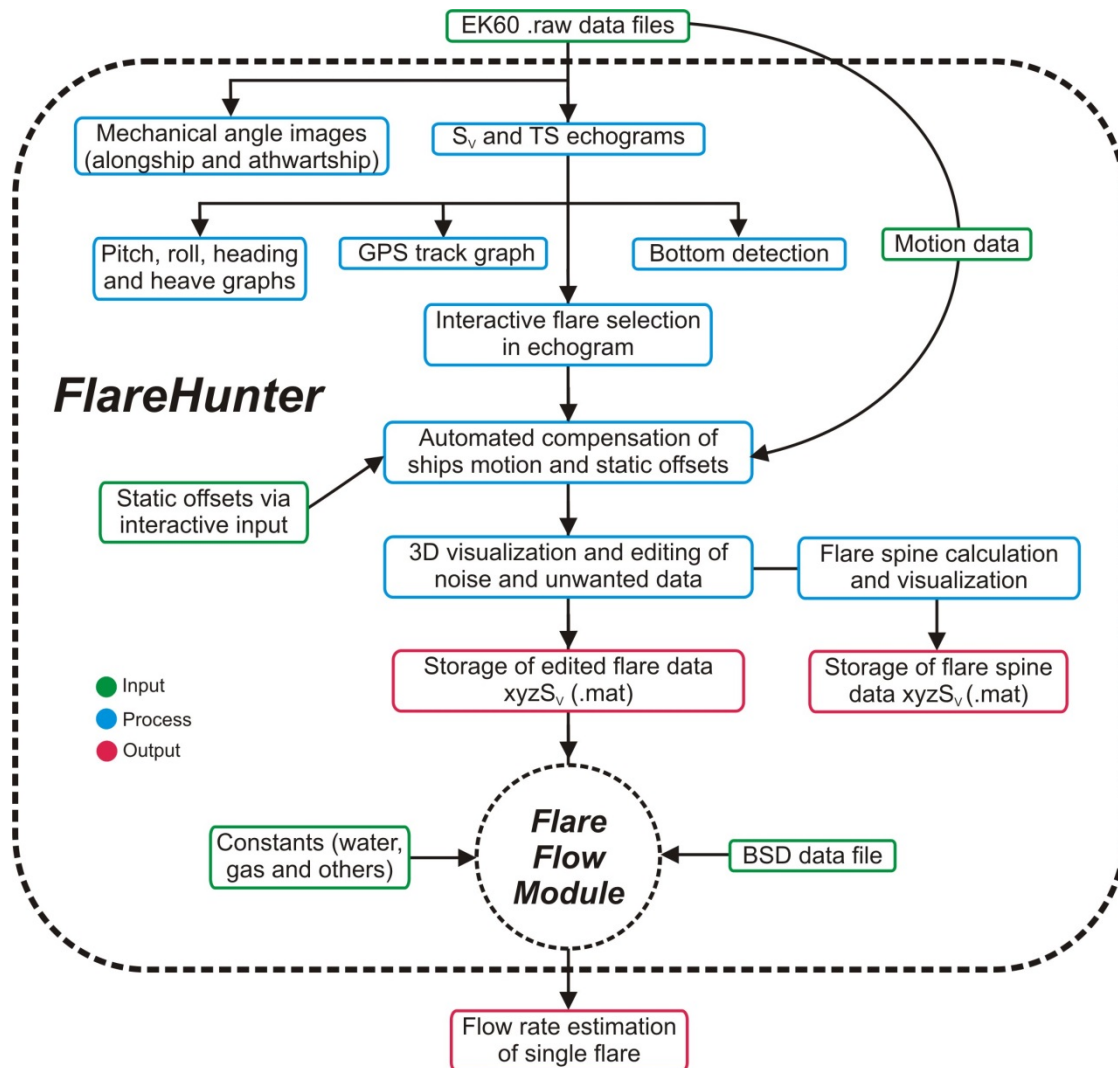


Figure1. *FlareHunter* Flowchart

7. Starting the *FlareHunter* GUI

Once the *FlareHunter* folder is uncompressed and MATLAB has launched, the *FlareHunter* GUI can be started by typing 'FlareHunter' at the Command Window of MATLAB. To start the GUI the Current Directory of the MATLAB Command Window must be the folder where the *FlareHunter* scripts are located.



Figure 2. Screenshot of the MATLAB Command Window and the starter command to open the *FlareHunter* GUI.

The *FlareHunter* GUI window is launched (Figure 3) and used to load, display, process and export data.

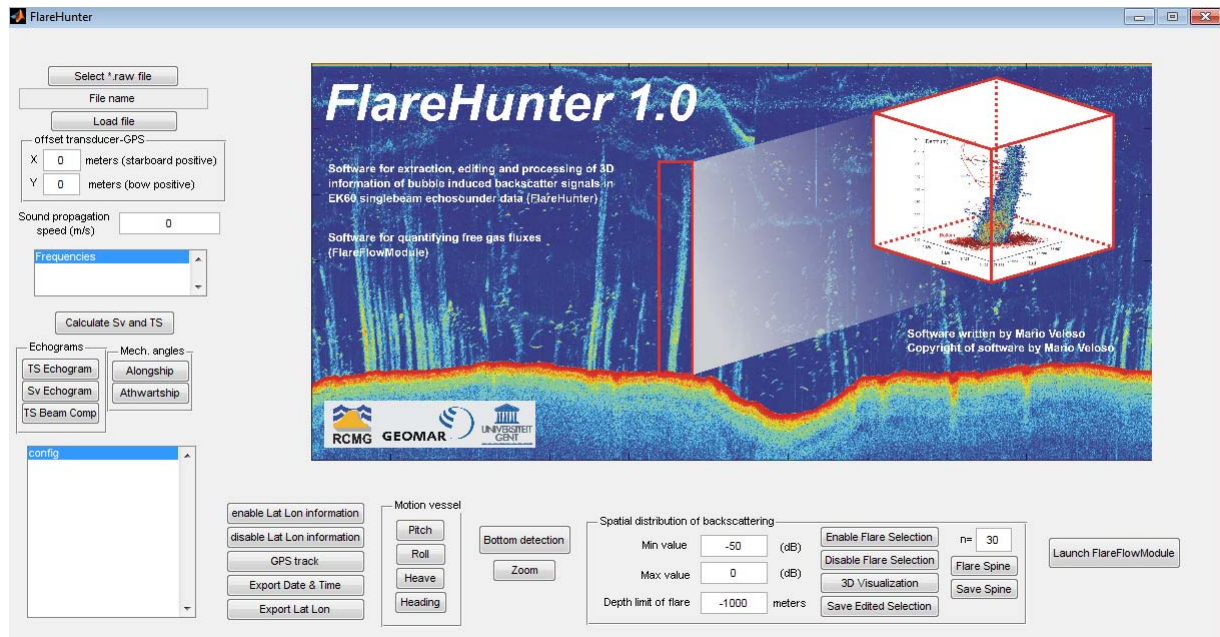


Figure 3. Screenshot of the *FlareHunter* GUI window.

7.1 File selection and load data

For processing, only one file at a time can be selected via the *Select *.raw file* button and loaded (read) by pressing the *Load file* button (Fig. 4). The .raw data files must be located in the same directory as the *FlareHunterscripts*.



Figure 4. Screenshot of the Select file and Load file push buttons. a) Image shows buttons before the file selection. b) Image shows how the file name is displayed after the file selection

7.2 GPS static offset correction

Lateral offsets between the absolute latitude and longitude positing recorded in the position datagram of the .raw file and the actually position of the transducer can be entered into the respective text fields (Figure 5). These static offsets are

then included for the 3D reconstruction of the exact location of the backscattering source (Figure 27). The inputs are in meters with X being positive towards starboard and Y being positive towards the bow of the ship.

offset transducer-GPS	
X	0 meters (starboard positive)
Y	0 meters (bow positive)

Figure 5. Screenshot of the GPS offset input table.

7.3 Selection of available frequencies

Once a data file is loaded, the *Frequencies window* (Figure 6a) will display the available frequencies that can be analyzed.

!! The list of frequencies is displayed (Figure 6b) after clicking once on the word *Frequencies*.



Figure 6. Screenshots of the Frequency window (a) with the given frequencies (b).

7.4 Sound velocity

The sound velocity box displays the mean sound velocity stored in the raw data automatically after loading the data file (Figure 7).

Figure 7. Sound speed propagation box

The value can be changed by entering a new value. The value in the text box is used for all calculations.

!! In case the sound speed is replaced, calculations of S_V , TS and TS_{COMP} (see section 7.5) must be done again

7.5 S_V , TS and TS_{COMP} calculation

The received backscattered signal can be displayed as TS , TS_{COMP} and S_V values after calculation. TS and S_V values are obtained using the following equations (Echoview formulas verified by SIMRAD derived from the online support of Echoview):

$$TS = P_r + 40 \log R + 2\alpha R - 10 \log \left(\frac{P_t G_0^2 \lambda^2}{16\pi^2} \right) \quad [1]$$

$$S_V = P_r + 20 \log R + 2\alpha R - 10 \log \left(\frac{P_t G_0^2 \lambda^2 c \tau \psi}{32\pi^2} \right) - 2S_a \quad [2]$$

where:

- R : corrected range (m) = $r - sT$
- r : uncorrected range (m) = $nc(\tau/2)$
- s : TVG range correction offset value
- T : sample thickness (m) = $c(\tau/2)$
- n : sample number
- P_r : received power (dB re 1 W)
- P_t : transmitted power (W)
- α : absorption coefficient (dB/m)
- G_0 : transducer peak gain (non-dimensional)
- λ : wavelength (m) = c/f
- f : frequency (Hz)
- c : sound speed (m/s)
- τ : transmit pulse duration (s) - also known as the pulse length
- ψ : Equivalent Two-way beam angle (Steradians)
- S_a : Simrad correction factor (dB re 1m^{-1}) determined during calibration of the EK60

TS_{COMP} is calculated using the following equation:

$$TS_{COMP} = TS + COMP \quad [3]$$

where:

$$COMP = 6.0206(x^2 + y^2 - 0.18x^2y^2)$$

$$x = \alpha_M / BW\alpha \quad (-1 \leq x \leq 1)$$

$$y = \beta_M / BW\beta \quad (-1 \leq y \leq 1)$$

and:

- α_M : Alongship mechanical angle (degrees)

β_M : Athwartship mechanical angle (degrees)
 $BW\alpha$: Beam width alongship (degrees)
 $BW\beta$: Beam width athwartship (degrees)

The calculation of these values is done after pressing the *Calculate Sv and TS* button (Figure 8). After the calculation is finished, a message is shown in the MATLAB Command Window (Figure 8b) and the respective data are ready to be displayed in the main echogram view.

