SXPTools User Guide

(Version 2.0)

David Finlayson
U.S. Geological Survey
400 Natural Bridges Drive
Santa Cruz, CA
dfinlayson@usgs.gov

Introduction

SXPTools is a collection of command line programs that are used to enhance the quality of SEA SWATHplus interferometric sidescan sonar data. The programs are primarily focused on enhancing the backscatter quality, but there is some benefit to the bathymetry as well (and little harm is done to the bathymetry). SXPTools is a research project of the US Geological Survey and should be used with the understanding that unintentional errors in the code may have undesirable effects on the accuracy and suitability of your survey results. By using these tools you are participating in a research project. Bugs should be reported to me as you uncover them for correction in the next release. The appendix to this document contains a complete disclaimer.

This document describes the recommended installation and usage of the SXPTools programs but feel free to experiment with your data to find the best process for your own workflow. If you do have recommendations that you would like to share with the rest of the (smallish) userbase, please contact me and I will add them to this manual.

Installation

Step 1: Install a Java virtual machine.

You must have a Java 5.0 or newer compatible runtime available at the command line for running SXPEGN. You can find out what version of Java (if any) you have installed by opening up the command line (Bash on Unix; DOS on Windows) and typing the following command:

```
$ java -version
java version "1.6.0_07"
Java(TM) SE Runtime Environment (build 1.6.0_07-b06-153)
Java HotSpot(TM) 64-Bit Server VM (build 1.6.0_07-b06-57, mixed mode)
```

If you see a java version string that is greater than 1.5 you are good to go. However, for the best performance on workstation hardware, you should have the server version of Java. Try adding the ?-server? flag to the above command. If you still see the version string as before you have the server virtual machine, otherwise you will see an error.

If you have an out-of-date version of java or no java at all, I recommend downloading the JDK 6 Update 11 (or newer) from Sun:

http://java.sun.com/javase/downloads/index.jsp

You want the JDK not the JRE version (the JDK contains the advanced -server hotspot compiler.

On OS X:

Leopard comes with Java 5 and if you are up-to-date on your patches, you should already have Java 6.0 (it was installed as an update a few months ago). However, if for some reason you do not have Java 5 or Java 6, navigate your browser to the Apple web site and download the latest version of Java for your machine:

http://developer.apple.com/java/download/

Step 2: Installing the software.

On Windows:

Run the `setup.exe` program. All of Step 2 is automatically completed. You can also manually install by following the Unix directions below.

On OS X/Unix:

To install the programs unzip the sxptools.zip file into a folder without spaces in the path name. For example: ~/usr/local/sxptools.

Add the programs to your PATH.

Run the program create_bin.sh to create the launcher scripts.

Add the ~/usr/local/sxptools/bin directory to your path.

Update the launcher scripts to reflect your installation directory. If you do not have the ?-server? virtual machine (see Java section above) remove the ?-server? flag from the command line.

Step 3: Test your installation.

Open a new console (or DOS box) so that your path variable is updated. Try typing the commands: "sxpascii -h", "sxpegn -h" and "sxpsplit -h" at the prompt. You should see a brief description of the program arguments if your installation is properly set up.

Quick Start

1. Create processed bathymetry files (`*.sxp`) from the raw data.
2. Move all of the `*.sxp` files you want to process to a directory called "soundings"
3. Create an empty directory called "`egn`".
4. Run `SXPEGN -v soundings egn`

   If this is the first time you have run sxpegn in this directory, it will create an SXPTOOLS.PROPERTIES file and stop.
5. If necessary, edit the `SXPTOOLS.PROPERTIES` file to reflect the offsets you used during your survey. The default offsets are for the R/V Snavely with the CRP virtualized to the head.
6. Run `SXPEGN -v -c soundings egn` again. Wait for the processing to complete.
7. You can now Grid your SXP files foundin the "egn" folder in Grid Proc or Caris. Note that the backscatter is much improved over the original files.
8. Create another folder called "filled"
9. Run SXPMAGIC -v egn filled. Wait for the processing to complete
10. Re-grid the data in the filled folder using Grid Processor or Caris. You should now see less of a nadir artifact. This command moves soundings into the nadir to fill the gap, so you may not want to use `sxpmagic` if bathymetric accuracy is more important than the nadir hole.

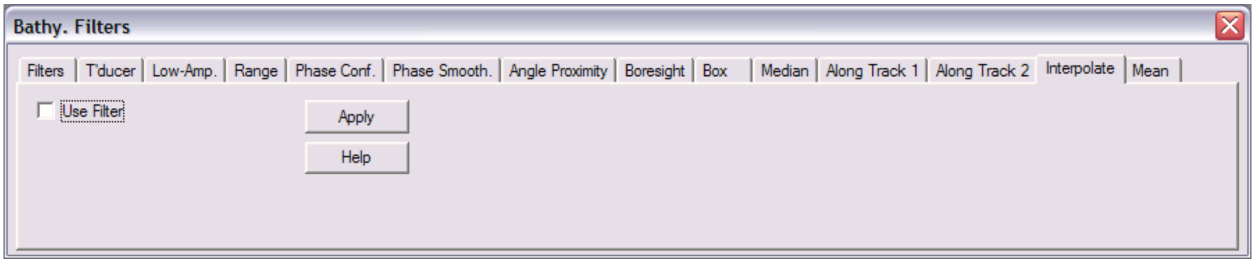
USAGE

Step 1: Create an SXP file in Swath Processor.

