NAME

mbareaclean – Tool to automatically flag bad beams in swath sonar bathymetry data within a specified area.

VERSION

Version 5.0

SYNOPSIS

mbareaclean **–R***west/east/south/north* **–S***binsize* [**–D***threshold* **–F***format* **–I***infile* **–B** **–G** **–H** **–M***threshold*[*/nmin*[*/nmax*]] **–N**[-]*min_beam*[*/maxbeam*] **–T***type* –V*]*

DESCRIPTION

mbareaclean identifies and flags artifacts in swath sonar bathymetry data within a specified area of interest. The input data are one swath file or a datalist referencing multiple swath files. The user may specify the work area in longitude and latitude bounds, along with a bin size in meters. If these are not specified, the program will attempt to set useful values. The area is divided into a grid with square cells of the specified bin size. As the data are read, each of the soundings that fall within one of the bins is stored. Once all of data are read, one or more statistical tests are performed on the soundings within each bin, providing there are a sufficient number of soundings.

The user may specify one or both of the following actions:
1) Previously unflagged soundings that fail a test are flagged as bad
2) Previously flagged soundings that pass all tests are unflagged (the **–G** option)..

The edit events are output to edit save files which can be applied to the data by the program **mbprocess**. These are the same edit save files created and/or modified by **mbclean** and **mbedit**. If a sounding's flag status is changed, that flagging action is output to the edit save file of the swath file containing that sounding. This program will create edit save files if necessary, or append to those that already exist.

At present only two algorithms for identifying good and bad beams is implemented. The first is a simple median filter controlled by the **–M***threshold/nmin[/nmax]* option. Soundings that differ from the median depth by a value greater than *threshold* times the sonar altitude will be considered "bad". So, if *threshold* = 0.05, then any sounding that is 5% greater or less than the median depth will be considered bad. The *nmin* parameter sets the minimum number of soundings required to use the median filter. The *nmax* parameter sets the maximum number of soundings allowed within a cell. If the number of initially good beams in a cell exceeds *nmax*, then only the *nmax* soundings nearest the median value remain unflagged, and the rest are flagged even if they meet the threshold test. The default values are *threshold* = 0.25, *nmin* = 10, and *nmax* = infinite.

The second algorithm is a simple standard deviation filter controlled by the **–D***threshold/nmin* option. Soundings that differ from the mean depth by a value greater than *threshold* times the standard deviation of the bin will be considered "bad". So, if *threshold* = 2, then any sounding that is more than 2 standard deviations greater or less than the mean depth will be considered bad. The *nmin* parameter sets the minimum number of soundings required to use the filter, The default values are *threshold* = 2.0 and *nmin* = 10.

Two options allow limiting the soundings that may be flagged. **–N**[-]*min_beam/maxbeam* limits flagging to only those beams in (or out) of the given range. **–T***type* limits flagging to soundings using the specified bottom detection algorithm. Many sounders use amplitude detection algorithms for the central beams and phase detection algorithms for the outer beams, falling back to amplitude detection when phase detection fails. Amplitude detections in the outer beams are second rate soundings and frequently have a higher range of errors than other soundings. Using **–N** and **–T** together allows stricter cleaning algorithms to be applied to these lower quality data. For example to flag bad all amplitude detections in the outer thirty beams which are more than one standard deviation from the mean: **–N-30 –T1 –D1**.

AUTHORSHIP

David W. Caress (caress@mbari.org)
Monterey Bay Aquarium Research Institute
Dale N. Chayes (dale@ldeo.columbia.edu)
Lamont-Doherty Earth Observatory

OPTIONS

–B

This option causes **mbareaclean** to flag as bad any previously unflagged soundings that fail one of the specified statistical tests. If neither **–B** or **–G** are specified, then the program will by default use this option.

–D *threshold*[*/nmin*]

Turns on use of a standard deviation filter test for the soundings. Soundings that differ from the mean depth by a value greater than *threshold* times the standard deviation will be considered "bad". So, if *threshold* = 2.0, then any sounding that is twice the standard deviation from the mean depth will be considered bad. The *nmin* parameter sets the minimum number of soundings required to use the standard deviation filter. The default values are *threshold* = 2.0 and *nmin* = 10.

–F *format*

Sets the data format used to read *infile*. If *format* < 0, then the input file specified with the **–I** option will actually contain a list of input swath sonar data files. This program uses the **MBIO** library and will read or write any swath sonar format supported by **MBIO**. A list of the swath sonar data formats currently supported by **MBIO** and their identifier values is given in the **MBIO** manual page. Default: *format* = 11.

–G This option cause **mbarealcean** to unflag (set good) any previously flagged soundings that pass all of the specified statistical tests. If neither **–B** or **–G** are specified, then the program will by default apply the **–B** option.

–H This "help" flag cause the program to print out a description of its operation and then exit immediately.

–I *infile*

Sets the input filename. If *format* > 0 (set with the **–F** option) then the swath sonar data contained in *infile* is read and processed. If *format* < 0, then *infile* is assumed to be an ascii file containing a list of the input swath sonar data files to be processed and their formats. The program will read and process the data in all of these files. Each input file will have an associated output file with either the ".esf" suffix. In the *infile* file, each data file should be followed by a data format identifier, e.g.:

datafile1 11
datafile2 24

This program uses the **MBIO** library and will read or write any swath sonar format supported by **MBIO**. A list of the swath sonar data formats currently supported by **MBIO** and their identifier values is given in the **MBIO** manual page. Default: *infile* = "datalist.mb-1".