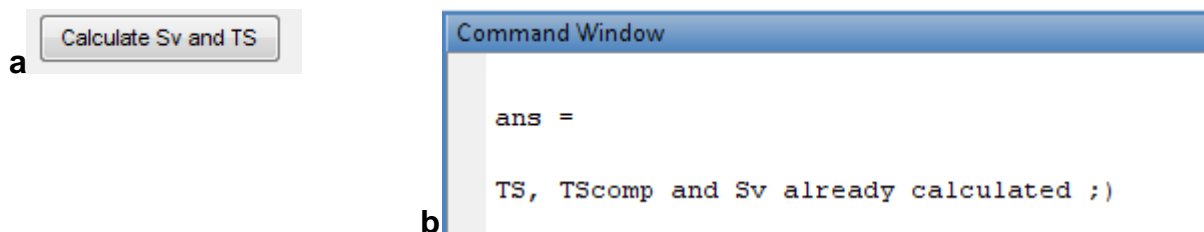


Figure 8. Screenshots of the “Calculation Sv and TS” push button (a), and the notation in the MATLAB command window in a successful end of the process.

7.6 Echograms and mechanical angles

Once TS , TS_{COMP} , and S_V values are calculated, they can be graphically displayed in the main echogram view. Echograms are consecutive traces of each ping color coded according to the amplitude of the signal. TS , S_V and TS_{COMP} echograms can be displayed by pushing the respective button (Figure 9). Additionally, images of the mechanical angles can be displayed by pushing the *Alongship* or *Athwartship* button. Figure 10 to 14 show the different echograms.

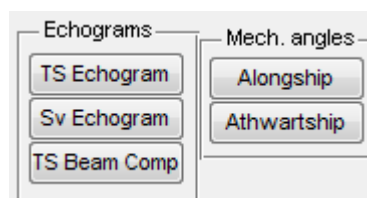


Figure 9. Screenshot of the “Echograms” and “Mechanical angles” push buttons.

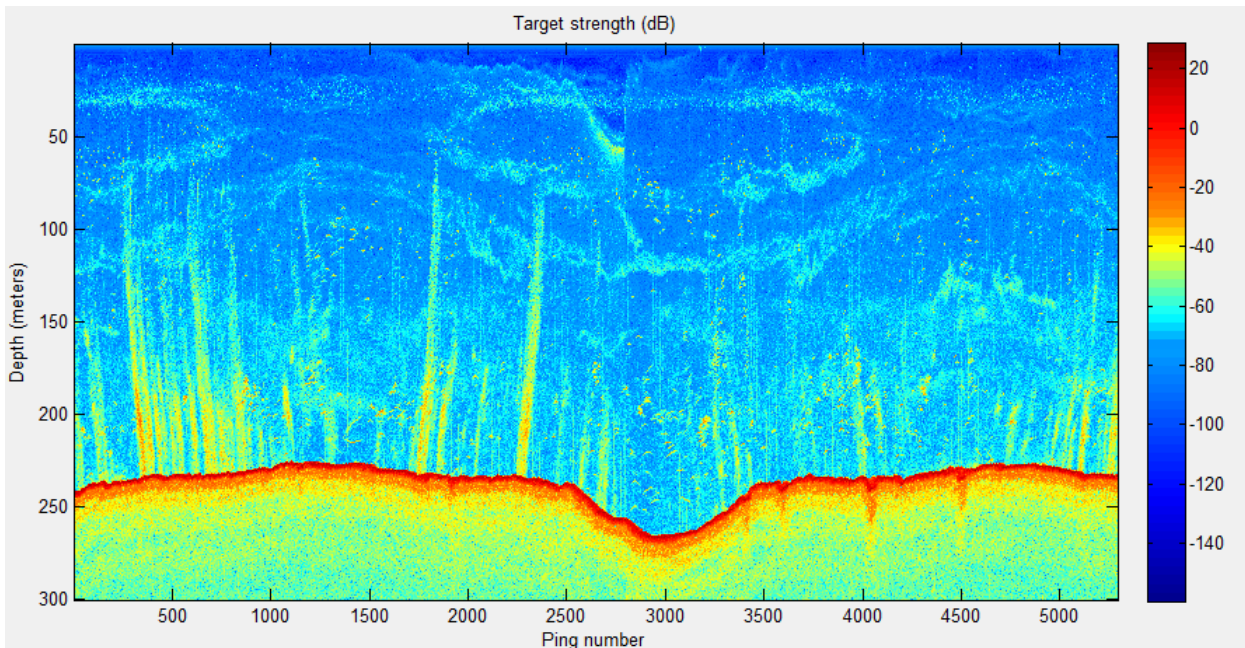


Figure 10. The TS echogram representation

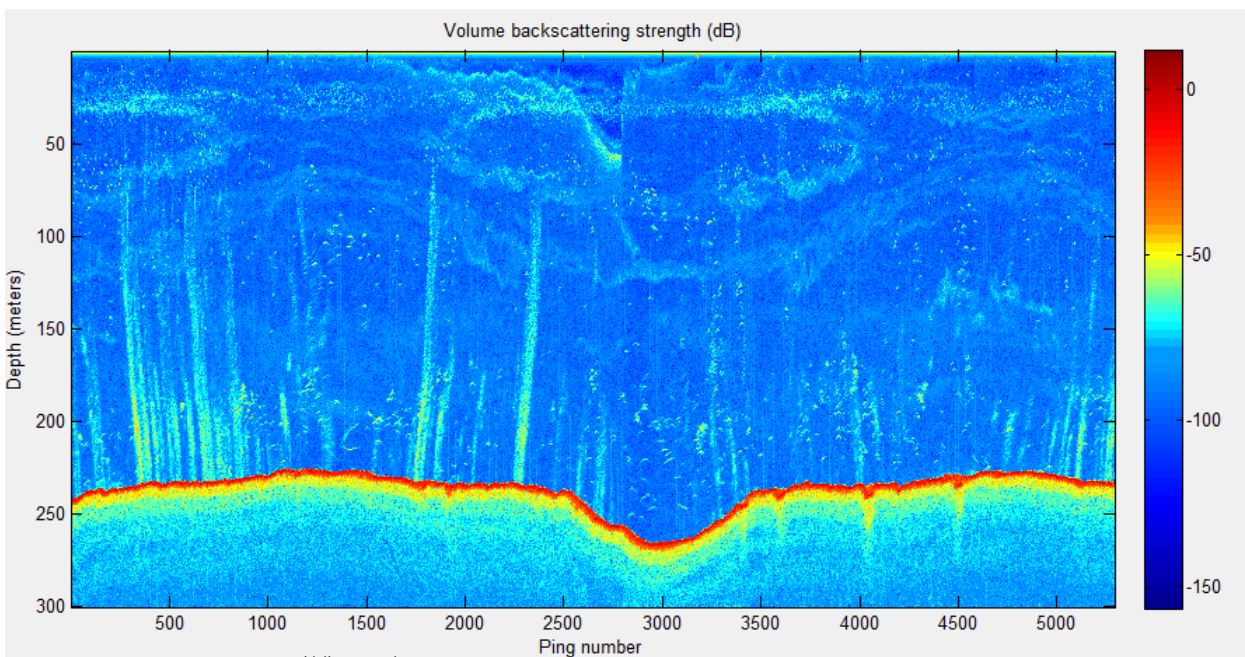


Figure 11. The S_V echogram representation.

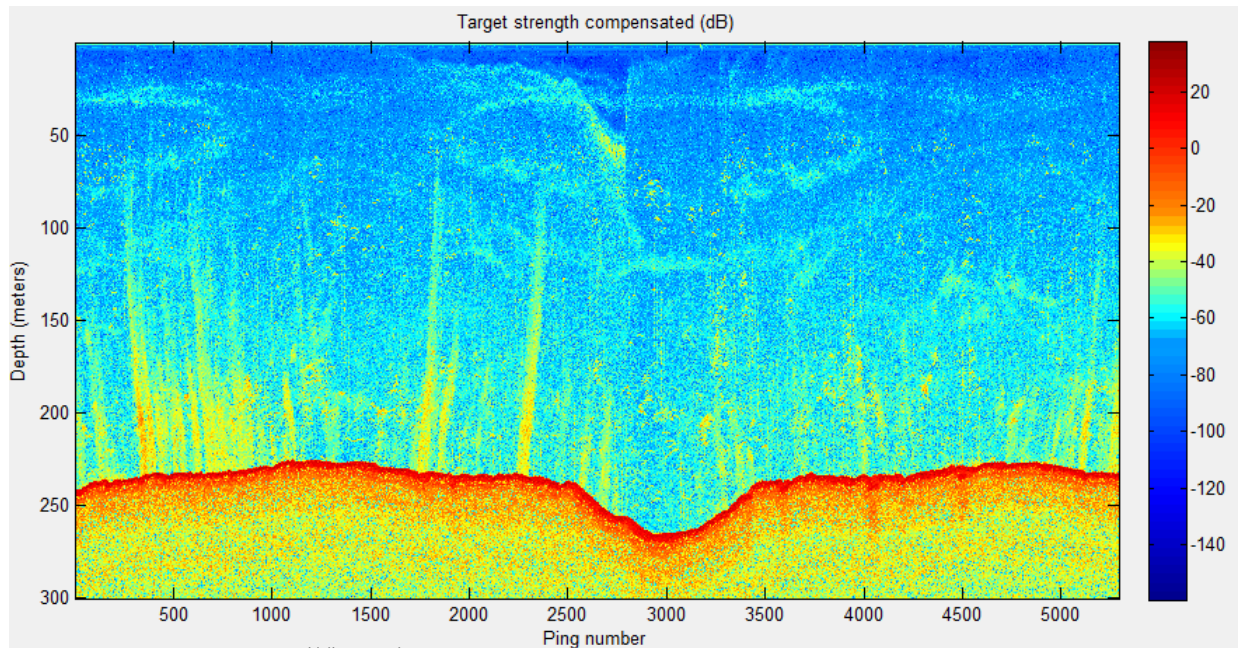


Figure 12. The *TS* beam comp representation.