SXPEGN has only been tested against data sets where all motion compensation and georeferencing was performed by the Swath Processor software at the time the SXP file was created. If the SXP file headers do not contain the true attitude and position of the transducers (both horizontal and vertical), there is no way this program can do a full radiometric correction of the dataset.

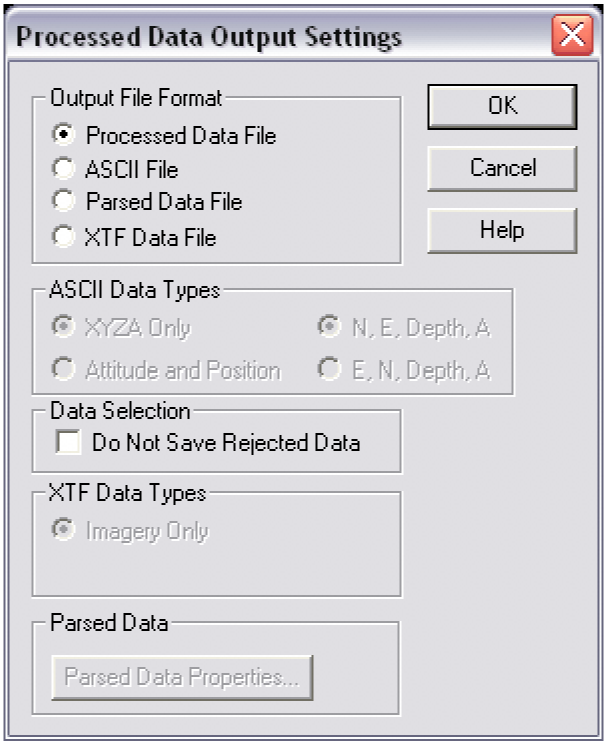
Using the bathymetric filters in Swath Processor, filter the bathymetry aggressively. The goal is to have high confidence in the x,y,z coordinate of each accepted sample. In particular, try to filter out all non-seafloor returns (schools of fish, sea-surface, multipath, etc.) that might confuse the statistical model what average sea floor returns look like.

If aggressive bathymetric filtering is resulting in too few soundings for your desired backscatter resolution, you can use the Interpolate filter in newer versions of SWATH Processor to increase the backscatter resolution. This setting uses valid bathymetric samples to "Interpolate" X,Y,Z coordinates for rejected data thereby greatly increasing the backscatter resolution. This works because many rejected data points are actually valid backscatter that has a bad phase angle. Not always though, sometimes samples are rejected because they are fish balls in the water column or surface returns, you don't want to use the interpolation filter on lines with lots of surface return or lots of water column artifacts.



☐ Use Filter

For older versions of SWATH Processor that do not have the Interpolate filter, go to the Output Processed File dialog in the sidebar of the main windows of Swath Processor, open up the Props... > Processed Data Output Setting dialog and select Output File Format: Processed Data File; Data Selection: (uncheck) Do Not Save Rejected Data. The dialog should look like the following picture:



Output File Format
● Processed Data File
○ ASCII File
○ Parsed Data File
○ XTF Data File

ASCII Data Types
● XYZA Only ● N, E, Depth, A
○ Attitude and Position ○ E, N, Depth, A

