A Tutorial for

SeismicUnixGui,
a graphical user interface for Seismic Unix
(John Stockwell)

Juan M. Lorenzo

Version: 0.87.1 (March, 2024)

1 General Information

1.1 Acknowledgements

This project is possible only because of the selfless work of others. I have borrowed notes extensively from the Colorado School of Mines website (Stockwell, 1999) for Seismic*nix. Over the years, many students have also contributed to these notes: Class of 2008: Erin Walden, Kody Kramer, Erin Elliott, Andrew Harrison, Andrew Sampson, Ana Felix, John D'Aquin, Russell Crouch, Michael Massengale, and David Smolkin; Chang Liu (2013), Nevra Bulut (2019), Daniel Locci-Lopez (2023).

I would greatly appreciate any, and all questions you have regarding installation and running of any of the programs to help us continue the development of SeismicUnixGui. Please send your questions to ***gllore@lsu.edu***. Please indicate what your operating system is, whether you have administrative privileges and send a copy of the data flows with which you are having an issue.

Thanks,

Juan Lorenzo, Socorro, NM, June 2023.

1.2 What is SeismicUnixGui?

SeismicUnixGui, a graphical user interface (GUI), serves to select and build sequences of Perl modules and their parameters. SeismicUnixGui generates two versions of these instructions in text files. These text files contain a shell and a Perl script version that can be modified and executed independently of this GUI and from the command line.

Seismic*nix (Stockwell, 1999) is a widely distributed free software package for processing seismic reflection and signal processing. In Seismic*nix (SU), a sequence of independent programs generate a stream of data that can be displayed on the screen or saved directly to a file. The data file that is read and the generated output data are handled internally by stdin, stdout functions in C while the data exchanges between programs and the linux operating system are managed from the command line via pipes "|" and redirections "> or <" respectively. Traditionally, the instructions on the command line can be assembled and saved as re-usable bash scripts. SeismicUnixGui assembles these same scripts for the operating system to run with the help of

modules written in Perl. SeismicUnixGui generates these scripts within the directory of the user and these scripts can be run independently of SeismicUnixGui running.

SeismicUnixGui is written using Perl/Tk which is mature, well-documented Perl module that allows its users to construct graphical user interfaces.

In a classroom environment, shell scripting of SU modules engages students and helps focus on the theoretical limitations and strengths of signal processing. However, complex interactive processing stages, e.g., selection of optimal stacking velocities, killing bad data traces, or spectral analysis requires advanced flows beyond the scope of introductory classes. In a research setting, special functionality from other free seismic processing software such as SioSeis (UCSD-NSF) can be incorporated readily via an object-oriented style to programming. Recently, we have incorporated a 1D raytracing package built by E. Vera of the University of Chile to display interactively forward model or arrival times over real data.

An object-oriented approach is a first step toward efficient extensible programming of multi-step processes, and a simple GUI simplifies parameter selection and decision making. Currently, in SeismicUnixGui, Perl (5.X) packages wrap over 300 of the most common SU modules that are used in teaching undergraduate and first-year graduate student classes (e.g., filtering, display, velocity analysis and stacking). Perl packages (classes) can advantageously add new functionality around each module and clarify parameter names for easier usage. For example, through the use of added methods, packages can isolate the user from repetitive tasks and having to keep track of the location of files within the system. SeismicUnixGui simplifies the short naming conventions for parameters (e.g., “a”, “b”, etc) with self-describing names (e.g., “first”, or “inter-gather increment”. Moose, an extension of the Perl 5 object system, greatly facilitates an object-oriented style when writing the code. Perl wrappers are self-documenting via Perl programming document markup language.

An automatic directory structure is created for the user in which data and programs are distributed according to a pre-defined hierarchy. All the directories and minimal files needed by SeismicUnixGui are created whenever a new 'Project' is created within the 'Project Selector' tool. The user can also create new projects within main GUI of SeismicUnixGui as well as selecting different projects. At all times the user can use linux commands to navigate freely through the directories. Sometimes the user may find it convenient to create new subdirectories within the existing file structure, which SeismicUnixGui would not be able to detect.

1.3 GUI Sections

1.3.1.1 Overview

The main GUI is divided into four sections (1) A top menu with an option of over 300 modules, (2) a menu on the left side for saving, opening, deleting and running a flow (sequence of assembled instructions). (3) For each geophysical module the parameter names and values are located in the central area of the image.

There is also a convenient cross symbol (**X**), in the top-left corner of the window, that will kill all open windows, if the screen becomes to cluttered.

Before running each flow, the user must give the flow a name and save it to a file. The correct clicking sequence, is always "**Save**" or "**Save As**" followed by "**Run**"

Up to four, independent processing flows can be tested in the GUI (grey, pink, green and blue rectangles). The modules within each processing flow can be rearranged using the up (inverted **V**) and down (**V**) symbols or deleted (**-).** An entire flow (colored box) can be cleared by clicking on another cross (**X),** found on the right-hand-side of the window.

Messages to the user will either appear in short-lived boxes or within the black box at the bottom of the GUI.

Data file names usually have a suffix that indicates the type of data contained, e.g., header.txt or 1.su contain data in ASCII and seismic-unix format respectively. Generally, the menus shown exclude the suffix and rely on the directory path to determine the type of file to be read. If you are reading a data file or writing a data file, the termination will determine the container directory. A common problem with not finding a file is that its suffix may be slightly different to the one that the GUI expects to find; e.g., ".sgy" instead of ".segy", or ".text" instead of ".txt"

Example 1: If "su" Is entered then the GUI will assume the data is found in directory

~/Servilleta/seismics/data/loma_blanca/053018/H/1/**su**/**user**/

Example 2: If "bin" Is entered then the GUI will assume the data is found in directory

~/Servilleta/seismics/data/loma_blanca/053018/H/1/**bin**/**user**/

Example 3: If "segy" is entered then the GUI will assume the data is found in directory

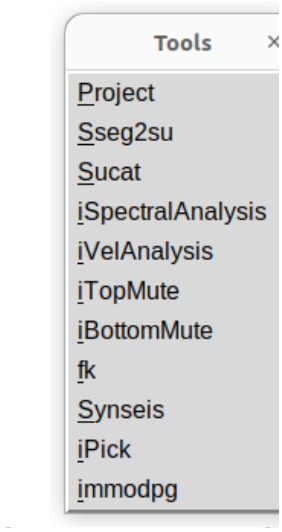
~/Servilleta/seismics/data/loma_blanca/053018/H/1/**segy**/**user**/

When, in error a proper suffix is not found, then the following directory is selected instead:

~/Servilleta/seismics/pl/loma_blanca/053018/H/1/**user**

1.3.1.2 Side Menus

1.3.1.2.1 Tools



Project: Defines the directory structure for data sets and programs in many languages, e.g. Matlab©, R, Perl etc.

Sseg2su: Converts SEG-2 formatted data into the su format which is a simplified SEG-Y format (requires Sioseis installation)

```
% Sseg2su
```

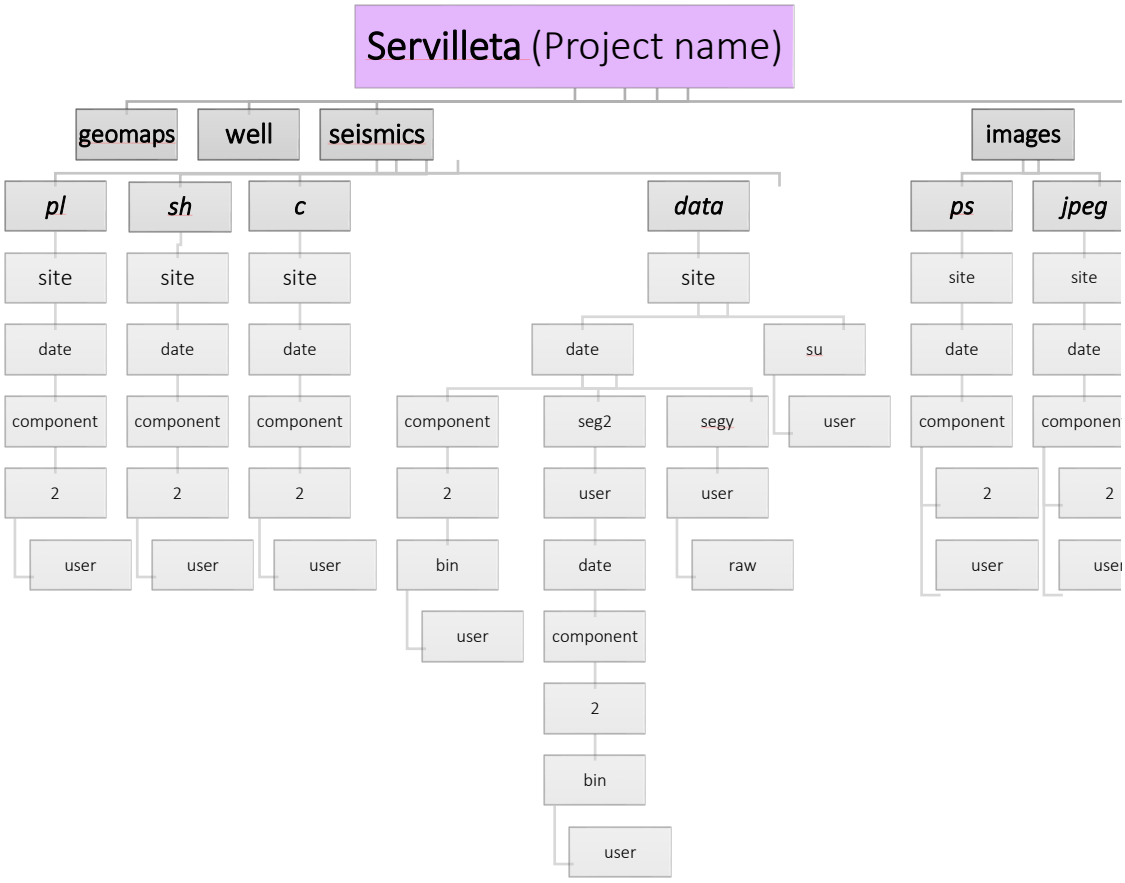
Sucat: Concatenates multiple files of any format into a single file. These files can have names related by a continuous sequence of integers, e.g., Seismic Unix data files: 1000.su, 1001.su, 1002.su. If not, a list of names can be specified. Output files from interactive muting or velocity analysis and that have specific "par" formats can be handled.

```
% Sucat
```

1.3.1.3 Some Help

Traditional manuals for each of the Seismic Unix programs are available. Instructional information will require two mouse clicks. First make sure you select the program (click of left button will darken the background) and then, select the program via a click of the right-side button of your mouse.