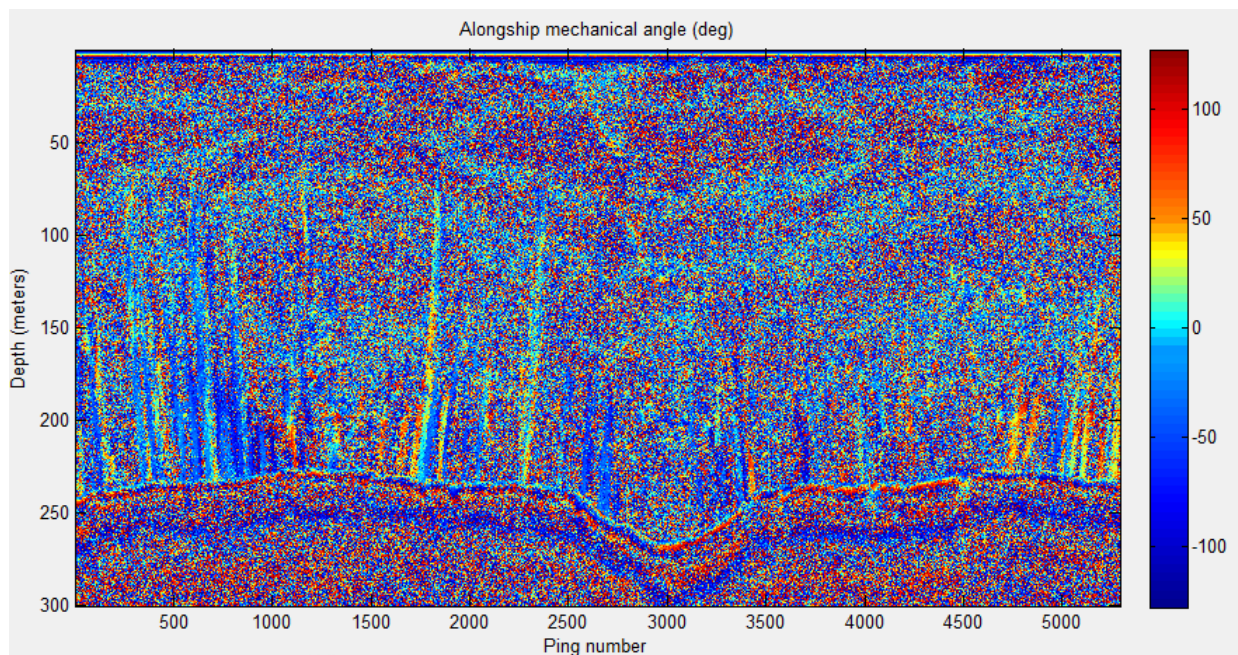


Figure 13. Image of mechanical angles –*Alongship*.

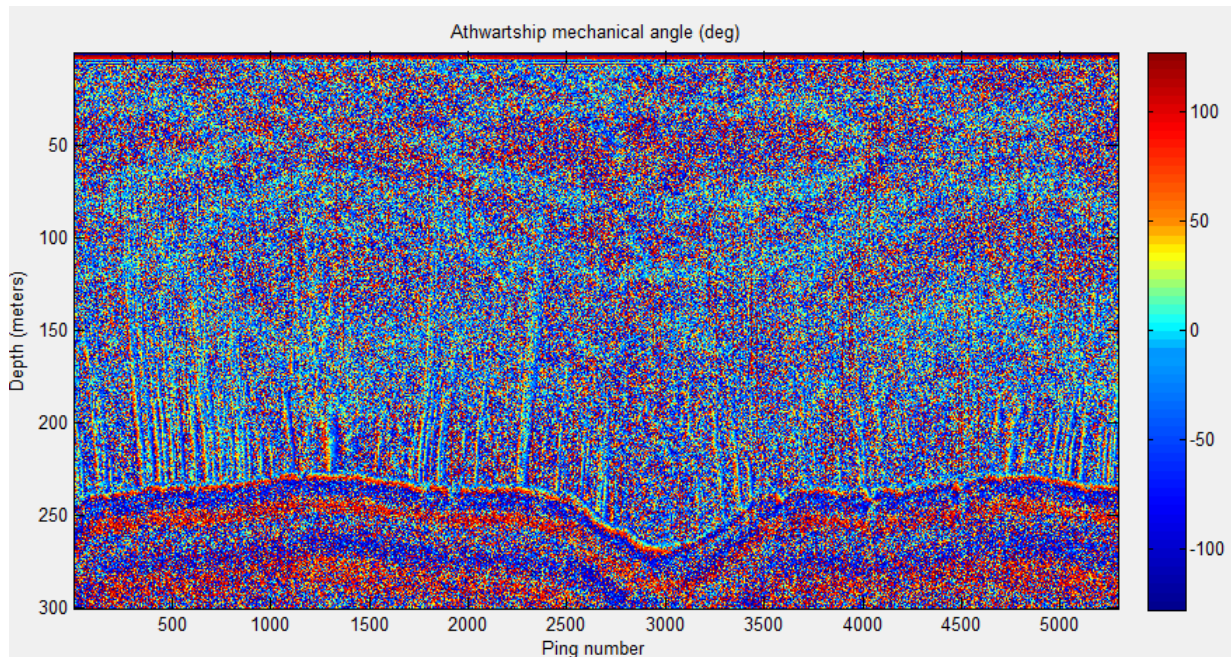


Figure 14. Image of mechanical angles - *Athwartship*

7.7 Configuration display

The configuration listbox displays transducer specific configurations and values (Figure 15b) after pressing once the word *config* inside the listbox (Figure 15a).

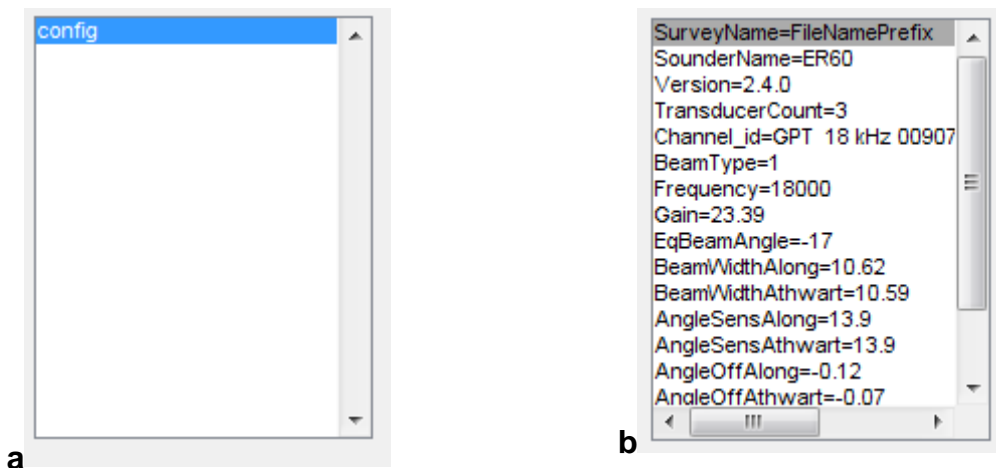


Figure 15. Screenshots of the configuration listbox (a) and a sample of contents.

7.8 Display position, date and time information

A 'draggable point' can be enabled in the main echogram view by pressing the *Enable Lat Lon information* button (Figure 16). The coordinates, date and time of the current position are displayed at the top of the echogram. The 'draggable point' can be disabled by pressing the *Disable Lat Lon information* button.

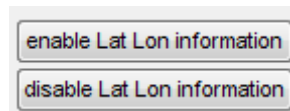


Figure 16. Screenshot of the Latitude and Longitude options push buttons.

7.9 GPS track

The survey track of the respective data file is displayed in a separate window when the *GPS track* button is pressed (Figure 17). The MATLAB code which executes this process is a sub-function of the *FlareHunter.m* script.

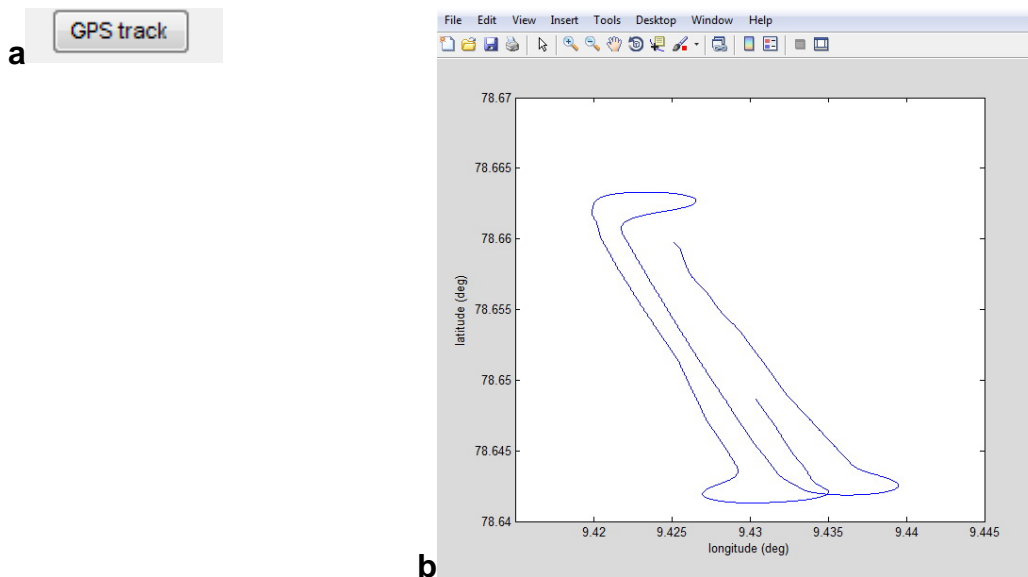


Figure 17. Screenshots of the graphical representation of the GPS coordinates generated by *FlareHunter*. The GPS track corresponds to the echogram shown in Fig. 2.

Latitude and Longitude information can be exported by pressing the *Export Lat Lon* button (Figure 18). The information is saved in a .txt file.

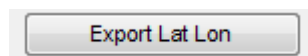


Figure 18. ExportDate & Time push button.

7.10 Motion and heading

In a similar way as for the survey track, motion and orientation of the vessel are displayed in separate images when pressing the *Pitch*, *Roll*, *Heave* or *Heading* buttons (Figure 19). Each graph shows its respective information as a function of the ping number (Figure 20, 21, 22 and 23).



Figure 19. Screenshot of the push buttons that are referred to the motion of the vessel.