Data Selection
☐ Do Not Save Rejected Data

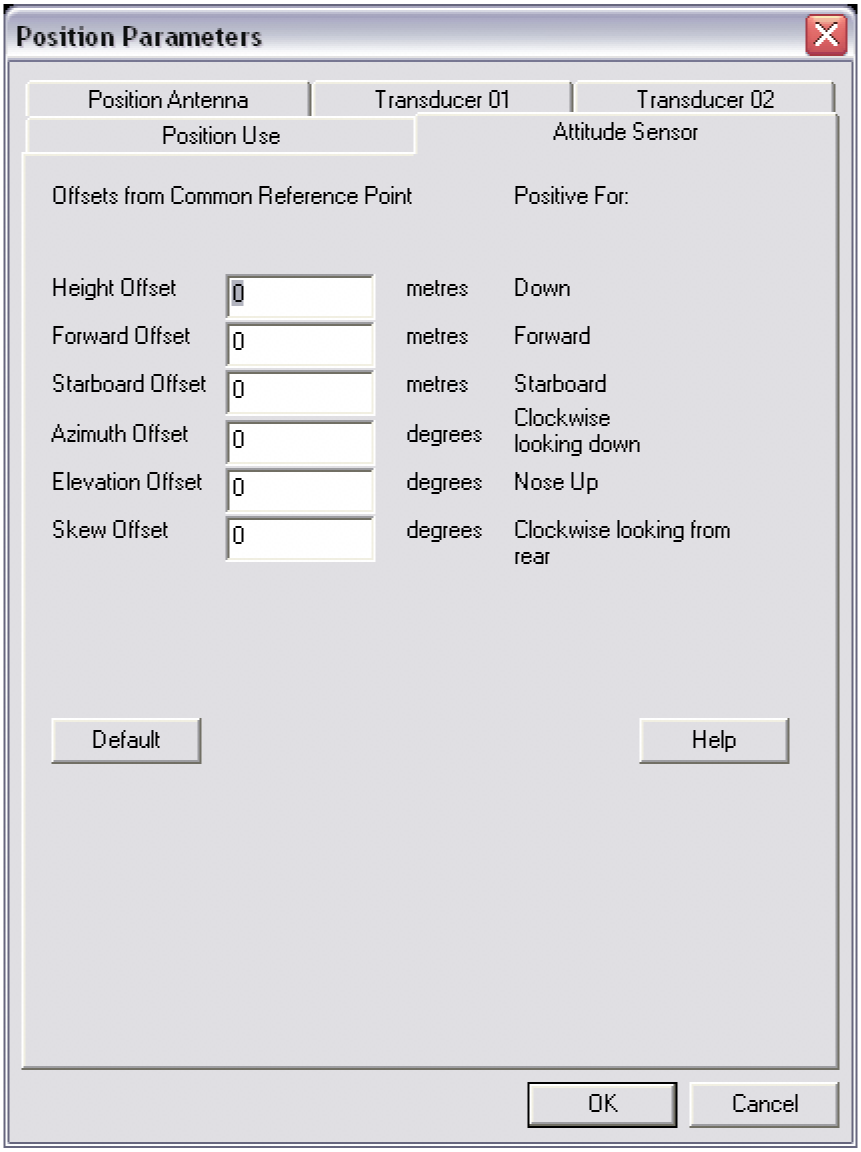
XTF Data Types
● Imagery Only

Parsed Data

SXPEGN has an interpolate filter built in (which inspired the SEA version). Either way, your backscatter resolution will be much higher; up to the theoretical resolution of the system of around 5 to 10 cm. However, there are two caveats: (1) your sxp files will be much larger and take longer to process several hours instead of several minutes. There is no reason to do this for regional data sets where 1 m resolution (or lower) is sufficient; and (2) Do not try this when there are a lot of artifacts such as fish in the water or strong surface returns, as this will result in fish balls being mapped onto the sea floor resulting in a patchy backscatter. In this later situation, try less aggressive bathymetric filtering to increase the number of valid bathymetric samples up to the backscatter resolution you need (still try to avoid selecting artifacts as valid sea floor returns).

Before closing down Swath Processor. Make a note of the offsets between the common reference point and motion sensor (found on the Attitude Sensor tab of the Position

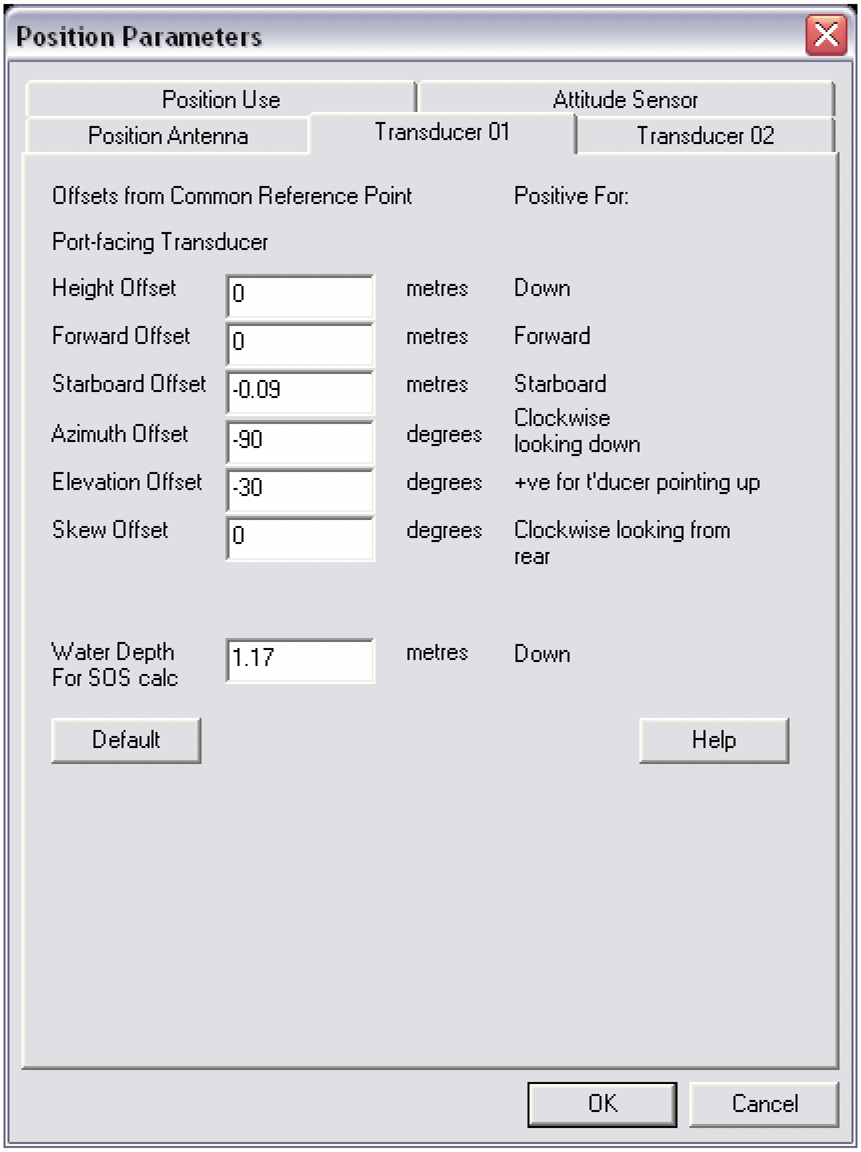
Parameters dialog):