1.4 What is an example directory structure for a Project?



1.4.1.1 Copying data into the project directory from elsewhere

If you want to copy seismic data that are already in SeismicUn*x format (SU), copy it into:

Example: /home/**user**/Servilleta/seismics/data/ loma_blanca/053018/H/1/su/**user**

1.4.1.2 Where are my flows kept?

Example: /home/**user**/Servilleta/seismics/pl/loma_blanca/053018/H/1/**user**

1.5 Text conventions in this tutorial and their meaning

Left Mouse click is abbreviated to <MB1> Instruction

Right Mouse click is abbreviated to <MB3> Instruction

Variable names are shown in a large bold-style font.

% Command-line instructions are shown with pink background

1.6 Glossary

Term	Explanation and Example	Brief
HOME	Full linux directory path to the user's home directory, e.g. /home/xavier45	home directory path
PROJECT_HOME	Located inside **HOME** directory -- can be a soft link	Project directory path
site	e.g., Servilleta -- a National Wildlife Refuge in New Mexico, U.S.A.	name of the project area
spare_dir	can be left empty	a bonus directory
date	053018	Field work date
component	Z stands for vertical and H can be horizontal but any name is possible	Geophone particle displacement component
line	1	used to identify a profile
user	e.g., xavier45	login name
subUser	must be set to the user's login name, e.g., also xavier45	Allows groups to share Project space
flow	data_in, sugain, suximage	Sequence of programs to execute
geomaps	Directories will be created when working withmaps	Directories for third-party software (if installed and accessible)

geopsy	Geopsy software for inversion of interpreted seismic dispersion plots generated via suphasevel	Directories for third-party software (if installed and accessible)
gmt	Generic Mapping Tools	Directories for third-party software (if installed and accessible)
grass	GRASS GIS	Directories for third-party software (if installed and accessible)
matlab	Matlab ©	Directories for third-party software (if installed and accessible)
mmodpg	Forward modeling of seismic reflections and refractions using 1D Ray Theory (E.E. Vera)	Directories for third-party software (if installed and accessible)
sqlite	Databases	Directories for third-party software (if installed and accessible)

Table 1: Definitions of terms used when creating working projects

2 Demonstration Project

When either creating a new project or accessing a pre-existing project instances, always start by running the following instruction:
% SeismicUnixGui

2.1 A Quick start

2.1.1 Where are my **data** sets **stored?**

Before starting a new project you should understand the file structure in which programs and data sets are stored. The main directories are shown above for the example of Servilleta in Section 1.4.

2.1.2 Install example flows and data sets

Several example projects that contain data and examples flows can help you become acquainted with the Seismic Unix Tools. For example:

- ***Servilleta contains files from the 2018 undergraduate research internship in seismology (URISE) orientation program***

The following is explained the SeismicUnixGui Installation manual (Section 1.3.6) but is repeated here for convenience of the user. Once you completely install SeismicUnixGui on your system, you can move or copy any of the accompanying demonstration folders to the home directory of the user, where /home/user is the complete path to the location of the user (="user").

```
% cp -R $installation_directory_for_demo_projects/Servilleta /home/user/
```

2.1.3 Create a new project, e.g., Servilleta (URISE demonstration data set)

The following instruction starts the program:

```
% SeismicUnixGui
```

If you do not have any projects created previously, then:

<MB1> Create New

(Otherwise, go to next section 2.2: *Open a pre-existing project*.)

After clicking on **Create New**, a default set of parameter names (e.g., **HOME**) and their values (e.g.**, /home/username**) appears:

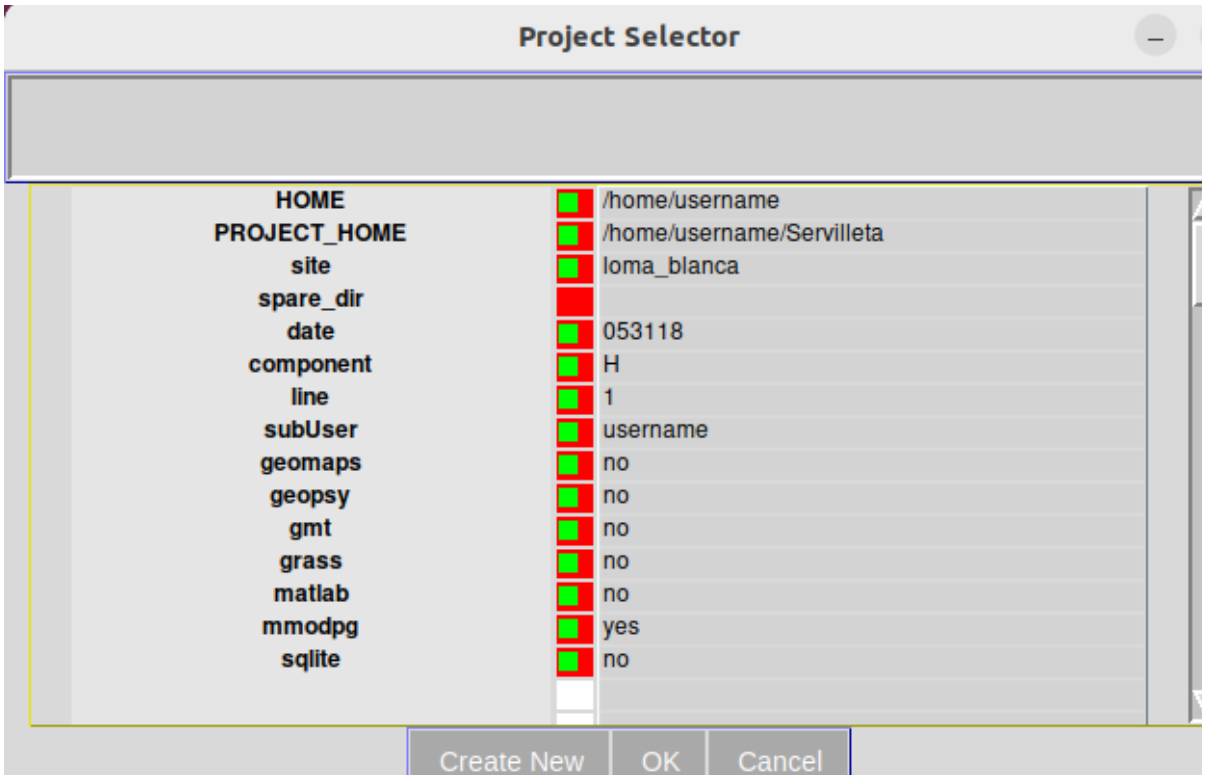


Figure 1: Screen capture of Project Selector Pane with parameters and their values.

The Project Selector pane displays several default options that work with the test data set that is included for this tutorial. The old variables are defaulted from prior projects and serve as an example to guide your input. The home directory of the user is required to follow the standard linux file structure naming system.

These options should be updated with your real username:

Parameter name	Default values
HOME	/home/**username**
PROJECT_HOME	/home/**username**/Servilleta
	Servilleta

site	053018
spare_dir	H
date	1
component	**username**
line	no
subUser	yes
geomaps	no
geopsy	no
gmt*	yes
grass*	yes
matlab*	yes
mmodpg	
r*	

Table 2: Substitute username with your login name in <u>three</u> locations.

*** if set to 'yes', then working directories will be created to accompany these programs (June, 2023)**

Finally, select: <MB1> OK

For the URISE data set, confirm you are working in a Project called "Servilleta". If you see a strange username (=gllore), this username should be replaced with your own.

In the top left menu, select <MB1> Tools->Project

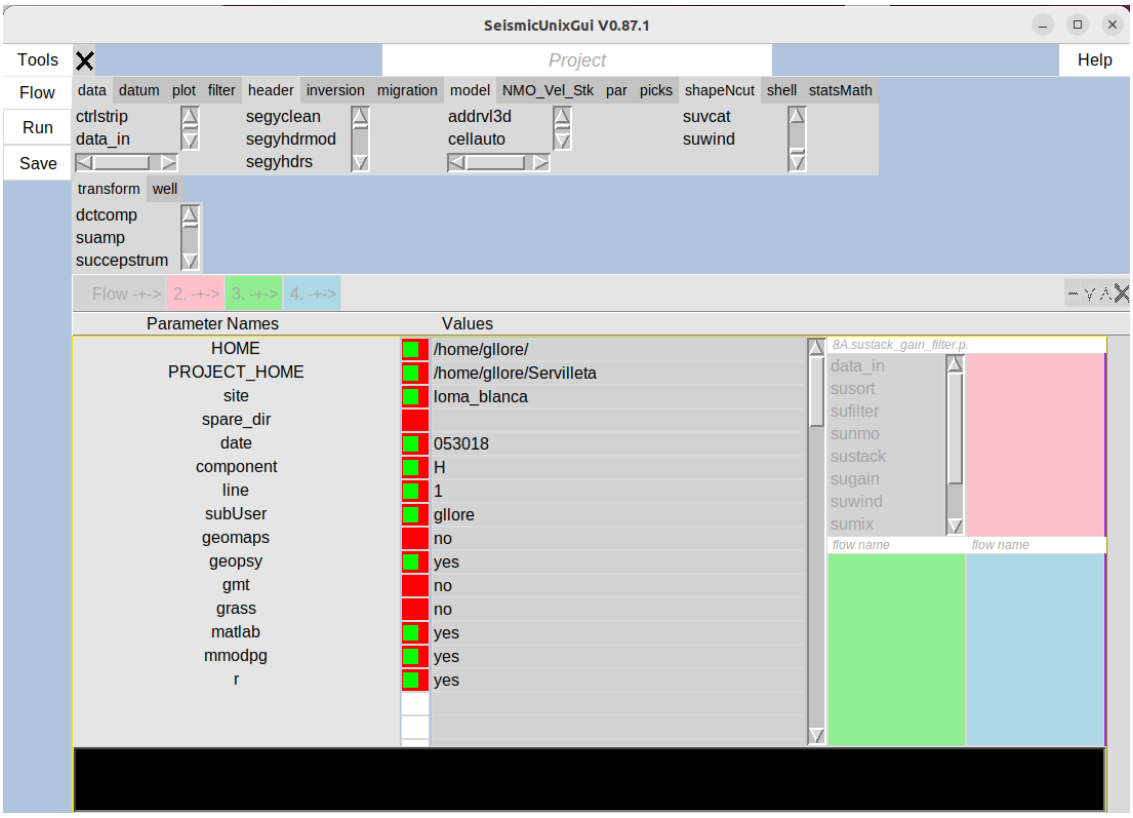


Figure 2: SeismicUnixGui main screen with Tools->Project selected and showing parameters and their values.