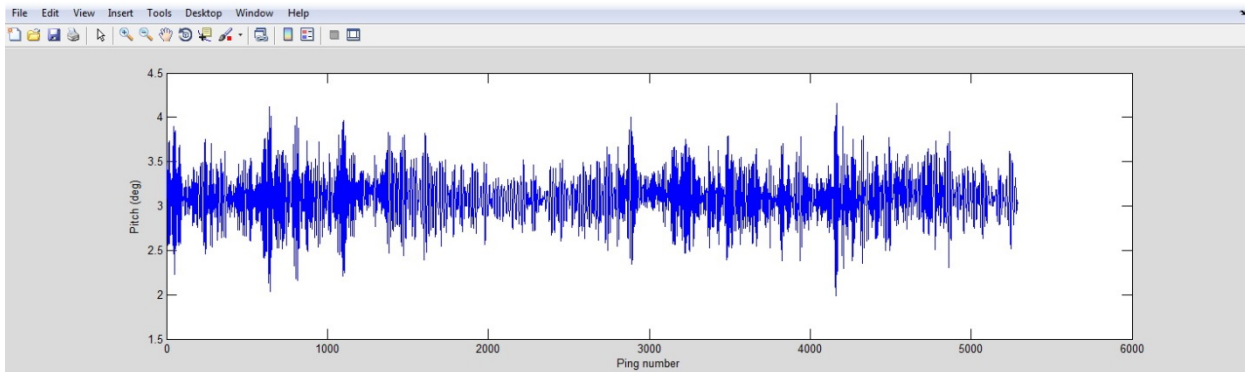


Figure 20. Screenshot of the graph showing the pitch information of the echogram shown in Fig.2. The pitch is represented in function of the ping number.

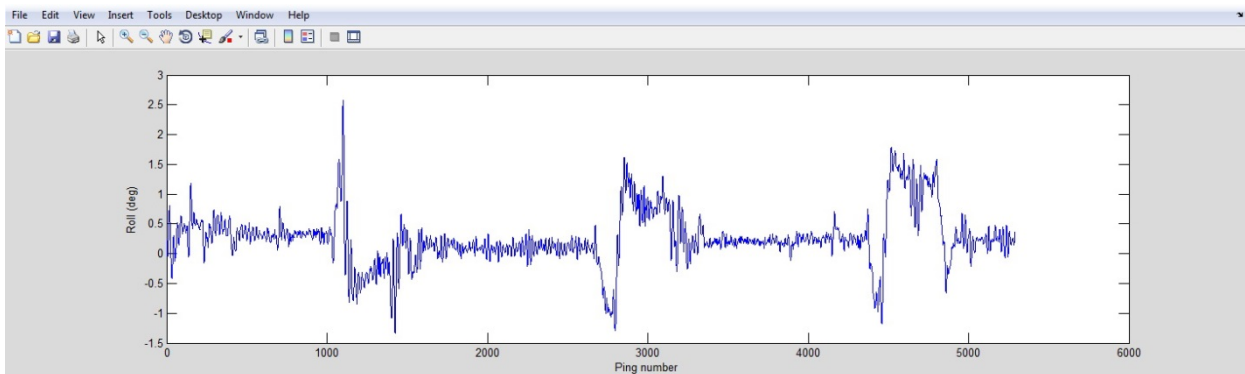


Figure 21. Screenshot of the graph showing the roll information of the echogram shown in Fig.2. Roll information is represented in function of the ping number.

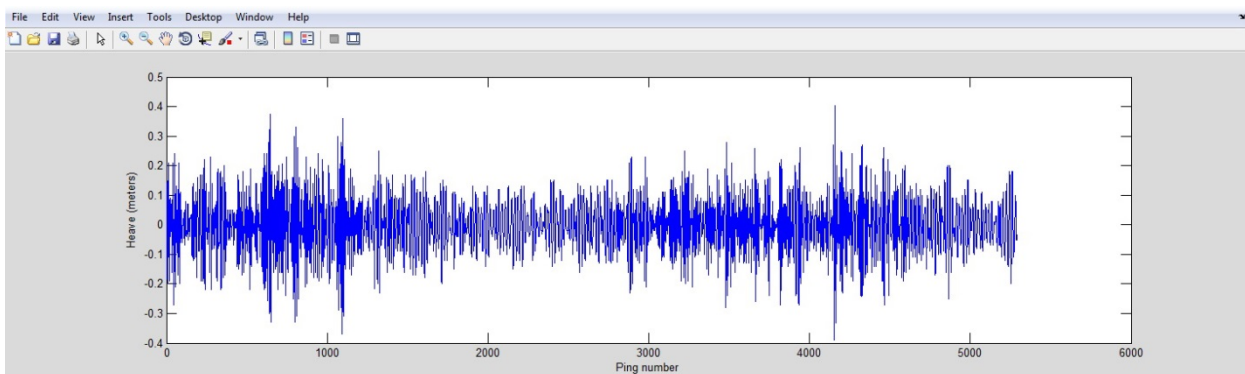


Figure 22. Image of graph showing the heave information of the echogram shown in Fig.2. Heave is

represented in function of the ping number.

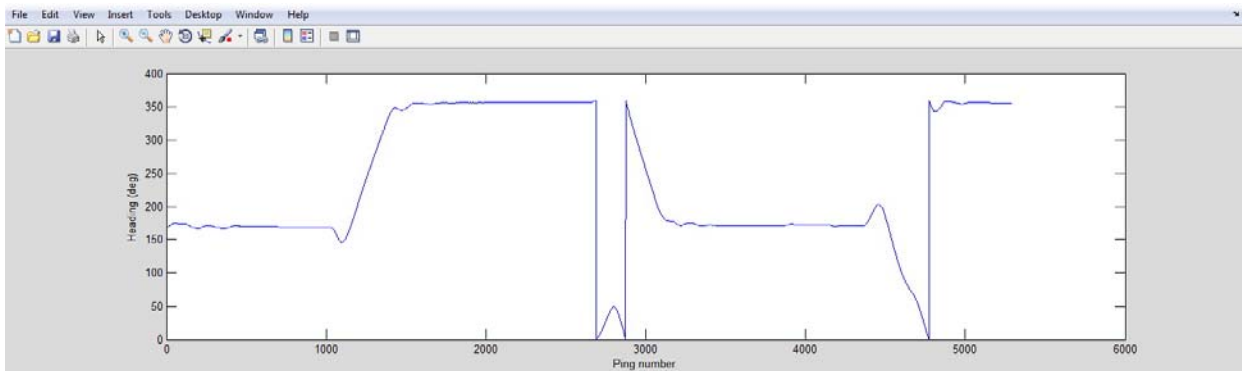


Figure 23. Screenshot of the graph showing the heading information of the vessel (echogram from Fig.2). The heading is shown as a function of the ping number.

7.11 Bottom detection

For bottom detection, the maximum amplitude of each trace (ping) of not close to the surface is detected as the seafloor. The detection is done when the *Bottom detection* button is pressed (Figure 24a). At the same time graph displays the bottom as a function of the ping number (Figure 24b).



Figure 24. Screenshot of the bottom detection figure generated by the *FlareHunter* GUI. The graph shows depth in function of the number of ping. The bottom detection corresponds to the echogram shown in Fig. 1

7.12 Zoom

A zoom button is available to zoom in and zoom out in the echograms (Fig. 25a). Once the button is pressed, a zoom can be done over a selected region in the echogram (left click to select region). To zoom out, right click and see the option (Fig. 25b). Alternatively, zoom in and zoom out are possible by using the scroll wheel of the mouse. To re-use the rest of the GUI options, deactivate the zoom functions by clicking the *Zoom* button again.



Figure 25. Screenshots of the zoom push button (a) and the table (b) with the zoom out option.

7.13 Spatial distribution of backscattering

The central part of *FlareHunter* is to extract spatially correct backscatter values. The GUI offers the possibility of editing backscatter values in a 3D space; data are extracted from the selected area in the main echogram view by considering maximum and minimum signal strength thresholds (Figure 26). A depth limit can be set to cut off data which are deeper than this depth limit. Water depth is given as negative value.

Figure 26 shows a window titled 'Spatial distribution of backscattering' with the following input fields:

Min value	-50	(dB)
Max value	0	(dB)
Depth limit of flare	-1000	meters

Figure 26. Screenshot of the “Spatial distribution of backscattering” input table.

The selection of the flare and its visualization can start once the *Enable Flare Selection* button has pressed (Figure 27). The selection can be done by clicking the left mouse button in the echogram view. The selection is stopped when the *Disable Flare Selection* button is pressed.

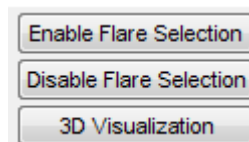


Figure 27. Screenshot of the push buttons which are referred to the tools that are required for the 3D visualization. By the “Enable Flare Selection” it is possible to select the desirable area and with “3D Visualization” to plot the included backscattering information.

The 3D visualization is initiated when the *3D Visualization* button is pressed. The selected and threshold filtered data are corrected using the information of the electrical angles recorded with the EK60 system (split-beam technique, e.g. see Medwin and Clay, 1998). The mechanical angles (alongship and athwarship angles) are obtained using the following relation:

$$\text{mechanical angle} = \left(\frac{\text{electrical angle}}{\text{angle sensitivity}} \right) \frac{180}{128} - \text{angle_offset} \quad [4]$$

The final positions of the backscattering location are computed by using the information of the vessel motion (pitch, roll and heave), heading and static offsets (see Figure 28). The final positions are calculated with the following set of equations:

$$\theta = \tan^{-1}(\text{sqrt}(\tan^2(\alpha + \text{roll}) + \tan^2(\beta + \text{pitch}))) \quad [5]$$

$$z = R \cos \theta \quad [6]$$

$$z_{\text{correct_heave}} = z + \text{heave} \quad [7]$$

$$x_{\text{alo}} = z_{\text{correct_heave}} * \tan(\beta + \text{pitch}) \quad [8]$$

$$y_{\text{ath}} = z_{\text{correct_heave}} * \tan(\alpha + \text{roll}) \quad [9]$$

$$E = E_0 + (x_{\text{alo}} + \Delta y) * \sin(\text{heading}) + (y_{\text{ath}} + \Delta x) * \cos(\text{heading}) \quad [10]$$

$$N = N_0 + (x_{\text{alo}} + \Delta y) * \cos(\text{heading}) - (y_{\text{ath}} + \Delta x) * \sin(\text{heading}) \quad [11]$$

$$z_{\text{scat}} = Z_{\text{correct_heave}} \quad [12]$$

Where:

α_M	: Athwarship mechanical angle
β_M	: Alongship mechanical angle
R	: Corrected range (m)
Z	: Depth scatterer (m)
$Z_{\text{correct_heave}}$: Depth scatterer including heave (m)
Y_{alo}	: Alongship position inside the beam (m)
X_{ath}	: Athwarship position inside the beam (m)
E_0, N_0	: GPS coordinates (UTM)
$\Delta x, \Delta y$: static offset transducer, athwarship and alongship respectively (m)
Z_{scat}	: Scatterer depth (m)

The corrections and inputs used in the above given equations [5] to [12] are graphically represented in Figure 28.