Offsets from Common Reference Point | Positive For:

Height Offset	0	metres	Down
Forward Offset	0	metres	Forward
Starboard Offset	0	metres	Starboard
Azimuth Offset	0	degrees	Clockwise looking down
Elevation Offset	0	degrees	Nose Up
Skew Offset	0	degrees	Clockwise looking from rear

Here you can see we locate the CRP at the MRU so there are no offsets in our configuration. You will also need the offsets between the CRP and the port and starboard transducers. That information is found on the Transducer 01 and Transducer 02 tabs of the same dialog (only the Transducer 01 (port-facing) is shown):



Offsets from Common Reference Point — Positive For:

Port-facing Transducer

Height Offset	0	metres	Down
Forward Offset	0	metres	Forward
Starboard Offset	-0.09	metres	Starboard
Azimuth Offset	-90	degrees	Clockwise looking down
Elevation Offset	-30	degrees	+ve for t'ducer pointing up
Skew Offset	0	degrees	Clockwise looking from rear
Water Depth For SOS calc	1.17	metres	Down

Step 2: Organize your files

The program works on a directory of SXP files. It is assumed that all of the sxp files in the directory are from the same survey area and were collected by the same instrument in the same configuration, using the same power and pulse-length settings. (Note that in the future, I may be able to relax the power and pulse length settings restrictions).

I usually create a directory called “soundings” to hold my unprocessed SXP files and I make a second directory called “amplitude” to hold the processed SXP files. It doesn?t matter.

Step 3: Create a properties file.

The sxp file header does not contain the offsets from the CRP to the MRU or the Transducer. It is necessary to supply these values to the program in a “properties” file.

The first time you execute sxpegn in a new directory it will stop with an error saying the sxptools.properties file not found! And, it will create a template of the properties file which you can edit in a text editor:

```
# sxptools.properties
# version 1.0
#
# Edit these values to match the offsets entered into the Swath Processor
# program at the time the SXP files were created.

# Offsets from the CRP to the MRU (in meters)
CRP_TO_MRU_STARBOARD=0.0
CRP_TO_MRU_FORWARD=0.0
CRP_TO_MRU_DOWN=0.0

# Offsets from the CRP to the starboard transducer (in meters)
CRP_TO_STBD_TXER_STARBOARD=0.055
CRP_TO_STBD_TXER_FORWARD=0.0
CRP_TO_STBD_TXER_DOWN=0.06

# Offsets from the CRP to the port transducer (in meters)
CRP_TO_PORT_TXER_STARBOARD=-0.055
CRP_TO_PORT_TXER_FORWARD=0.0
CRP_TO_PORT_TXER_DOWN=0.06

# Specify the size and resolution of the Gain Normalization Table (in meters)
MIN_BORESIGHT_RANGE=0.0
MAX_BORESIGHT_RANGE=200.0
MIN_BOTTOM_RANGE=-100.0
MAX_BOTTOM_RANGE=100.0
CELLSIZE=0.25

# Specify the amplitude histogram expansion parameters
AMPLITUDE_HISTOGRAM_MIN_IN=0.8
AMPLITUDE_HISTOGRAM_MAX_IN=1.2
AMPLITUDE_HISTOGRAM_MAX_OUT=65535
```

Here is a brief explanation for each group of numbers:

Table 1: SXPTOOLS.PROPERTIES file keyword explanation

Parameter	**Explaination**
CRP_TO_MRU_STARBOARD	Attitude Sensor Starboard Offset (positive to Starboard) (m)
CRP_TO_MRU_FORWARD	Attitude Sensor Forward Offset (positive to Forward) (m)
CRP_TO_MRU_DOWN	Attitude Sensor Height Offset (positive Down) (m)