In the main window of the SeismicUnixGui you should see the previous changes you made to the same parameter values. The figure above (Figure 2) shows a consistent use of the username ("gllore") If, however, you have not made the proper changes, you are free to do it within this window. Then, you do the following:

In the top left menu:

<MB1> Save, followed by:

<MB1> Run

2.2 Open a pre-existing project

The following instruction starts the program, and opens the pane of the Project Selector window (Figure 3):

```
% SeismicUnixGui
```

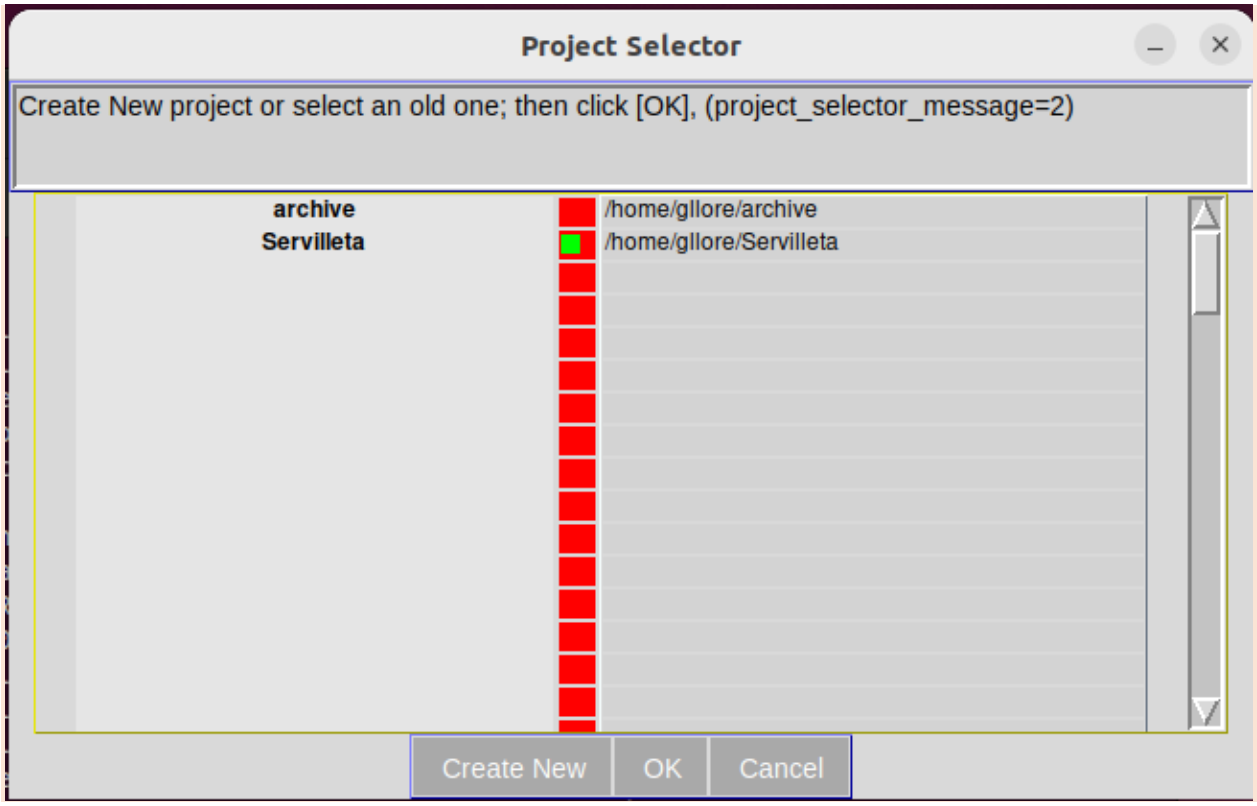


Figure 3: Project selection should have ONLY one green button when selected

If the project of interest (in this case Servilleta) is selected (square button is green):

Select: <MB1> OK

2.3 Running your first flow

Assemble a sequence of modules to carry out a processing procedure. Choose one of four differently colored flow windows (grey, pink, green and blue) in which to place your sequence. The colored window appear on the right-hand side of the main window.

A module, with a specific functionality, is selected by clicking on its name found within one of the categories on the left-hand side of the main window. There are 17 different categories that group over 300 programs.

The module name must be transferred to the list on the right by clicking one of the four different colored flow arrows, just to the right of the word “Flow”.

A final assembled flow must first be saved to a file before it is executed (**File->SaveAs**). Thereafter, all executions require that the flow be first saved before running.

In a simple sequence of modules, data are usually read in first, the data are modified and the result is placed into another file or displayed to the screen using an imaging module (e.g., suxi-

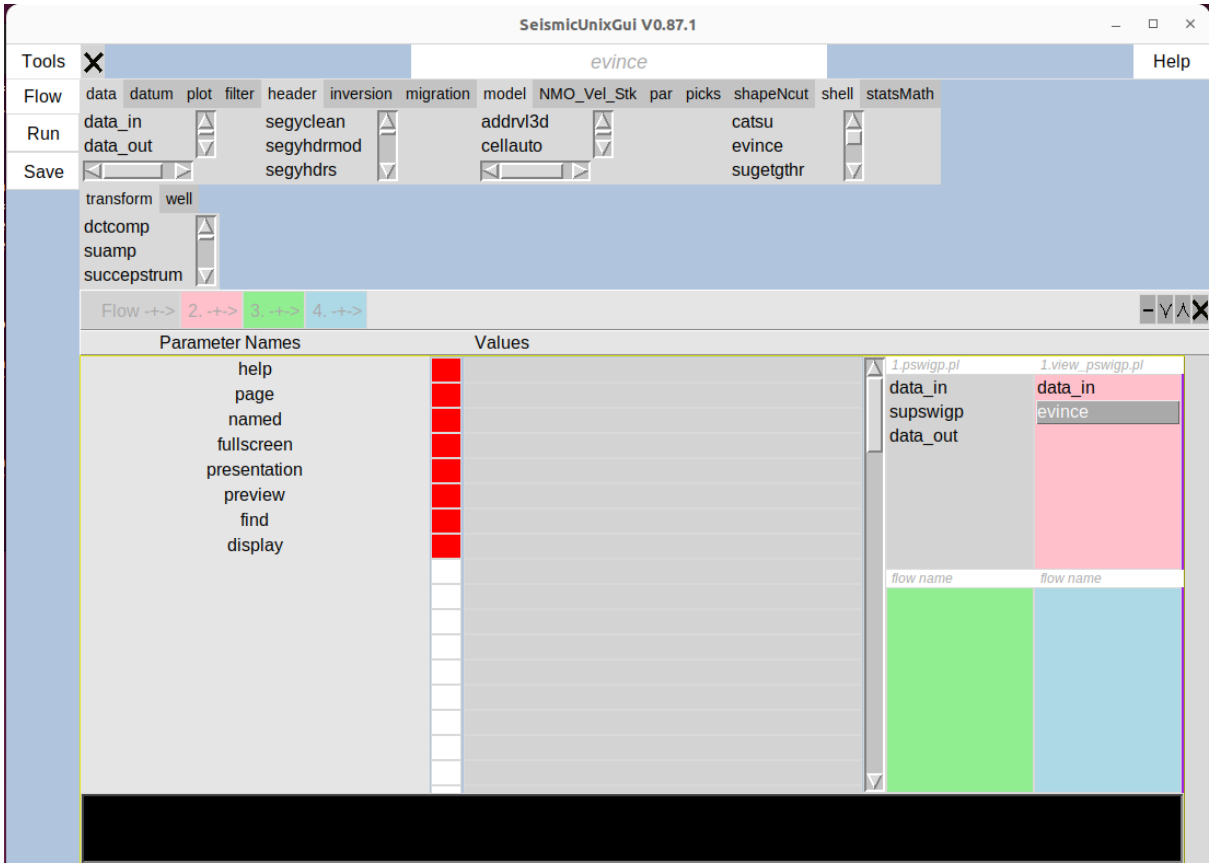


Figure 4: Main SeismicUnixGui window showing two flows to generate (1.pswigp.pl) and view a postscript plot (2.view_psuwigp.pl)

mage, suxwigb) or they can be saved as a postscript file and viewed with a free postscript viewer such as **evince** (Figure 4) as follows:

To create the left flow (grey box) in Figure 4, select the following named modules:

data_in *(*under the data category list of programs*)*

supswigp *(*under the plot category list of programs*)*

data_out *(*under the data category list of programs*)*

(Each time you select a specific module (<MB1>), its parameter names and some values will immediately appear in the center of the GUI.)

Transfer the selected program into one of the four colored flow boxes:

<MB1> Flow-+-> 1 (grey) or 2 (pink) or 3 (green) or 4 (blue)

You should be able to see the name of the program that you just selected move over to a colored box on the right-hand side of the window.

You are required to select a **Value** for **base_file_name** (= "file name").

To do so, move your cursor into the corresponding column to the right of **base_file_name**.

A click of the right-mouse-button (<MB3>) will automatically open a second window from which you can select a file, e.g. **"1.su"**.

Before you can run the program you have built, it must be saved:

For SeismicUnixGui GUI
<MB 1> File/SaveAs

Save the resultant perl script file as, e.g.,
"1.pswigp.pl"

Then, click as follows to execute Perl and Shell script flows generated by SeismicUnixGui
Tool: <MB 1> Run

To view the resultant postscript file, you can build the flow from scratch or directly load an existing version into the GUI window: **2.view_pswigp.pl.**
<MB 1> Flow-> Open