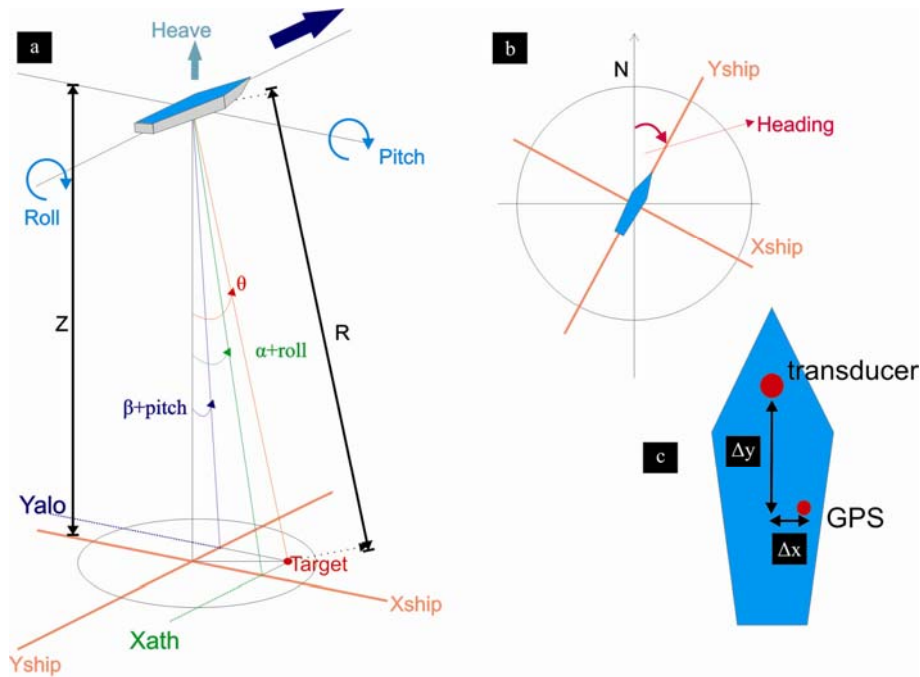


Figure 28. Schematic description of correction of the backscattering positions using mechanical angles, motion compensation and static offsets. a) descriptive image of the correction using pitch, roll, heave, and mechanical angles. b) Description of the axes rotation using the heading of the vessel. c) A simple description of the static offsets.

When the positioning correction is done, the spatial distribution of the selected backscattering is displayed in 3D (Fig. 29a). Once the figure is displayed *FlareHunter* enables to edit the backscatter signals in 3D and save the edited information in *.mat files.

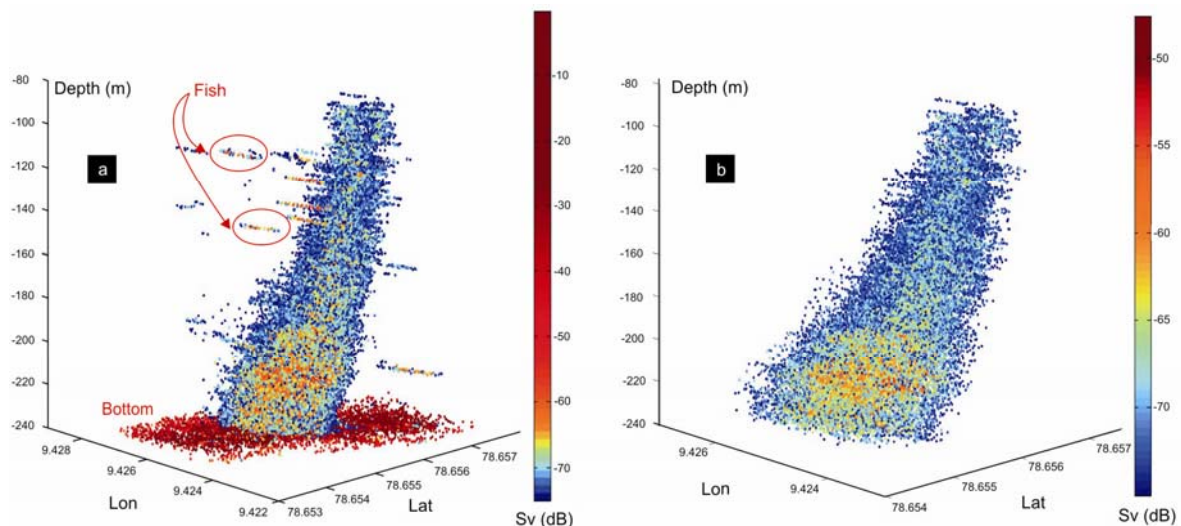


Figure 29. 3D visualization of the selected backscattering under a set threshold. The image shows the backscattering produced by bubble release at the seafloor (acoustic flare). The backscattering corresponds to the selected area in the echogram of the Fig.2. a) Spatial distribution of the backscattering produced by different targets (gas bubbles, fish and seafloor). b) 3D image of edited backscattering, showing information coming ideally from only gas release.

7.14 3D visualizationtools

Editing the extracted backscatter in 3D uses several tools available from MATLAB (Figure 30). Six tools are briefly mentioned.

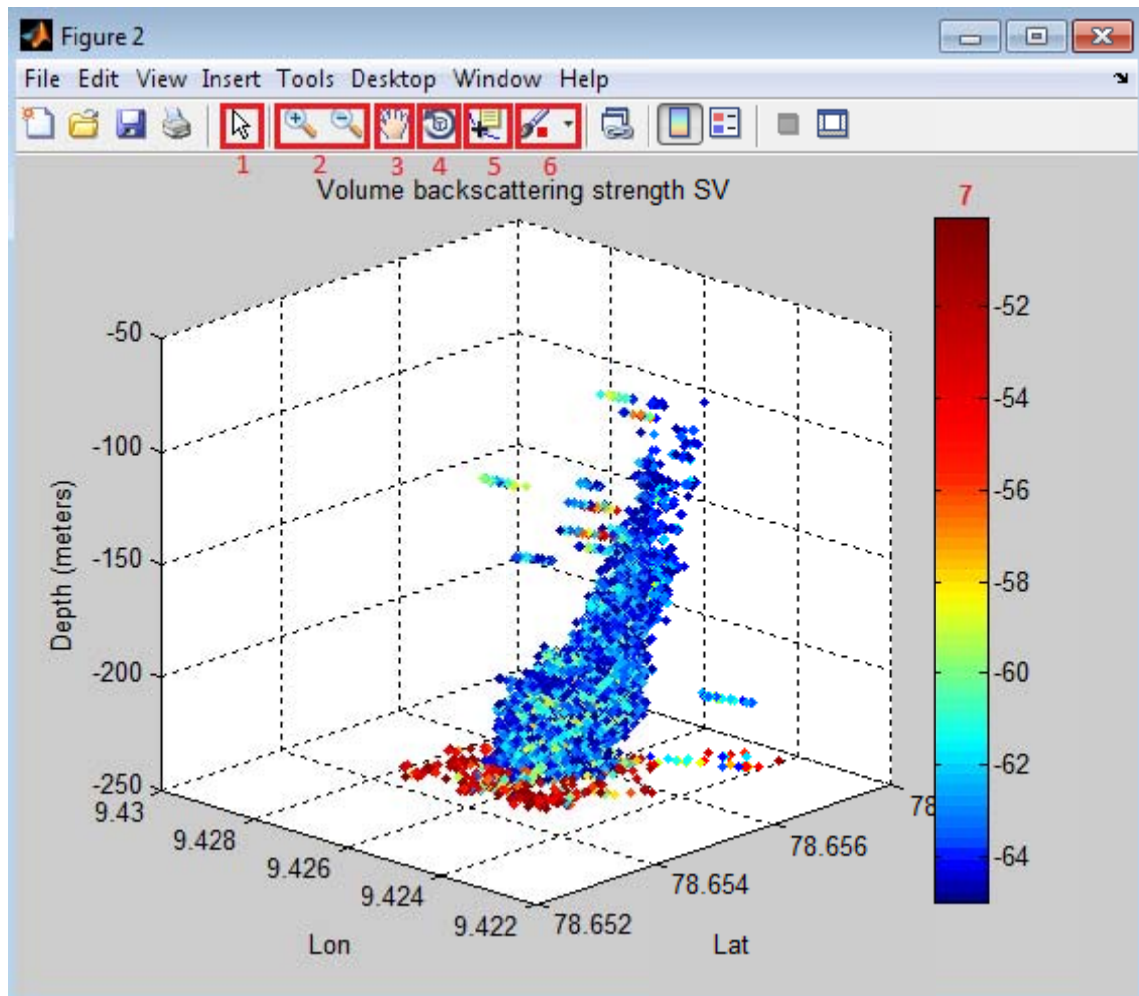


Figure 30. Screenshot of the 3D Visualization window with a sample of backscattering information selected.

1. *Edit plot*: Starting edit plot mode allows changing the color bar (7); clicking the right mouse button over the color bar and selecting 'Interactive Color Map Shift'.

2. *Zoom In and Out*: Enables to zoom in and out by positioning the mouse where the center of the plot needs to be.

3. *Pan*: Turns on mouse-based panning in the current figure.

4. *Rotate 3D*: Enables mouse-base rotation on all axes within the current figure.

5. *Data Cursor*: Displays the x, y, and z value of the nearest data point by clicking on one of the data values.

6. *Brush/Select Data*: Enables to delete data points by selecting them in groups or point by point.

Once the data have been edited, e.g. the seafloor has been removed and all unwanted noise has been deleted, the edited data can be saved by pressing the *Save Edited Selection* button (Figure 31)



Figure 31. Screenshot of the push button “Save edited selection”.

Data of the flare are saved in the *edited flares* folder. This data is appended as a variable in the .mat file with the name of the corresponding raw file analyzed (i.e. *name_raw_file_flares.mat*). The variable is a cell array which contains data as detailed below.