CRP_TO_STBD_TXER_STARBOARD	Transducer 02 Starboard Offset (positive to Starboard) (m)
CRP_TO_STBD_TXER_FORWARD	Transducer 02 Forward Offset (positive Forward) (m)
CRP_TO_STBD_TXER_DOWN	Transducer 02 Height Offset (positive to Down) (m)
CRP_TO_PORT_TXER_STARBOARD	Transducer 01 Starboard Offset (positive to Starboard) (m)
CRP_TO_PORT_TXER_FORWARD	Transducer 01 Forward Offset (positive to Forward) (m)
CRP_TO_PORT_TXER_DOWN	Transducer 01 Height Offset (positive to Down) (m)
MIN_BORESIGHT_RANGE	Minimum sounding range normal to transducer face (m)
MAX_BORESIGHT_RANGE	Maximum sounding range normal to transducer face (m)
MIN_BOTTOM_RANGE	Minimum sounding range parallel to transducer face (m)
MAX_BOTTOM_RANGE	Maximum sounding range parallel to transducer face (m)
CELLSIZE	Resolution of mean amplitude table (m)
AMPLITUDE_HISTOGRAM_MIN_IN	Lower Amplitude cut-off as a percentage of the mean
AMPLITUDE_HISTOGRAM_MAX_IN	Upper Amplitude cut-off as a percentage of the mean
AMPLITUDE_HISTOGRAM_MAX_OUT	Set the output maximum integer value to assign to AMPLITUDE_HISTOGRAM_MAX_IN above. Note the maximum value is 65536.

The first 9 values in the table should be taken from the values you wrote down in Step 1 above. They are copied directly from the Position Parameters tabs in Swath Processor. The Boresight and Bottom ranges and Cellsize set the size and resolution of the Mean Amplitude Table (blue field in Figure 1 and 2 below). The Mean Amplitude Table is oriented to the face of the transducer as shown below in Figure 1. The goal is to set the Boresight and Bottom Range large enough so that all pings will fall within the table (if a ping sample is larger than the table, the closest value is used); but not so large that you are wasting empty space (which takes up memory). The smaller the cellsize, the better the gain normalization will be. This is especially important near the transducer in shallow water where gains fall off very rapidly due to the R^2 nature of spreading loss. However, if the cellsize is set too small, the table memory footprint grows too large and you will run out of ram to hold the whole table in memory. A good rule of thumb is that if you see along-track artifacts in gain (a bright stripe along the nadir for example), decrease the cellsize. If you run out of memory, try decreasing the ranges.