and select 2.view_pswigp.pl. Press OK:
<MB 1> OK

Then, save and execute this newly loaded file:
<MB 1> Save

<MB 1> Run

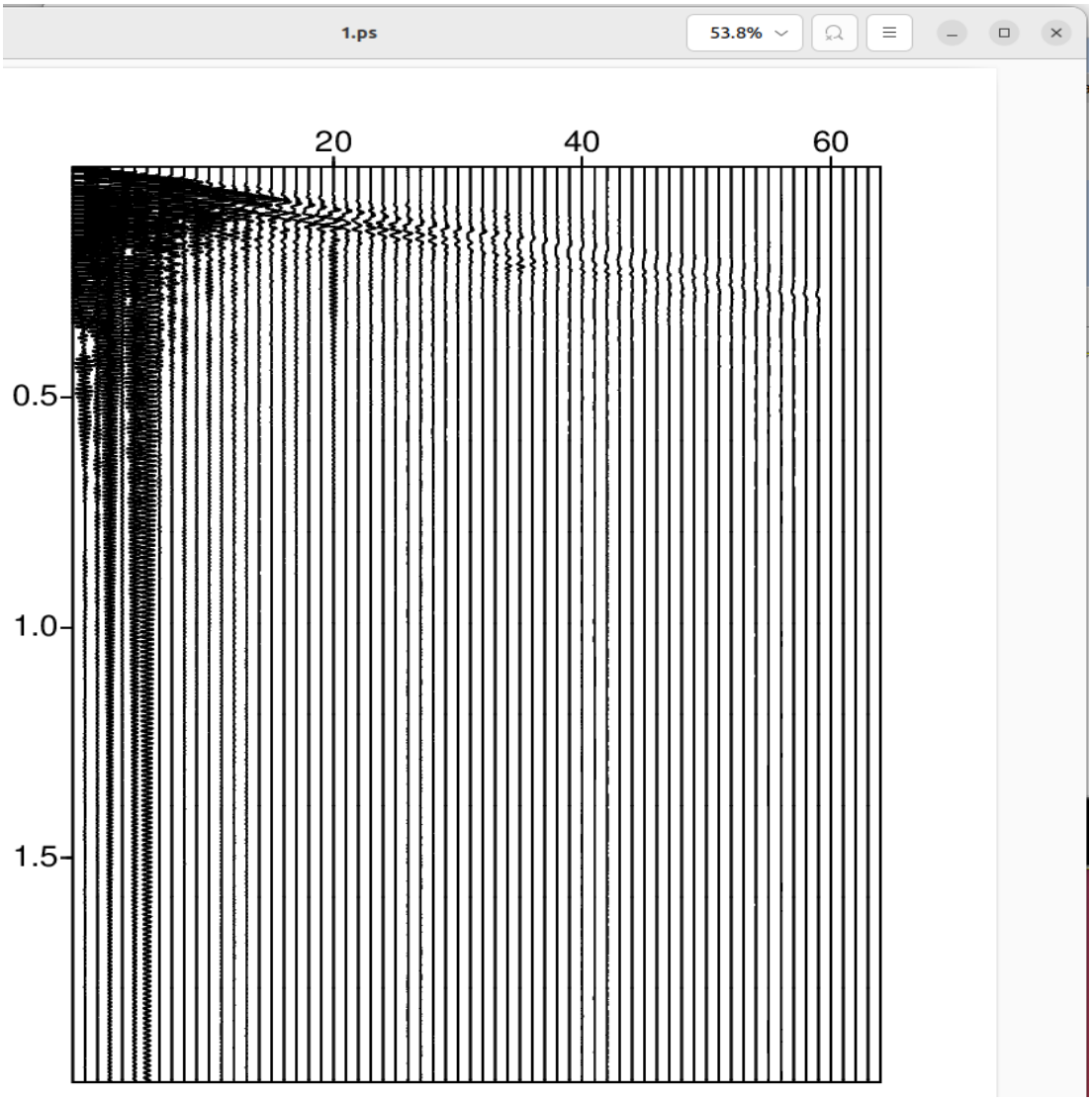


Figure 5. Postscript plot viewed using the GUI

2.3.1 Perl and Shell script flows generated by SeismicUnixGui

Alternatively, as for all flows generated by SeismicUnixGui, you may run this instruction from the command line, in most cases, in the directory where the perl flows are kept (see ***1.4.2***):

```
% /home/user/Servilleta/seismics/pl/loma_blanca/053018/H/1/user/perl 2.view_pswigp.pl
```

If you are already located in the proper directory it is sufficient to only enter the following:

```
% perl 2.view_pswigp.pl
```

Another option is to run the bash script that appears in the window whenever the Run instruction is clicked by the mouse. This command can be captured and pasted into the command line using the Mouse (copy and paste).

A third option, also using a command-line instruction, is to directly plot the postscript file using a postscript viewer such as evince:

```
% evince /home/user/Servilleta/seismics/images/ps/loma_blanca//053018/H/1/user/junk.ps &
```

(Note that 1.**pswigp.pl** is run first and **1.view_pswigp.pl** second)

2.3.2 Access to Documentation

Select <MB3> over the name of any program:

Figure 6. Conventional Seismic Unix documentation for the module: suwind

3 Simple Processing Flow: URISE Data Set, Socorro New Mexico

Each year the Earthscope Consortium in Seismology holds an orientation week as part of an undergraduate research internship experience (URISE). As part of this week of training, in the town of Socorro, New Mexico on May 30 of 2018, the students collected an active-source seismic data set, which we process using Seismic Unix.

3.1 Processing steps

The following outline is taken from a called notes.pl. This files exists in the perl flow directory (***1.4.2***) of the Servilleta project. To get there, change to the following directory:

```
% cd  /home/user/Servilleta/seismics/pl/loma_blanca/053018/H/1/user
```

To see the marked-up contents of the perl file:

```
% perldoc notes.pl
```

LOMA BLANCA

URISE 2018 survey May 30 2018
on S bank of Rio Salado
along same line as pseudo-walkaway taken on 032618
shoot-through

Acquisition parameters

Date 053018
Sample rate 1000 S/s
delrt -11 ms
rec. length 2 s
num tracr 64

Live channels 1-64			
Channel 1	**closest to recorder-- toward SE**		
Channel 64	**farthest from recorder-- toward NW**		
geophones:	**Sercel L-28-3D, 4.5 Hz, 3 components**		
geophone spacing:	**1 m**		
line orientation:	**NW-SE later shots more toward NW**		
Number of Geophones	**60**		
Shotpoint Spacing	**1 m**		
GPS is available (etrex garmin 10 m)			
	(sx-m)	**NOMINAL offset-m**	**ACTUAL (m)**
Raw SP 1	**0**	**1-60**	**0.5 - 59.5**
Raw SP 2	**1**	**0-59**	**-0.5 - 58.5**
Raw SP 3	**2**	**-1-58**	**-1.5 - 57.5**
Raw SP 4	**3**	**-2-57**	**-2.5 - 56.5**
Raw SP 60	**59**	**-58-1**	**-58.5 - 0.5**
Striker plate	**I-beam**		
Hammer	**10 lb sledge**		
No. blows	**3 per side (vertically stacked)**		
Noise sources:	**5 - 10 mph from SE**		
I-25	**to E**		

Table 3. Nominal acquisition parameters taken from the file notes.pl

3.1.1 STEP 1. File format conversion

<MB 1> Tools:->Seg2su (from GUI)

Purpose: Convert data in Seg2 to Seismic Unix (su) format

Input: **1 to 121.dat** in folder: /home/**user**/Sevil-leta/seimics/data/loma_blanca//053018/H/1/seg2/**user**/

Output:1 to 121_clean.su in folder: /home/**user**/Sevil-leta/seimics/data/loma_blanca//053018/H/1/su/**user**/

Default configuration file:
/home/**user**/Servilleta/seismics/pl/loma_blanca/053018/H/1/**user**/**Seg2su.config**

Equivalent command-line instruction if file Seg2su.config has been created.

```
% Seg2su
```

Note 1: This Tool-based instruction will work from any directory, as long as the configuration file has been created via the GUI. Sseg2su also removes some unwanted information from the headers of the files.

Note 2: This Tool requires that you have SioSeis installed. In case you do not want to install SioSeis, a set of converted files are already converted for immediate use.

Note 3: A Third-Party software (seg2segy) is available under Seismic Unix but currently does not handle directory paths preceding a file name.

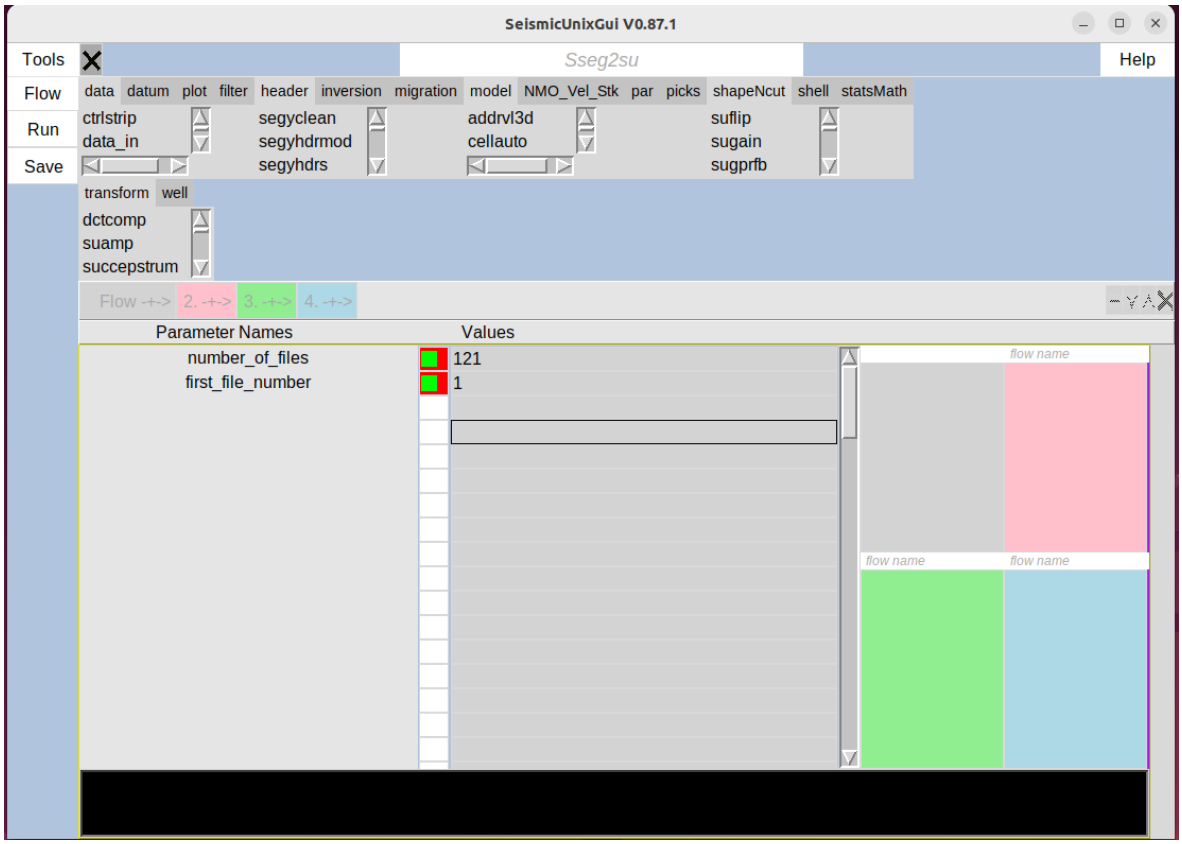


Figure 7 View of Tool Sseg2su parameters. Note that the file names comprise a numerical sequence starting at 1 and ending at 120; a total of 121 files.