Table 3. Content of the cell array of edited data of the flare.

cell index number	1	2	3	4	5	6	7	8	9	10	11	12
Data	UTM X	UTM Y	UTM Zone	Lat	Lon	Depth	TS	Sv	Ping number	UTM X at seafloor	UTM Y at seafloor	Footprint radius at seafloor

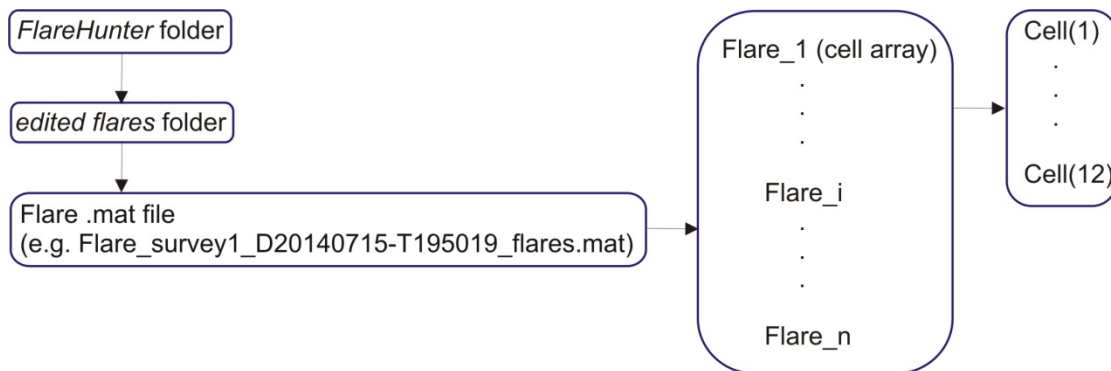


Figure 32. Location and organization of stored information of edited flares

7.15 Flare spine

A geometrical average can be applied over the position of the scatterers of a flare which represents a single bubble stream. The geometrical average, also named ‘flare spine’, can be generated by pressing the *flare spine* button (Figure 33). The flare spine is generated only if a flare image is open (Figure 34a). Then, an image like the Figure 34b is generated. The number of vertical divisions of the geometrical average can be pre-set before the flare spine is generated (by default $n=30$, Figure 31).

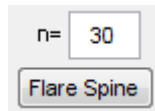


Figure 33. Screenshot of the “flare spine” push button.

The generated image (Figure 34b) shows the geometrical average of the positions (x, y, z coordinates) of the scatterers inside each depth cell (bin).

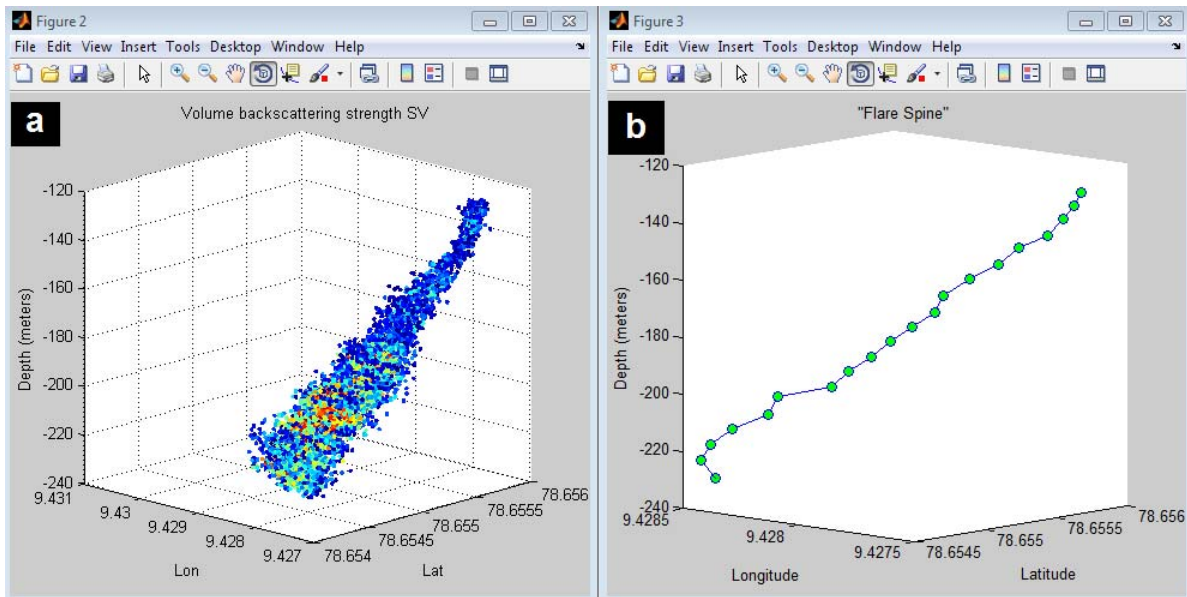


Figure 34. Screenshots representing the backscattering information (a) of a selected flare and the geometrical average (b) of the points.

Once the flare spine is generated, the data can be saved by pressing the *SaveSpine* button (Figure35)

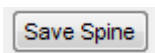


Figure 35. *Save Spine* button

Data of the spine are saved in the *spines* folder. This data is appended as a variable into the .mat file with the name of the corresponding raw file analyzed (i.e. *name_raw_file_spine.mat*). The variable is a cell array which contains the data as detailed bellow.

Table 4. Content of cell array of edited data of the flare spine

cell index number	1	2	3	4	5	6	7	8
Data	UTM X	UTM Y	UTM Zone	Lat	Lon	Depth	TS	Sv

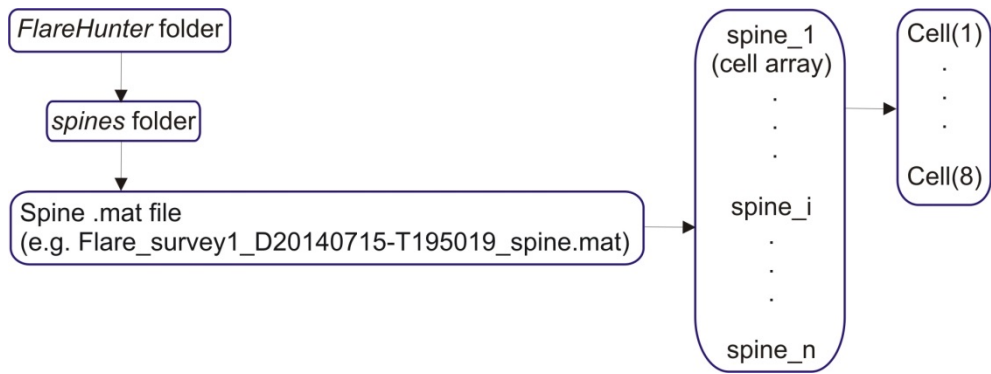


Figure 36. Location and organization of stored information of flare spines

8. FlareFlowModule

The *FlareFlowModule* is a specialized tool to estimate the gas flow rate of single flares using the hydroacoustics inverse method published by Muyakshin and Sauter (2010) which has been further developed by Veloso et al. (in review). The result is a flow rate in mL/min; using the appropriate gas density at the depth of analyses allows to give the flow rate in Mol/min. The method for deriving gas flow rate from hydroacoustic data is based on the received volume backscattering strength S_V from bubbles when they are insonified with a certain frequency. The advanced method by Veloso et al. (in review) is used as it applies the scattering cross-section model of Thuraisingham (1997). This model is valid for all ka values, where k is the wave number and a represents the bubble radius.

FlareFlowModule uses as inputs the hydroacoustic data of free gas release edited using *FlareHunter*, the bubble size distribution (BSD) e.g. based on visual observations or assumptions, the bubble rising speed BRS from observations or models that take into account the radius range of the BSD. Furthermore, *FlareFlowModule* requires values of physical properties of the seawater and the gas, values of the environmental properties as well as echosounder parameters. A flowchart describing the process of the flow rate estimation is detailed in Figure 37.

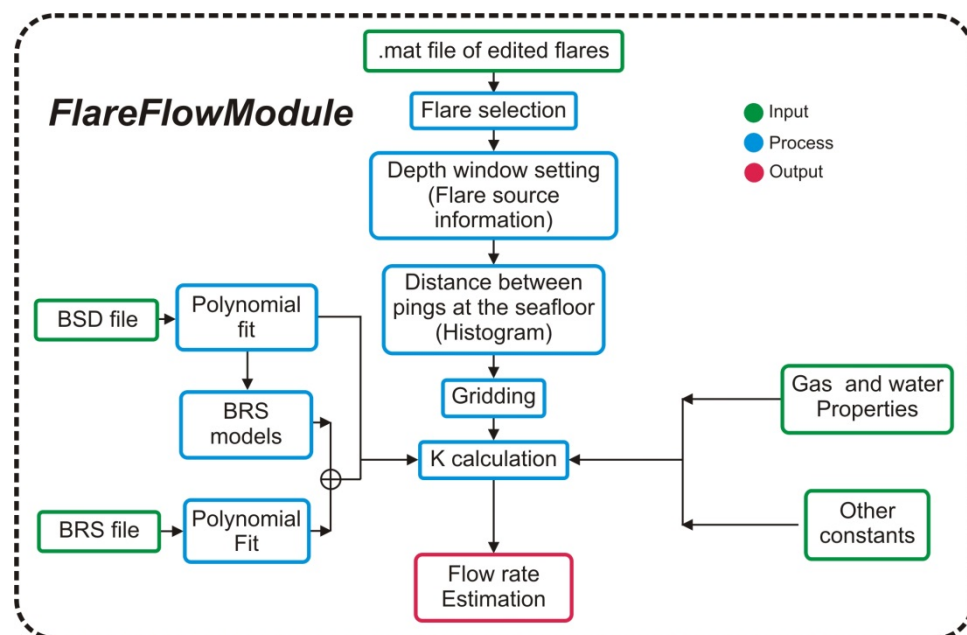


Figure 37. *FlareFlowModule* flowchart

The *FlareFlowModule* can be launched from *FlareHunter* GUI by pressing the *Launch FlareFlowModule* button (Figure 38a) or it can be run directly from the command window of MATLAB (Figure 38b) when the current folder is the *FlareHunter* folder.