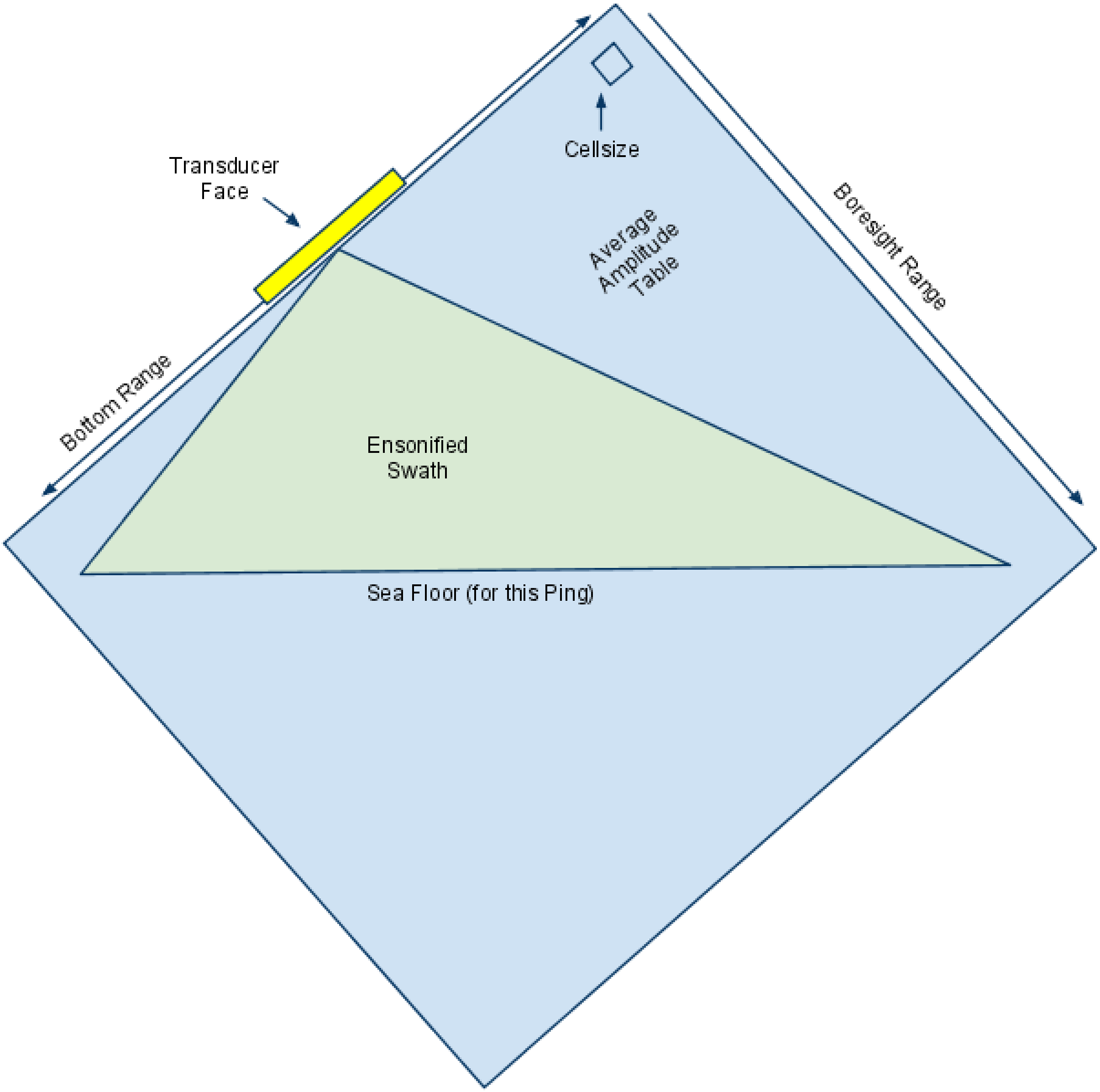


Figure 1: Orientation of the Mean Amplitude Table relative to the transducer face and ping cross-section

The mean amplitude tables that are created by the program are actually Surfer rasters. You can plot them in surfer and take a look at the mean amplitude of your system as a function of range from the transducer face. The following image shows the mean amplitude table for the port transducer on a recent survey in Puget Sound using the 234 kHz SWATHplus-M system. Maximum depths during the survey were about 80 m with a maximum range of about 200 m. In the reference frame of the transducer (30 degree elevation angle), this translates to a bottom range of about -80 to 80 and a boresight range of 0 to 200 m as shown in Figure 2.