3.1.1.1.1 Visually inspect output file

<MB 1> Flow->Open

Select file: 1A.view_sp_clean.pl (from GUI)

Purpose: Quality control of STEP 1

Input: **1_clean.su**, found in folder:
/home/**user**/Sevilleta/seimics/data/loma_blanca//053018/H/1/su/**user**

Equivalent command-line instruction:

```
% perl /home/user/Servilleta/seismics/pl/loma_blanca/053018/H/1/user/1A_view_sp_clean.pl
```

Output: image (Figure 8)

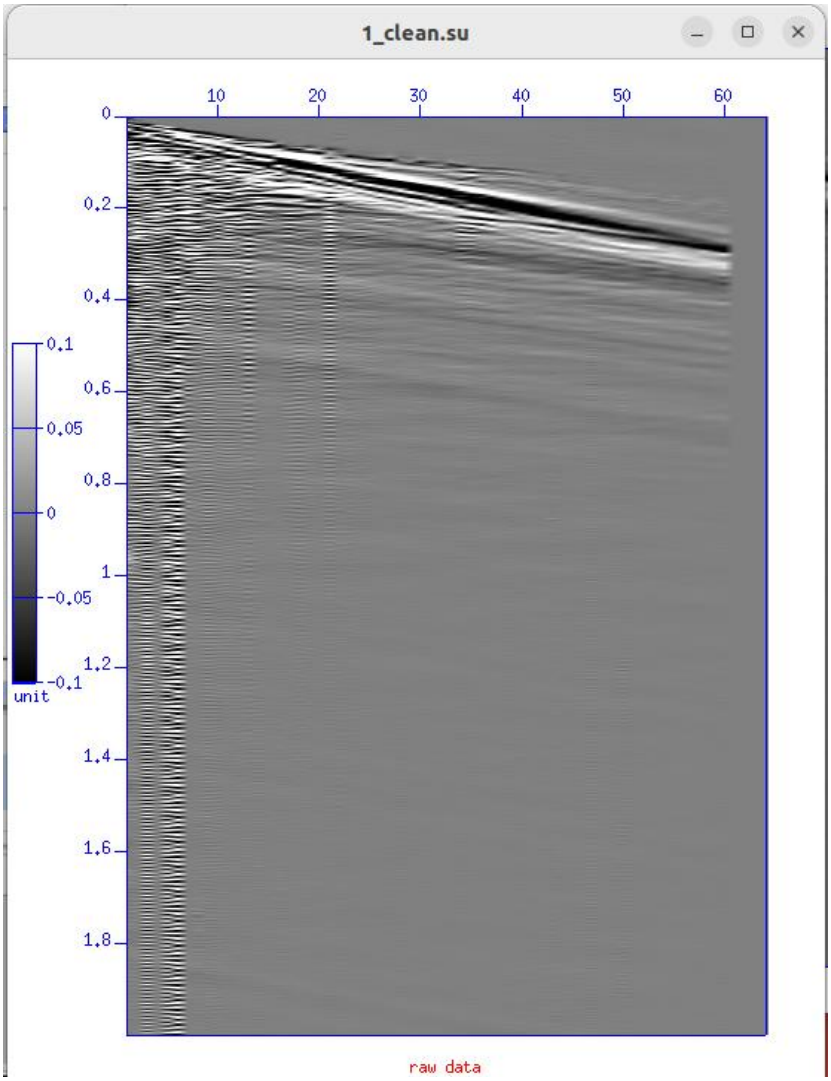


Figure 8. Screen display of seismic shotpoint gather 1 after conversion from a SEG2-formatted file and header cleaning

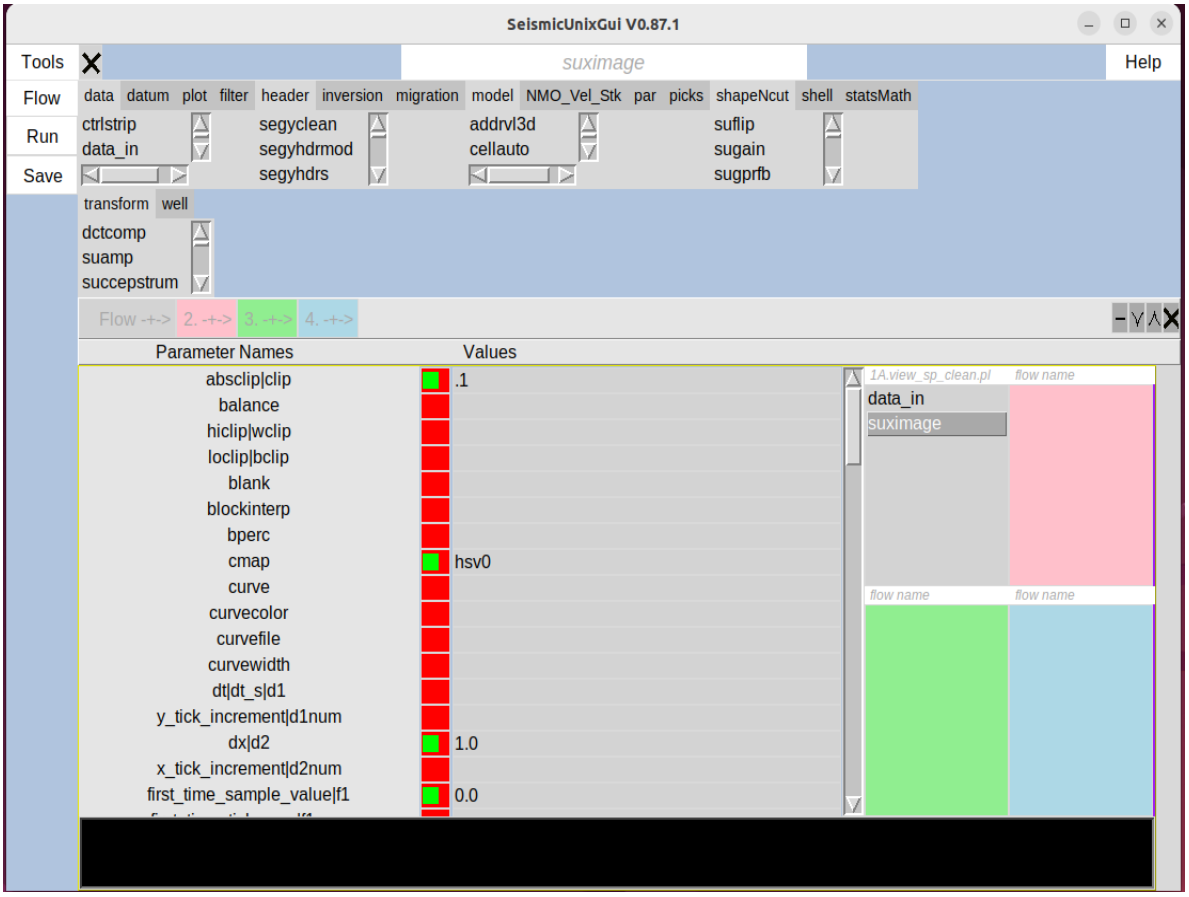


Figure 9. View of suximage parameters, second item in flow contained within file: 1A.view_sp_clean.pl.

3.1.2 STEP 2. Concatenation of seismic files

<MB 1> Tools:->Sucat (from GUI)

Purpose: Combine individual files into a single file

Input: 1_clean.su to 120_clean.su, found in folder:
/home/**user**/Sevilleta/seimics/data/loma_blanca//053018/H/1/su/**user**

Output:/home/**user**/Sevilleta/seimics/data/loma_blanca//053018/H/1/su/**user**
/1_120_clean.su

Default configuration file:
/home/**user**/Servilleta/seismics/pl/loma_blanca/053018/H/1/**user**/**Sucat.config**

Equivalent command-line instruction, if file Sucat.config has been created:

```
% Sucat
```

Note: For convenience, a copy the Sucat.config file is saved as:
/home/**user**/Servilleta/seismics/pl/loma_blanca/053018/H/1/**user**/**Sucat.config_1**

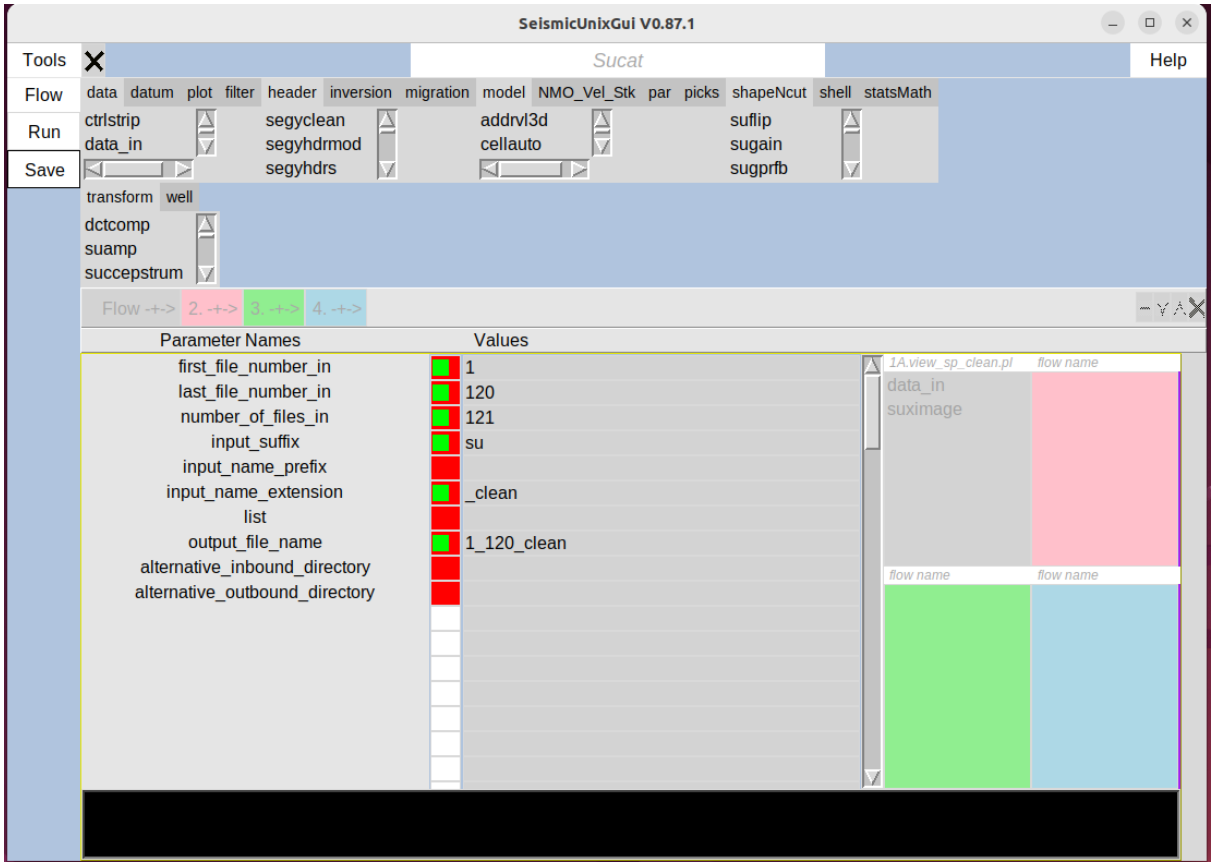


Figure 10 View of Tool Sucat parameters.