Figure 38. Two ways to launch *FlareFlowModule*

Once the *FlareFlowModule* is launched, the GUI is displayed (Figure 39)

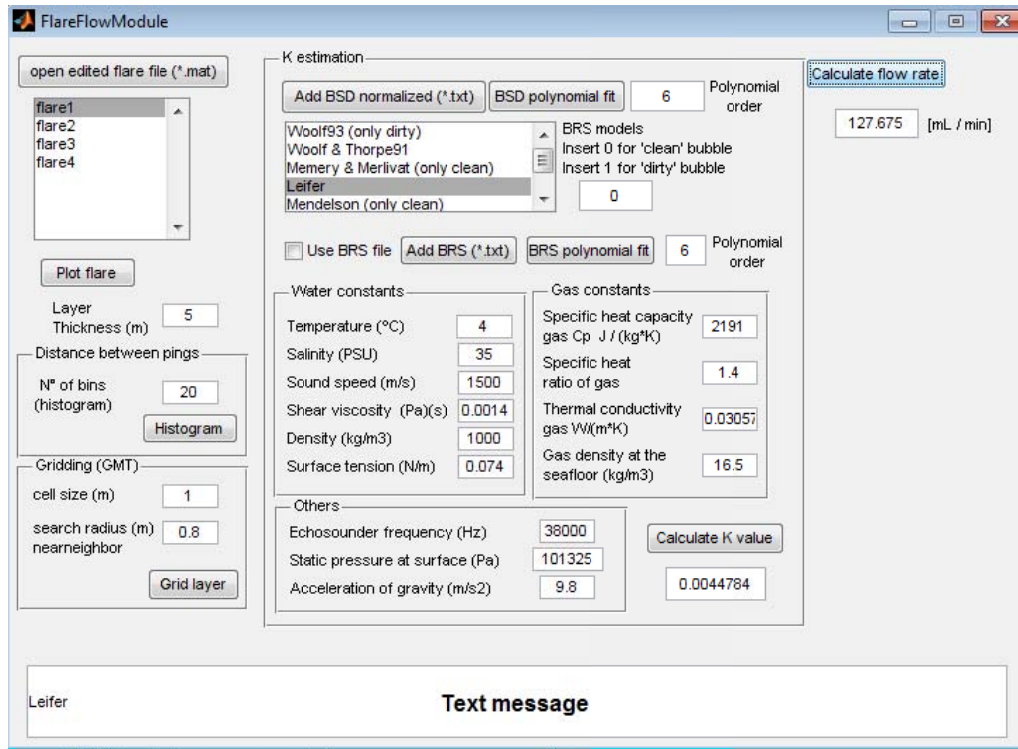


Figure 39. *FlareFlowModule* GUI

8.1 Loading edited flares

Flare files edited in *FlareHunter* can be loaded by pressing the *open edited flare file* button (Figure 40).

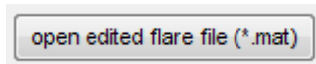


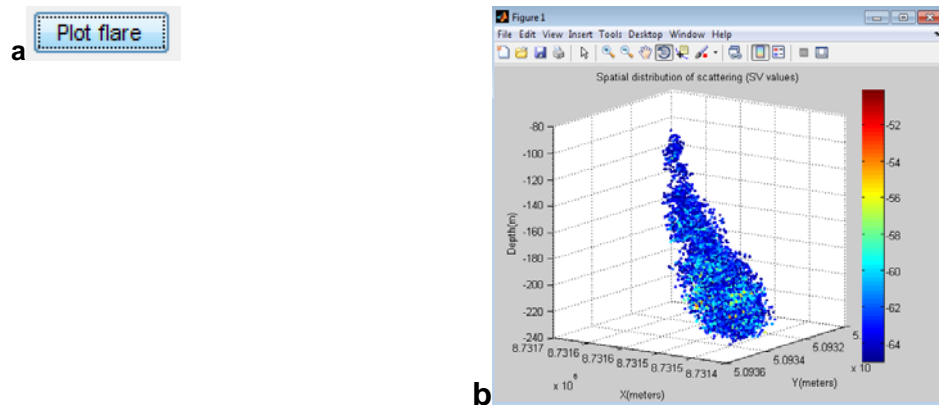
Figure 40. *Open edited flare file* push button

Once the file is loaded, a list of the edited flares that the file contains appears after clicking *Flare list* in the listbox.



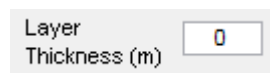
Figure 41.List of flares of loaded .mat file

A flare from the listbox must be selected to carry out the flow rate evaluation. The selected flare can be visualized by pressing the *Plot Flare* button (Figure 42a). The three dimensional view of the scattered data is displayed (Figure 42b).

**Figure 42.**a) *Plot Flare* push button. b) Three dimensional representation of flare scattering.

8.2 Layer thickness selection of the flare source

Flow rate estimation is evaluated using a **hydroacoustic layer which always starts at the deepest scatterer of the edited flare(!)** and ends at a depth defined by the layer thickness in the *FlareFlowModule* GUI. The layer thickness (in meters) can be defined at the *Layer Thickness* text box (Figure 43).

**Figure 43.***Layer Thickness* box

8.3 Data gridding and interpolation

The flow rate is calculated by using the spatially distributed discrete values of the volume backscattering coefficient (σ_V). To evaluate the flow rate at the source of the flare, σ_V values from the selected layer (derived from S_V values) are gridded and interpolated using the *nearneighbor* command of GMT (Wessel et al. 2013).

Cell size and search radius can be chosen by the user (Figure 46 a); the analysis of the distances between consecutive pings as discussed by Veloso et al. (in review) should be considered for the cell size. A histogram of the distance between consecutive pings can be launched to choose an appropriate cell size value (e.g. mode; Figure 44b). The histogram is generated by pressing the *Histogram* push button (Figure 44a). A text box is available to modify the number of bins of the displayed histogram (Figure 44a).

Once the cell size and the search radius are chosen, the gridding and interpolation of the σ_v values can be generated by pressing the *Grid layer* button (Figure 45a). An image of the gridded values is displayed after the nearneighbor gridding is executed (Figure 45b).

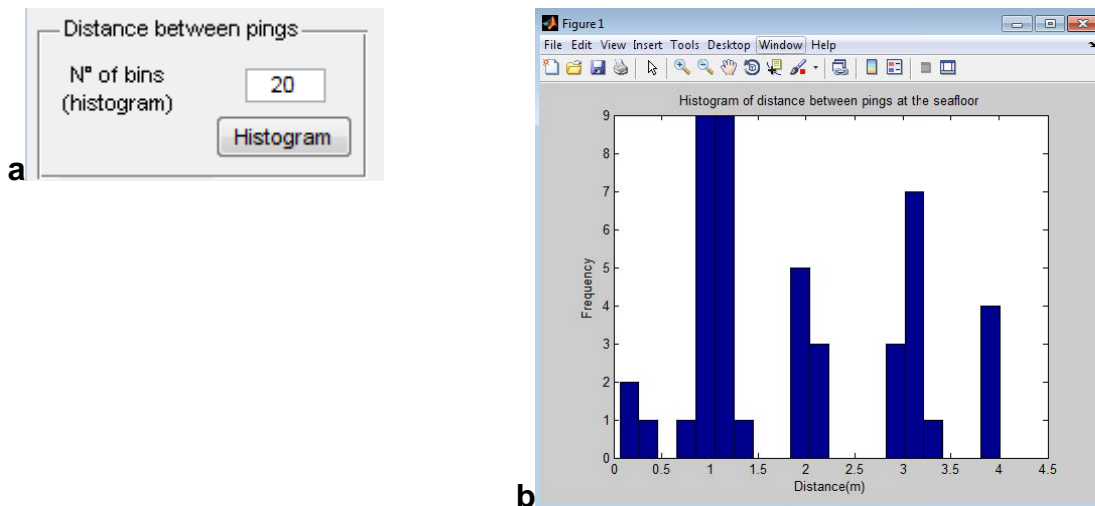


Figure 44. a) Histogram push button and text box to modify the number of bins of the histogram. b) Image of the histogram representing the distance of consecutive pings at the seafloor.

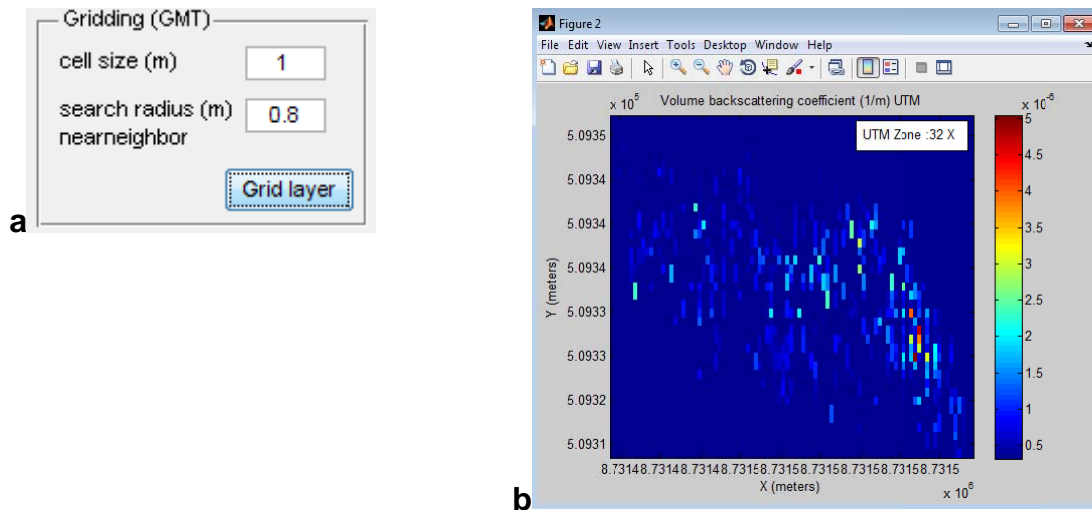


Figure 45. a) *Grid layer* push button and text boxes to set the cell size of the gridding and the search radius of the interpolation. b) Image of the gridded layer.

8.4 Estimation of K value

The K value of the inverse method defined by Muyakshin and Sauter (2010) has to be calculated to evaluate the flow rate of the single flare (see Veloso et al, in review). One necessary input to calculate K is the bubble size distribution (BSD). A BSD file ASCII file containing a column with the bubble sizes (in meters) and a

column specifying the frequency of occurrence of the respective bubble sizes (normalized to the maximum frequency, maximum value 1) can be used as input. An example of a BSDfile is shown in Figure 46.

radius	normalized frequency
0.001	0.054054
0.0015	0.21622
0.002	0.66216
0.0025	0.66216
0.00275	0.75676
0.003	1
0.00325	0.90541
0.00375	0.41892
0.004	0.48649

← max. value

Figure 46. BSD *.txt file example

The BSD file can be loaded with the *Add BSD normalized* button (Figure 47). Once the file is loaded, a polynomial fit needs to be carried out to obtain the BSD function. This can be done by pressing the *BSD polynomial fit* button (Figure 48a). A text box allows to set the order of the polynomial. After executing the polynomial fit, the original BSD and the polynomial function are displayed in the same graph for visual comparison (Figure 48b).