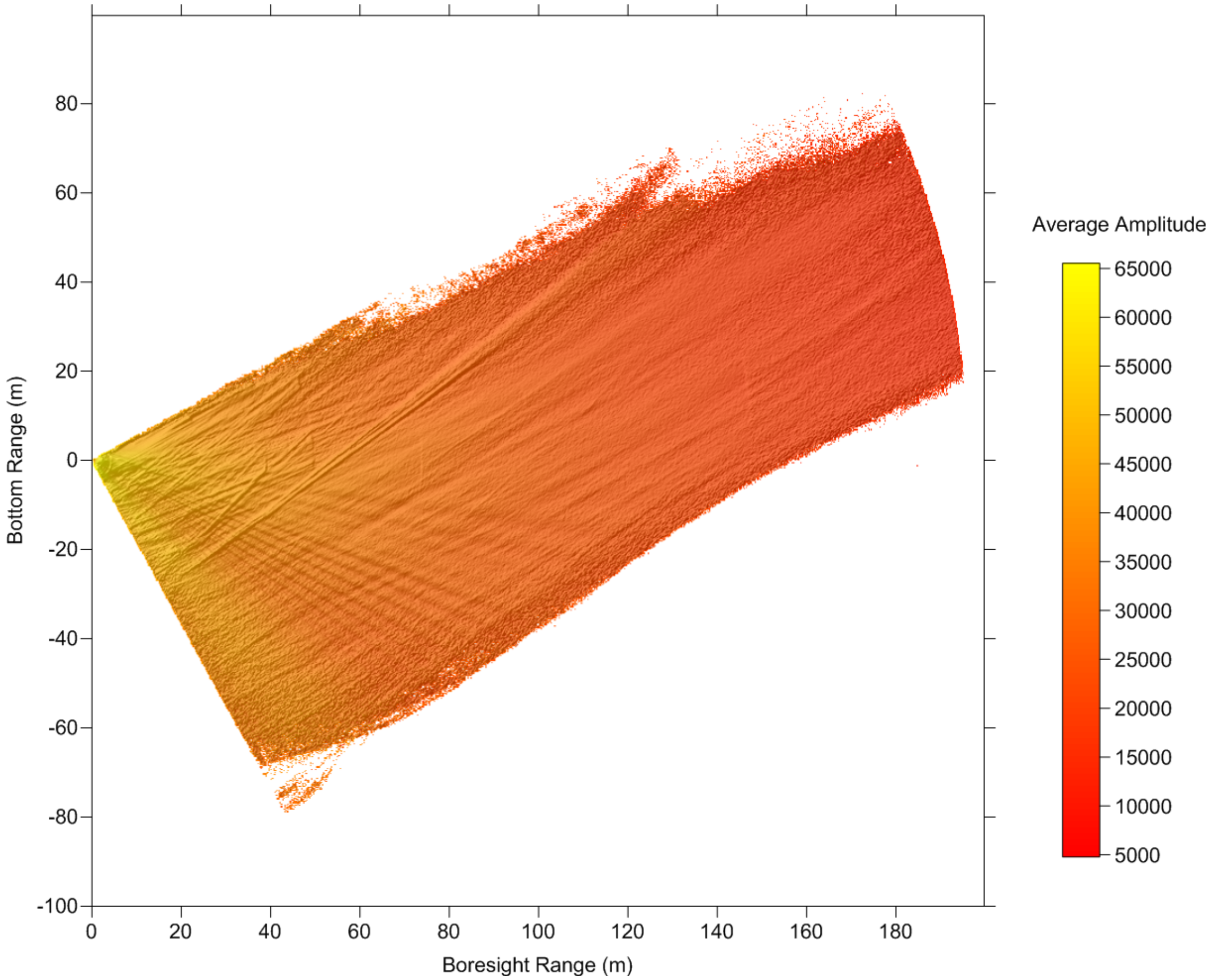


Figure 2: The Mean Amplitude Table for the port transducer after processing thousands of pings. (shading added to enhance contrast)

There is a lot of interesting information in the average amplitude tables used by the program (note the beams radiating out from the transducer, for example). The algorithm section has more details on how the tables are used to normalize the sonar gain.

Step 4: Run the program SXPEGN

You run the program by supplying an input directory as the first argument and an output directory as the second argument. The arguments are relative to the directory you are presently in, so for example, if my directory looks like this:

```
david@Ping:~/Projects/BlockH$ ls
egn soundings
```

Where my source sxp files are in the `soundings` directory (there are only 2 for this example):

```
david@Ping:~/Projects/BlockH$ ls soundings/
JD203_135.sxp jd344_007.sxp
```

and my destination directory is “egn”. I run the program like this:

```
$ sxpegn -v -c soundings/ egn/
```

the “-v” switch turns on verbose output (which is REALLY verbose right now) and will keep you informed about the progress of the program. This is really reassuring when you are processing 10s of Gb of data and the program will run all night. The "-c" switch turns on special processing for CARIS. Caris doesn't use the processed amplitude field in SXP files and so this switch overwrites the raw amplitude with the processed amplitude data.

You will see the program launching threads for the various procedures of the system and it will show you errors and inconsistencies it finds in the SXP files themselves (there are many). Generally speaking, the program will only continue to process if it has a suitable way of handling the errors or warnings.

Depending on the size of your survey and your computer hardware, it can take a few minutes to many hours to process a survey. Currently, if any fatal errors occur, you will need to correct the situation and begin again.

Step 5 (Optional): Run the program SXPMAGIC

SXPMagic is a program with highly experimental algorithms. Right now, the main benefit to using the program is its nadir filling algorithm which fills both the bathymetry and backscatter nadir at the same time. Run sxpmagic after running sxpegn.

Create a new directory to hold the "filled" sxp files:

```
david@Ping:~/Projects/BlockH$ ls
egn filled soundings
```

Now we will use the egn directory as the source and "filled" as the destinations when we run the program:

```
$ sxpmagic -v egn filled
```

Again, the program will run for a while depending on the size of your survey and your hardware. The resulting sxp files will have a greatly reduced nadir artifact. In many habitat mapping situations, these files are much easier to classify. However, if you are doing a hydrographic survey, you should use caution. The algorithm works by spreading out and smoothing good soundings near the nadir until the nadir gap is full. Effectively, the algorithm stretches out the bathymetry in the across-track direction to fill the nadir hole. You may find this an unacceptable degradation of survey accuracy. Ask yourself if this is more or less faithful than a straight-line interpolation through the nadir as most griding algorithms will do?

Step 5: View the results

Once the program has completed, you should see new SXP files in your output directory. You should be able to view your SXP files in Grid Processor or any other software that is designed to read SXP files. These files consist of only valid bathymetry and backscatter data. Data that couldn?t be corrected is removed to save on file sizes and further processing time. The normalized backscatter is put into the “processed amplitude” field of the SXP file unless the "-c" switch is used (for CARIS mode) which will overwrite both the processed and raw amplitude fields. Note that both CARIS and SonarWizMap require high-resolution SXP files for good backscatter results. I find that GridProc actually does the best job at mosaicing the backscatter right now; it may be that the histogram trimming settings are not optimized for the other programs.

Please let me know how things go with your data set and don't hesitate to contact me with questions or comments:

Detailed Program Description

Each program in the sxptools collection implements command line options. All programs will respond to "-h" which displays a short help screen to remind you of the program arguments and optional switches. For example, type:

```
$ sxpegn -h
```

Returns the following text:

```
Usage: java -jar sxpegn.jar [OPTIONS] indir outdir
Empirical Gain Normalization processing for SWATHplus backscatter data

Arguments:
    indir      source directory of unprocessed sxp files
    outdir     destination directory where processed sxp
               files will be placed.

Options:
   -H           display this help message and exit
   -h           display this help message and exit
   -j JOBS      number of JOBS to run concurrently [default: 2]
   -V           print the version information and exit
   -v           verbose output mode on [default: off]

Pre-Processing Filter Options:
   -s           Skip pre-processing and use existing gain tables
                [default: false]
   -d DEPTH     maximum accepted sample DEPTH [default: 300.0]
   -e DEPTH     minimum accepted sample DEPTH [default: 0.0]
   -m MULT      reject samples that are greater than MULT
                times the mean water depth [default: 10.0]
   -n MULT      reject samples that are less than MULT
                times the mean water depth [default: 0.0]
   -p RANGE     maximum horizontal sample RANGE [default: 300.0]
   -q RANGE     minimum horizontal sample RANGE [default: 0.0]

Post-Processing Filter Options:
   -r RESOLUTION output across-track sample RESOLUTION in meters
                 default: [0.25]
   -c            Overwrite raw amplitude value with processed
                 value (CARIS SIPS requires this) [default: false]

Report bugs to David Finlayson <dfinlayson@usgs.gov>
```