3.1.2.1 View concatenated data

<MB 1> Flow->Open

Select file: **1B.view_All_raw_data.pl** (from GUI)

Purpose: View and verify combined data set for only the first second of data

Input: /home/**user**/Sevil-
leta/seimics/data/loma_blanca//053018/H/1/su/**user**/1_120_clean.su

Output: images (Figures 11 and 12)

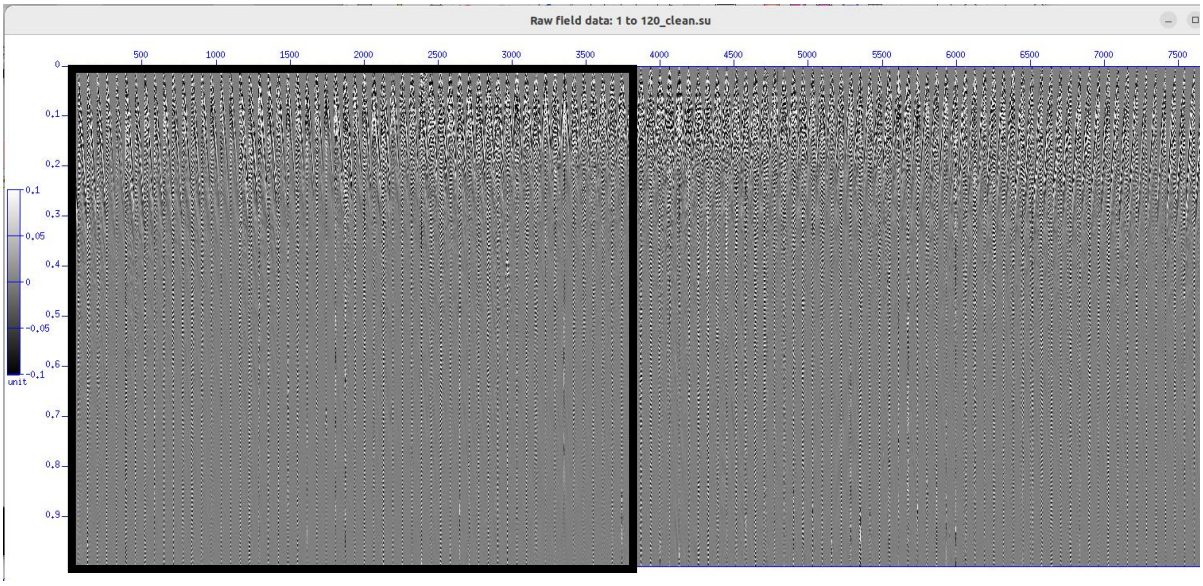


Figure 11. View of 120 shotpoint gathers from a single file (1_120_clean.su). For inset see Figure 12.

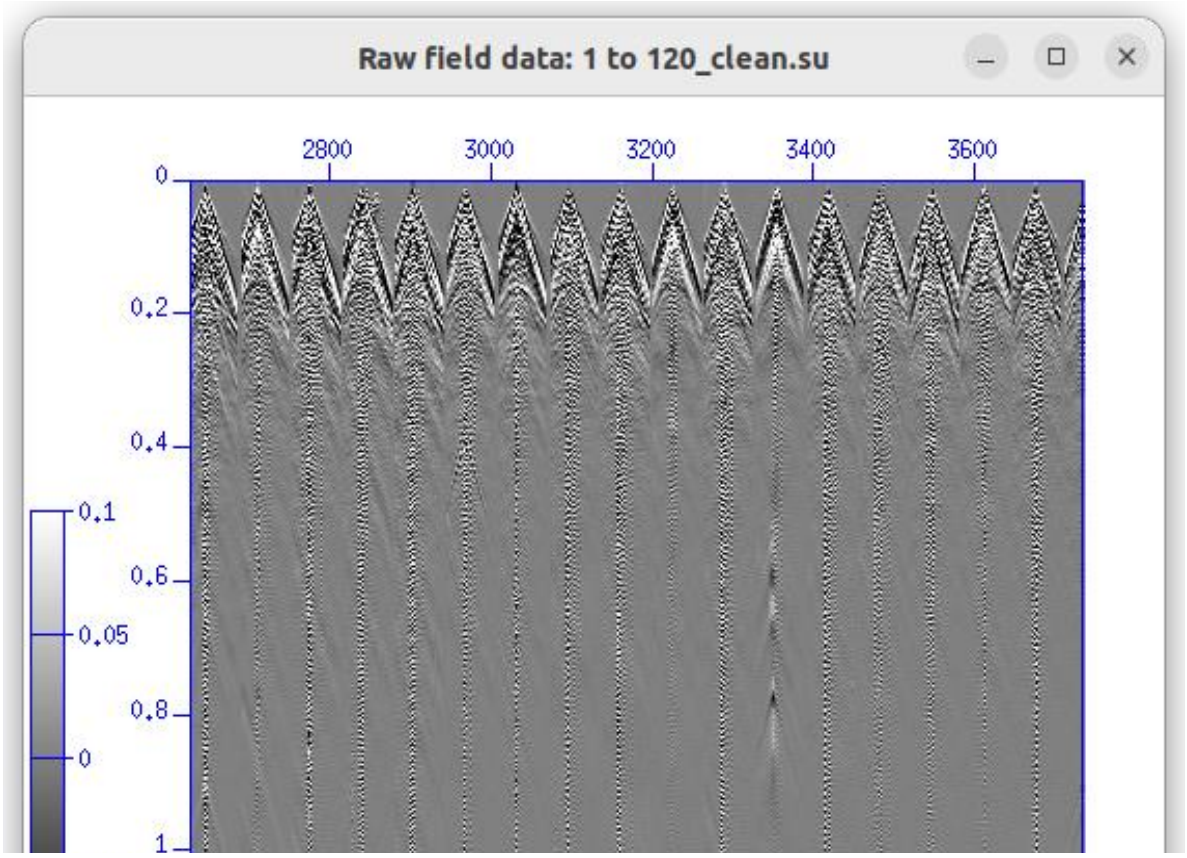


Figure 12. Close-up of field shot gathers, from Figure 11; traces: 2800 file to 3600.

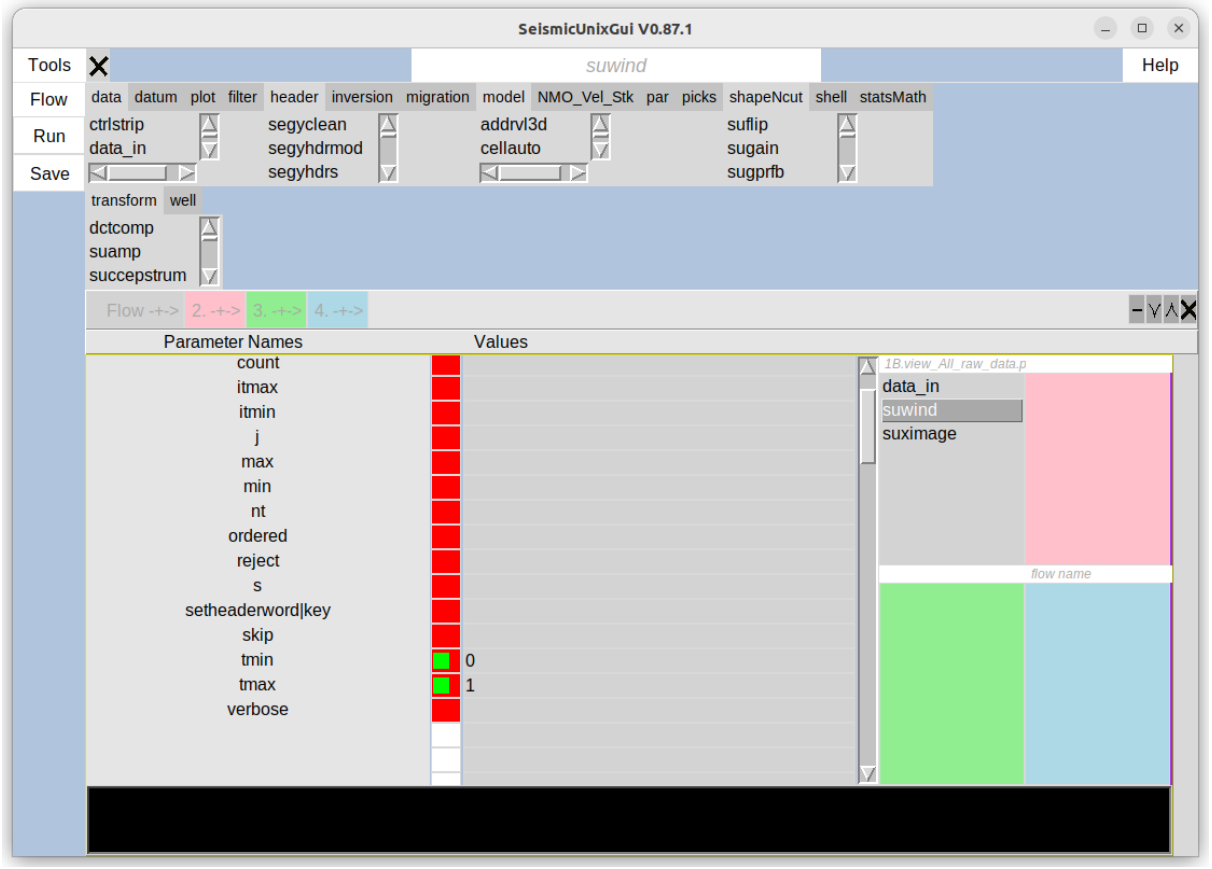


Figure 13. View of suwind parameters (second program in the flow) to select only the first second of data (tmin=0 and tmax=1).

Equivalent command-line instruction:

% perl

/home/**user**/Servilleta/seismics/pl/loma_blanca/053018/H/1/**user**/1B_view_All_raw_data.pl

SeismicUnixGui: Figure 12

3.1.3 STEP 3. Separation of shear-wave shot gathers by strike polarity

3.1.3.1 Concatenate files belonging to blows hit from NE

<MB 1> Tools->Sucat

Purpose: Concatenate a set of files

Input: Use a list of containing file names of shot gathers collected while striking the I-beam shear plate from the NE, i.e.