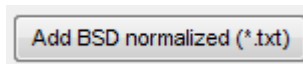


Figure 47. Push button to load BSD .txt file

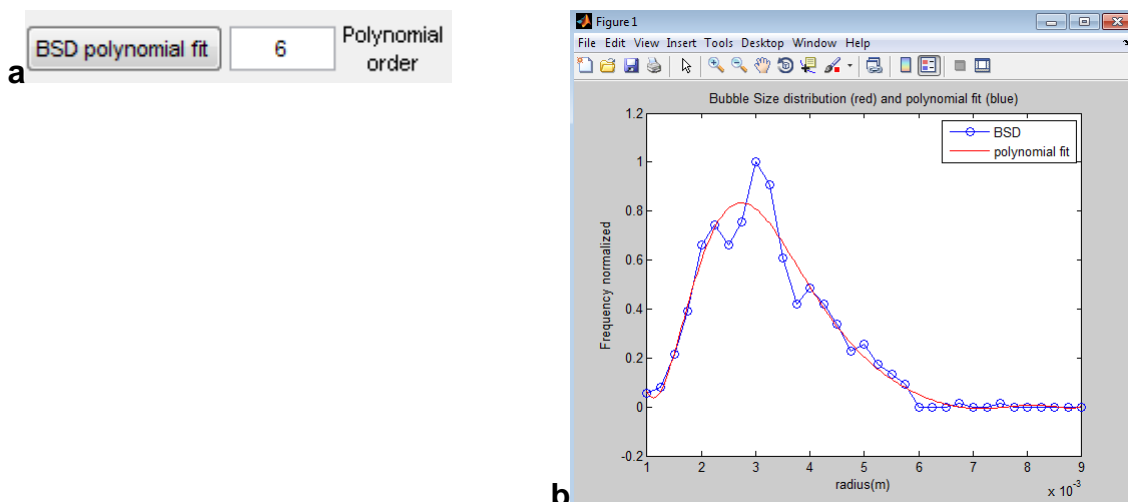


Figure 48.a) Push button to generate the BSD function and text box to set the polynomial order. b)

Image of the original BSD and the polynomial fit

The second important input to calculate the K value is the rising speed of the bubbles, accounting for the different bubble sizes. This can either be calculated based on several literature models for 'clean' and 'dirty' bubbles (Mendelson 1967; Woolf and Thorpe 1991; Woolf 1993; Leifer et al. 2000; Leifer and Patro 2002) or loaded using a .txt file which defines the bubble rising speeds for different bubble sizes that might have been measured directly in the field.

The different available models are displayed on a list box (Figure 49) and can be selected by pressing on the respective name of the model. The models can either be run for 'clean' (set to 0) or 'dirty' (set to 1) bubbles.

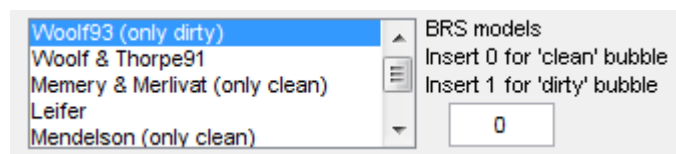


Figure 49. Bubble rising speed models and text box to set if the bubble is 'clean' or 'dirty'

A BRS .txt file can be loaded as an optional alternative to obtain the BRS function by pressing the *Add BRS* push button (Figure 50).

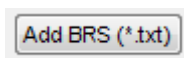


Figure 50. Push button to load the BRS .txt file

The BRS .txt file contains two columns, being the first one the radius of bubbles (meters) and the second one the bubble rising speed (m/s). An example of the BRS file is shown in Figure 51.

radius (meters)	BRS (m/s)
0.0010	0.23
0.0015	0.20
0.0020	0.19
0.0030	0.18
0.0040	0.19
0.0050	0.19
0.0060	0.20
0.0070	0.21
0.0080	0.22

Figure 51. Example of BRS .txt file

Once the BRS .txt file is loaded, the BRS function can be created by pressing the *BRS polynomial fit* button. A text box is available to define the order of the polynomial fit. The *BRS polynomial fit* button generates a figure that shows the loaded BRS information and their polynomial fit for visual comparison.

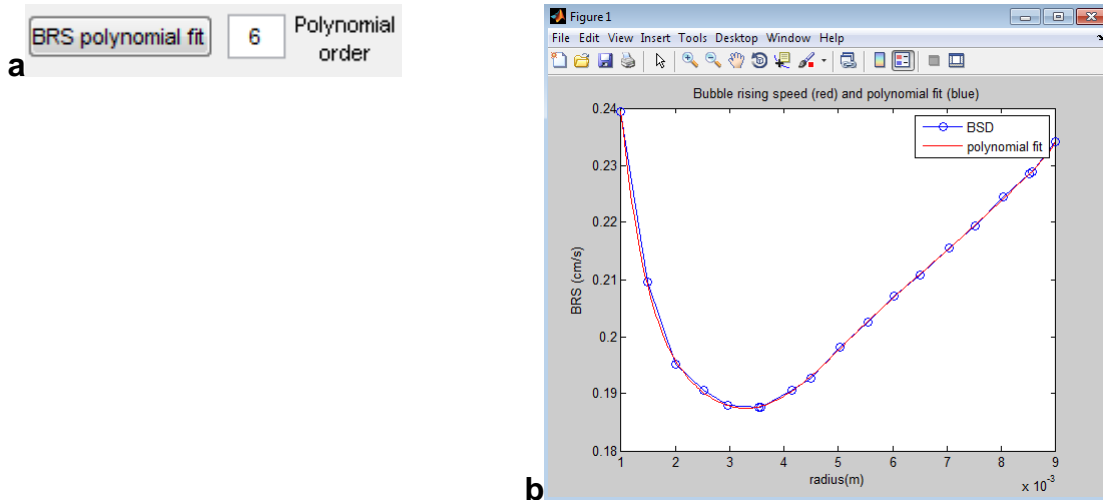


Figure 52. a) Push button to generate the BRS function and text box to set the polynomial order. b) Image of the original BRS and the polynomial fit

For Calculating the K value using a BRS .txt file the 'Use BRS file' check box needs to be checked (!!) (Figure 53)

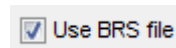


Figure 53. check box to choose BRS models or BRS .txt file to calculate K

To finalize the calculation of the K value, environmental and transducer specific input values need to be entered in the respective text fields

With all the constants entered, the K value can be calculated by pressing the *Calculate K value* button. The result is shown in a text box located below the push button (Figure 54).

Water constants		Gas constants	
Temperature (°C)	<input type="text" value="4"/>	Specific heat capacity gas Cp J / (kg*K)	<input type="text" value="2191"/>
Salinity (PSU)	<input type="text" value="35"/>	Specific heat ratio of gas	<input type="text" value="1.4"/>
Sound speed (m/s)	<input type="text" value="1500"/>	Thermal conductivity gas W/(m*K)	<input type="text" value="0.03057"/>
Shear viscosity (Pa)(s)	<input type="text" value="0.0014"/>	Gas density at the seafloor (kg/m3)	<input type="text" value="16.5"/>
Density (kg/m3)	<input type="text" value="1000"/>		
Surface tension (N/m)	<input type="text" value="0.074"/>		
Others			
Echosounder frequency (Hz)	<input type="text" value="38000"/>	<input type="button" value="Calculate K value"/>	
Static pressure at surface (Pa)	<input type="text" value="101325"/>		
Acceleration of gravity (m/s ²)	<input type="text" value="9.8"/>		<input type="text" value="0.0044784"/>

Figure 54. Text boxes of constants values, push button to calculate K and text box showing the result of the K value

8.5 Flow rate estimation

Finally the calculation of the flow rate of the defined depth layer in the selected flare is done by pressing the *Calculate flow rate* button. A text box located below the push button shows the flow rate in mL/min.

<input type="button" value="Calculate flow rate"/>
<input type="text" value="127.675"/> [mL / min]

Figure 55. Push button to calculate the flow rate of the single flare and text box showing the result.

A text box for messages can be found at the bottom of the GUI (Figure 39). This text box shows warnings, errors and selected options in the GUI during processing (e.g. Figure 56)

...please select BRS file!!!

Figure 56. Text message example

9. References

- Leifer, I., R. K. Patro, and P. Bowyer. 2000. A study on the temperature variation of rise velocity for large clean bubbles. *J. Atmos. Ocean. Tech.* 17:1392-1402, [doi:10.1175/1520-0426(2000)0172.0.CO;2]
- Leifer, I., and R. K. Patro. 2002. The bubble mechanism for methane transport from the shallow sea bed to the surface: A review and sensitivity study. *Cont. Shelf. Res.* 22:2409-2428, [doi:10.1016/S0278-4343(02)00065-1]
- Medwin, H., and C. S. Clay. 1998. Chapter 10—Sonar Systems: Measurements and Inversions, p. 405-467. *Fundamentals of Acoustical Oceanography*. Academic Press, [doi:10.1046/j.1365-2419.1999.00096.x]
- Mendelson, H. D. 1967. The prediction of bubble terminal velocities from wave theory. *AIChE J.* 13:250-253, [doi:10.1002/aic.690130213]
- Muyakshin, S., and E. Sauter. 2010. The hydroacoustic method for the quantification of the gas flux from a submersed bubble plume. *Oceanology* 50:995-1001, [doi:10.1134/S0001437010060202]
- Simrad. 2012. *Simrad EK60 Reference Manual Release 2.4.X, Rev.D*, Kongsberg Maritime AS, Strandpromenaden 50, 3190 Horten, Norway.
- Thuraisingham, R. A. 1997. New expressions of acoustic cross-sections of a single bubble in the monopole bubble theory. *Ultrasonics*, 35(5): 407-409, [doi:10.1016/S0041-624X(97)00021-8]
- Veloso, M., J. Greinert, J. Mienert, M. De Batist. 201x. A new methodology for bubble flux quantifications using split-beam echosounders in deep water: Examples from the Arctic offshore NW- Svalbard.

Wessel, P., W. H. F. Smith, R. Scharroo, J. Luis, and F. Wobbe. 2013. Generic MappingTools: Improved Version Released. *Eos, Trans. Amer. Geophys. Union*94:409-410, [doi:10.1002/2013EO450001]

Woolf, D. K. 1993. Bubbles and the air-sea transfer velocity of gases. *Atmos. Ocean*31:517-540, [doi:10.1080/07055900.1993.9649484]

Woolf, D. K., and S. Thorpe. 1991. Bubbles and the air-sea exchange of gases in near-saturation conditions. *J. Mar. Res.*49:435-466, [doi:10.1357/002224091784995765]