The Arguments are required inputs, usually the name of a directory or file on which the program operates. The Options are optional switches that modify the default behavior of the program. Each optional switch is briefly described with its default setting indicated in brackets. Default settings are usually appropriate for a first-run of the program until a specific setting obviously needs to be changed. All of the programs are safe to experiment with, they never modify the original file and always warn if they are about to over write an output file. The following section gives more detailed explanations of the programs in SXPTOOLS and the optional parameters.

SXPEGN

```
Usage: java -jar sxpegn.jar [OPTIONS] indir outdir
Empirical Gain Normalization processing for SWATHplus backscatter data

Arguments:
    indir      source directory of unprocessed sxp files
    outdir     destination directory where processed sxp
               files will be placed.

Options:
   -H           display this help message and exit
   -h           display this help message and exit
   -j JOBS      number of JOBS to run concurrently [default: 2]
   -V           print the version information and exit
   -v           verbose output mode on [default: off]

Pre-Processing Filter Options:
   -s           Skip pre-processing and use existing gain tables
                [default: false]
   -d DEPTH     maximum accepted sample DEPTH [default: 300.0]
   -e DEPTH     minimum accepted sample DEPTH [default: 0.0]
   -m MULT      reject samples that are greater than MULT
                times the mean water depth [default: 10.0]
   -n MULT      reject samples that are less than MULT
                times the mean water depth [default: 0.0]
   -p RANGE      maximum horizontal sample RANGE [default: 300.0]
   -q RANGE      minimum horizontal sample RANGE [default: 0.0]

Post-Processing Filter Options:
   -r RESOLUTION output across-track sample RESOLUTION in meters
                 default: [0.25]
   -c            Overwrite raw amplitude value with processed
                 value (CARIS SIPS requires this) [default: false]

Report bugs to David Finlayson <dfinlayson@usgs.gov>
```

Description

SXPEGN applies an empirical gain normalization procedure to remove systematic anomalies from the acoustic backscatter of the SWATHplus system. The technique is very simple, but powerful. The algorithm proceeds in two steps:

1. Build a table of average amplitude values as a function of boresite range and bottom range (see Figures 1 and 2 above) from the transducer face. The table is built empirically by the SXPEGN program by scanning the amplitude value for every sounding in the survey and placing it into the appropriate bin in the table. Separate tables are built for the port and starboard transducers. The larger the survey, the more samples are analyzed, the more robust the model of the sonar system becomes.

2. Normalize the amplitude of each sounding in the survey with the average amplitude value stored in the appropriate bin in the table. Mathematically:

$$a_n = \frac{a}{\overline{a}(x,y)}$$

Where a_n is the normalized amplitude; a is the raw sounding amplitude; and $\overline{a}(x,y)$ is the average amplitude stored in the table at the appropriate boresite range (x) and bottom range (y) (again, see Figures 1 and 2 above).

Arguments and Options

indir - A directory containing the sxp files to analyze. Hundreds of files can be stored here. On Unix systems, you can also set symlinks here to prevent making unnecessary copies of the files.
outdir - Processed files will be placed here with the same name as the input files. The program will not overwrite files as a safety precaution.

-j   This switch controls the number of jobs (sxp files) processed simultaneously. SXPEGN is a parallel processing program and will launch 1 job for each processor on your machine by default. This may not be desirable if other users/programs on the machine need cpu resources or also, if you are debugging the program. I run this program on 8-core machines and achieve throughput rates of 6 mb/s.

-V Version (build number) and license information

-v Verbose output mode. SXPEGN was programed in the Unix style: if the program has nothing to say, it will say nothing. However, for any program that will run for several hours, it is nice to get some feedback that everything is going OK. Use this swithc to turn on output. It will display what threads are being launched and a little message telling you how long until processing will be completed.

-s Skip pre-processing. IF you have previously built average gain tables that you are happy with, you can copy them into this directory and have the program use those instead of building new ones. This cuts the processing time in half and may lead to better results if you are processing only a hand full of files.

-d -e -m -n -p -q These options are all ways of trimming the swath prior to processing.

-r Resolution. Most post-processing programs cannot handle the data volumes produced by the swath plus. This filter down-samples the swath by setting an output resolution. It is similar to the mean filter in SWATH Processor.

-c Caris mode. Caris will not see the results of SXPEGN without this switch. It overwrites both the raw and processed amplitude fields with the results of SXPEGN processing.