/home/**user**/Sevil-
leta/seimics/data/loma_blanca//053018/H/1/txt/**user**/**cat_hit_fromNE.txt**

Output:
/home/**user**/Sevilleta/seimics/data/loma_blanca//053018/H/1/su/**user**/hit_fromNE.su

Default configuration file:
/home/**user**/Servilleta/seismics/pl/loma_blanca/053018/H/1/**user**/**Sucat.config**

Equivalent command-line instruction, if file Sucat.config has been created:

% Sucat

Note: For convenience, a copy the Sucat.config file is saved as:
/home/**user**/Servilleta/seismics/pl/loma_blanca/053018/H/1/**user**/**Sucat.config_2**

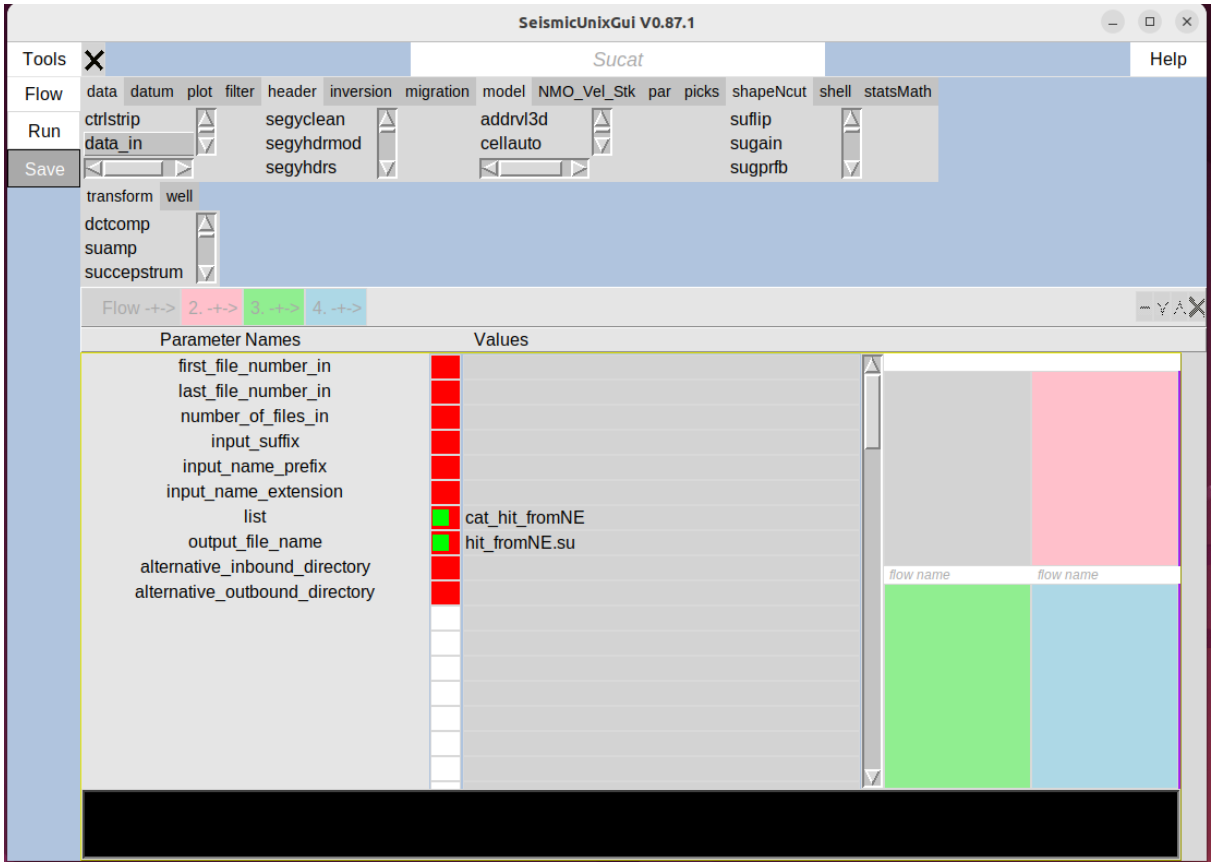


Figure 14. View of Sucat parameters for concatenating a set of files listed in file: cat_hit_fromNE.txt

.

3.1.3.2 View all concatenated data belonging to blows hit from NE

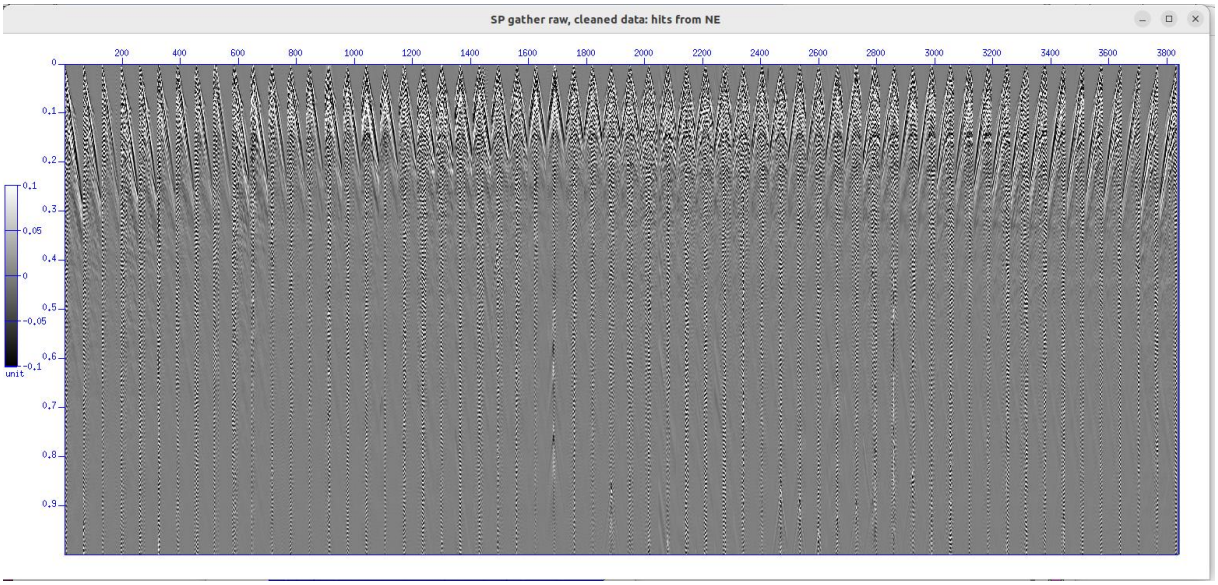


Figure 15 View of 60 shotpoint gathers from a single file (hit_fromNE.su.su).

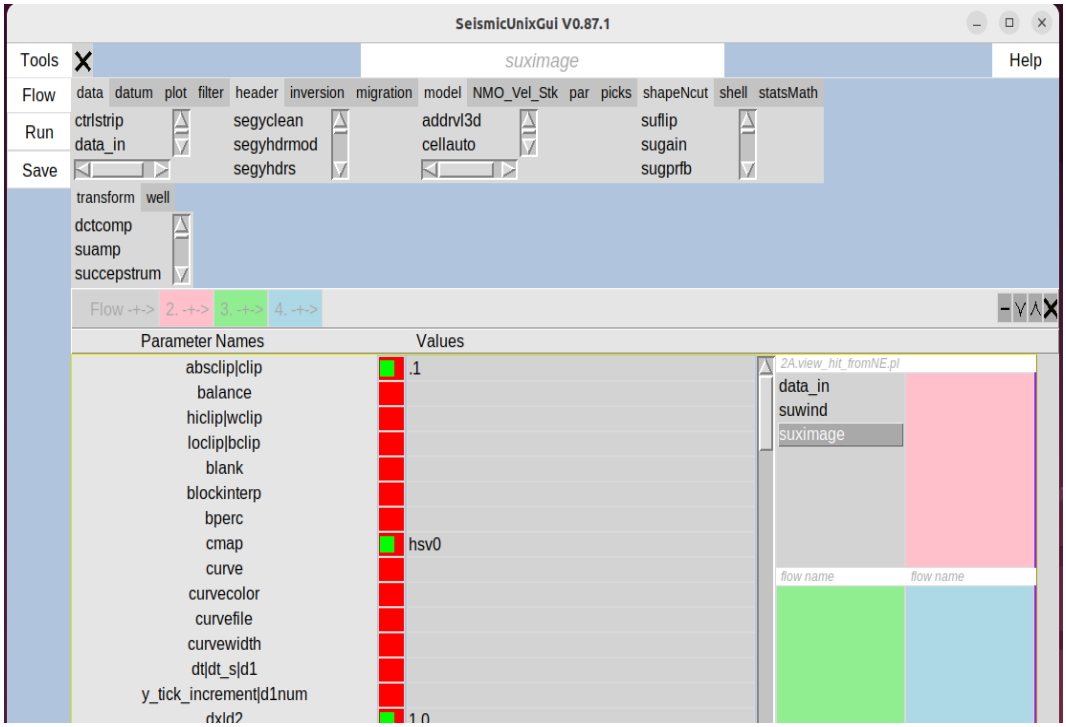


Figure 16 View of parameters, used in suximage, the third program in the flow of file: 2A.view_hit_fromNE.pl

<MB 1> Flow->Open

Select file: **2A.view_hit_fromNE.pl** (from GUI)
Purpose: View combined data set with the same strike polarity (from the NE)
Input: /home/**user**/Sevil-leta/seimics/data/loma_blanca//053018/H/1/su/**user**/**hit_fromNE.su**
Output: image (Figure 15)

3.1.3.3 View details of concatenated data belonging to blows from NE

<MB 1> Flow->Open

Select file: **2AA.view_hit_fromNE.pl** (from GUI)
Purpose: View close-up three SP gathers of previous combined data set
Input: /home/**user**/Sevil-leta/seimics/data/loma_blanca//053018/H/1/su/**user**/**hit_fromNE.su**
Output: image (Figure 17)

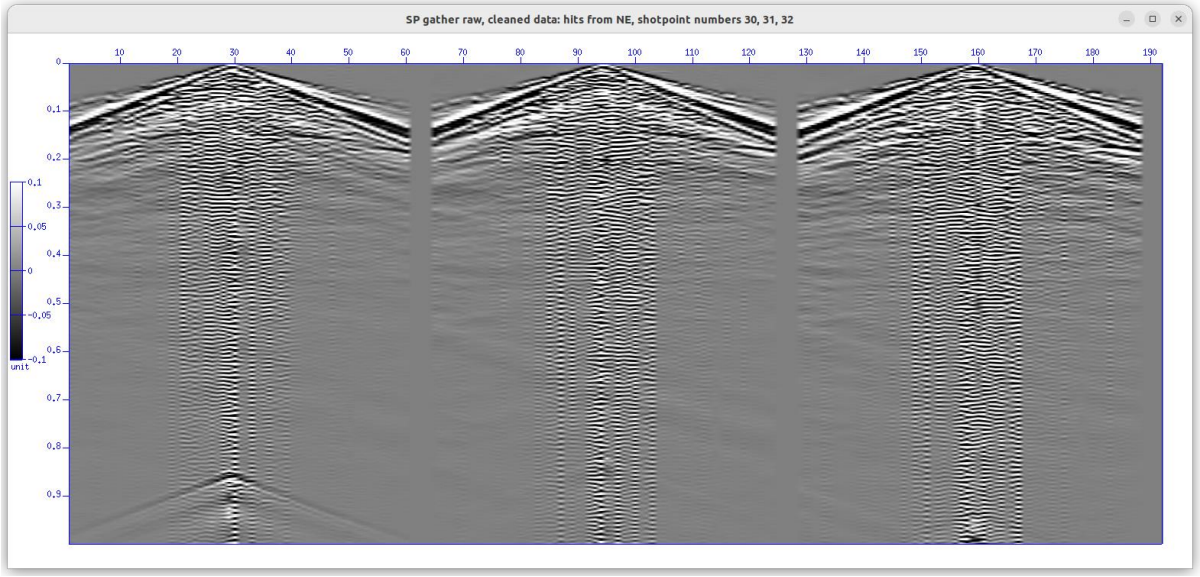


Figure 17. Raw SP gathers produced by hitting source plate from NE at SP 30, 31 and 32 where fldr= 59, 61 and 63. Trace numbers: 1857-1920 1921-1984 1985-2048