–M *threshold*[*/nmin*]

Turns on use of a median filter test for the soundings. If no other test is specified, the median filter test is used by default. Soundings that differ from the median depth by a value greater than *threshold* times the sonar altitude will be considered "bad". So, if *threshold* = 0.05, then any sounding that is 5% greater or less than the median depth will be considered bad. The *nmin* parameter sets the minimum number of soundings required to use the median filter, The default values are *threshold* = 0.25 and *nmin* = 10.

–N [-]*min_beam*[*/max_beam*]

Limits the beams to which flagging will be applied. Only soundings for beams from *min_beam* to *max_beam*, inclusive, will have flags applied. If *min_beam* is preceded by a **–** then only beams outside the range will be flagged. If *max_beam* is not given it will be set so that an equal number of beams lay above it as lay below *min_beam*. So for a Simrad EM1002 sounder with 111 beams **–N-30** will only flag beam numbers 1 to 29 and 82 to 111. All good soundings, regardless of beam, will be included in the calculations of median, mean and standard deviation for the bin.

–R *west/east/south/north*
Sets the longitude and latitude bounds within which swath sonar data will be read, binned, and tested. Soundings lying outside these bounds will be ignored.
Default: The program will determine and use a square area encompassing all of the data contained in the input files.

–S *binsize*
Sets the size of the bins to be used in meters. The area specified with the **–R** option will be broken into a grid with cells that are roughly *binsize* meters east-west and north-south.
Default: A binsize equal to 0.2 times the maximum sonar altitude will be used.

–T *detect_type*
Limits the soundings to which flagging will be applied to only those which use the specified bottom detection algorithm. *detect_type* must be one of:
0 – unknown algorithm
1 – amplitude detection
2 – phase detection

–V Normally, **mbareaclean** works "silently" without outputting anything to the stderr stream. If the **–V** flag is given, then **mbareaclean** works in a "verbose" mode and outputs the program version being used, all error status messages, and the number of beams flagged as bad.

EXAMPLES

Suppose we are working with a set of 5 Reson 8101 multibeam data files comprising a shallow water survey. One of these files has previously been edited with **mbedit**, so an esf file exists and contains a number of pre-existing edits. If we know that this survey is contained in the area specified by –R-122.42556/-122.41974/47.67111/47.67529, then we can invoke **mbareaclean** using:

```
mbareaclean –Idatalist.mb-1 \
        -B –M0.1/10 –S2.5 \
        -R-122.42556/-122.41974/47.67111/47.67529  \
        -V
```

where the bin size is 2.5 meters, the median filter threshold is 0.1 (or 10%) of the sonar altitude, and the minimum number of sounding required for filtering is 10. The results look like:

```
Program MBAREACLEAN
Version $Id: mbareaclean.ps 2191 2014-06-05 00:59:01Z caress $
MB-system Version 5.0.beta29
Area of interest:
   Minimum Longitude: –122.425560 Maximum Longitude: –122.419740
   Minimum Latitude:  47.671110 Maximum Latitude:  47.675290
   Bin Size:   2.500000
   Dimensions: 175 186
Cleaning algorithms:
   Median filter: ON
   Plane fit:     OFF
Output:
   Flag unflagged soundings identified as bad:  ON
   Unflag flagged soundings identified as good: OFF

Processing 001_1730.fbt
Sorting 41580 old edits...
10000 of 41580 old edits sorted...
```

```
20000 of 41580 old edits sorted...
30000 of 41580 old edits sorted...
40000 of 41580 old edits sorted...
pings:1169  beams:   20233 good   41686 flagged   56150 null

Processing 003_1733.fbt
pings: 991  beams:   52439 good     159 flagged   47493 null

Processing 005_1736.fbt
pings:1011  beams:   53080 good     588 flagged   48443 null

Processing 007_1739.fbt
pings: 922  beams:   48854 good     212 flagged   44056 null

Processing 009_1741.fbt
pings:1017  beams:   53416 good     586 flagged   48715 null

MBareaclean Processing Totals:
-------------------------
5 total swath data files processed
5110 total pings processed
271253 total soundings processed
-------------------------
  0 soundings:  61919 flagged:      0 unflagged:      0  file:001_1730
  1 soundings:  52598 flagged:      0 unflagged:      0  file:003_1733
  2 soundings:  53668 flagged:      0 unflagged:      0  file:005_1736
  3 soundings:  49066 flagged:      0 unflagged:      0  file:007_1739
  4 soundings:  54002 flagged:      0 unflagged:      0  file:009_1741
```

SEE ALSO

mbsystem(1), **mbclean**(1), **mbedit**(1), **mbinfo**(1) **mbprocess**(1),

BUGS

The algorithms implemented in **mbareaclean** simply don't detect all bathymetric artifacts that are obvious to the eye on contour charts. Although the autofiltering tools **mbareaclean** and **mbclean** often do a credible first pass at flagging obvious artifacts, we strongly recommend that any swath bathymetry processing stream include interactive editing of the bathymetry data (e.g. **mbedit**).