3.1.3.4 Concatenate files belonging to blows hit from SW

<MB 1> Tools->Sucat

Purpose: Concatenate a set of files

Input: Use a list of containing file names of shot gathers collected while striking the I-beam shear plate from the SW, i.e.,/home/**user**/Sevilleta/seimics/data/loma_blanca//053018/H/1/txt/**user/cat_hit_fromSW.txt**

Output:
/home/**user**/Sevilleta/seimics/data/loma_blanca//053018/H/1/su/**user/hit_fromSW.su**

Default configuration file:
/home/**user**/Servilleta/seismics/pl/loma_blanca/053018/H/1/**user/Sucat.config**

Equivalent command-line instruction, if file Sucat.config has been created:

```
% Sucat
```

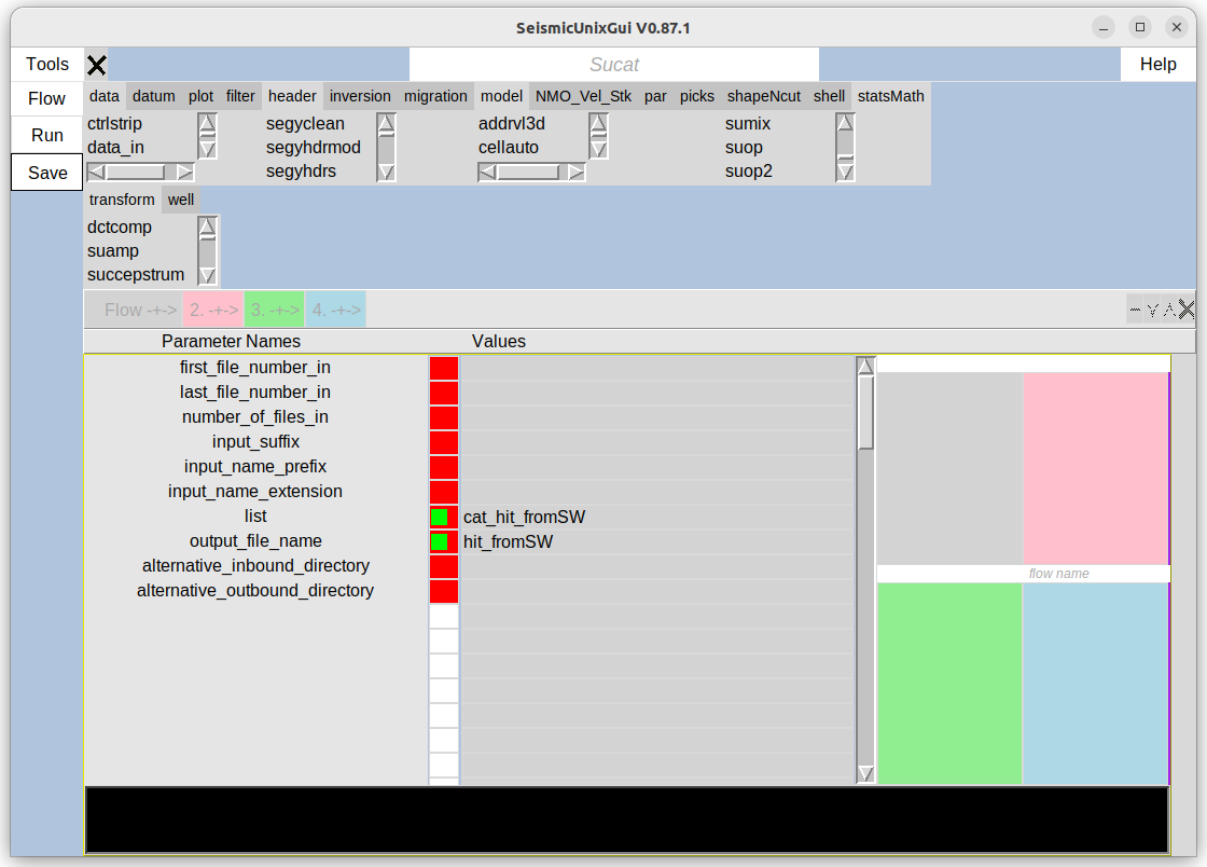


Figure 18. View of Sucat parameters for concatenating a set of files listed in file: cat_hit_fromSW.txt

Note: For convenience, a copy the Sucat.config file is saved as:
/home/**user**/Servilleta/seismics/pl/loma_blanca/053018/H/1/**user/Sucat.config_3**

3.1.3.5 View concatenated data

```
<MB 1> Flow->Open
```

Select file: **2B.view_hit_fromSW.pl** (from GUI)
Purpose: View combined data set with the same strike polarity (from the SW)
Input: /home/**user**/Sevil-leta/seimics/data/loma_blanca//053018/H/1/su/**user**/**hit_fromSW.su**
Output: image (Figure 19)

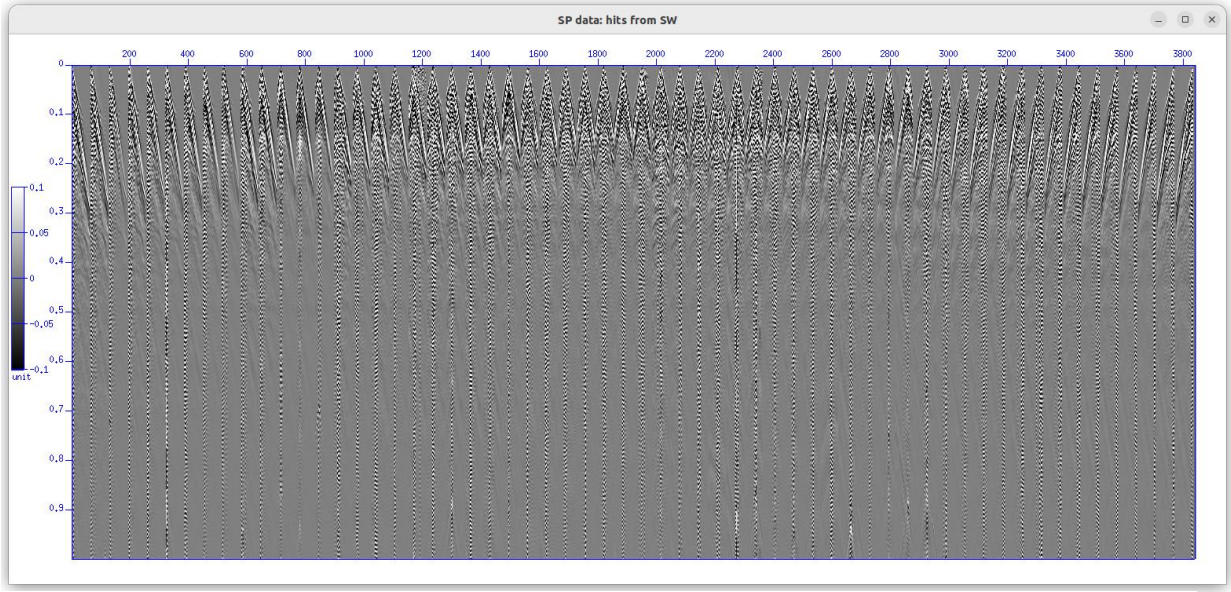


Figure 19. View of 60 shotpoint gathers from a single file (hit_fromSW.su).

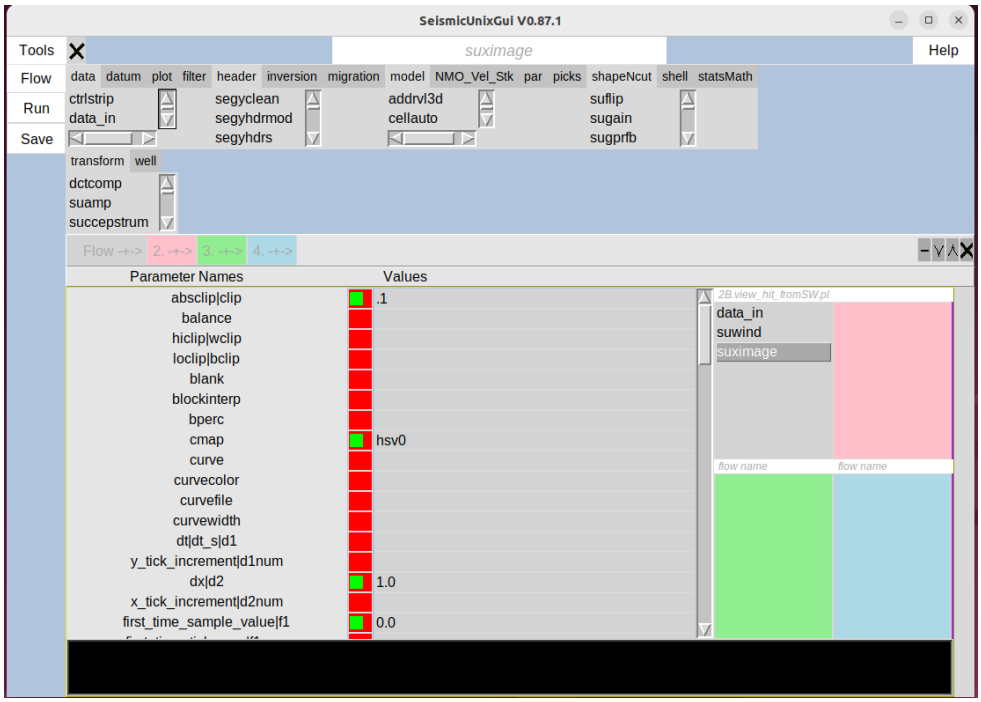


Figure 20. View of parameters, used in suximage, the third program in the flow of file: 2B.view_hit_fromSW.pl

3.1.3.6 View details of concatenated data

<MB 1> Flow->Open

Select file: **2C.view_hit_fromSW.pl** (from GUI)

Purpose: View close-up three SP gathers of previous combined data set

Input /home/**user**/Sevilleta/seimics/data/loma_blanca//053018/H/1/su/**user**/**hit_fromSW.su**

Output: image (Figure 21)

Equivalent command-line instruction:

% perl /home/**user**/Servilleta/seismics/pl/loma_blanca/053018/H/1/user/2C.view_hit_fromSW.pl

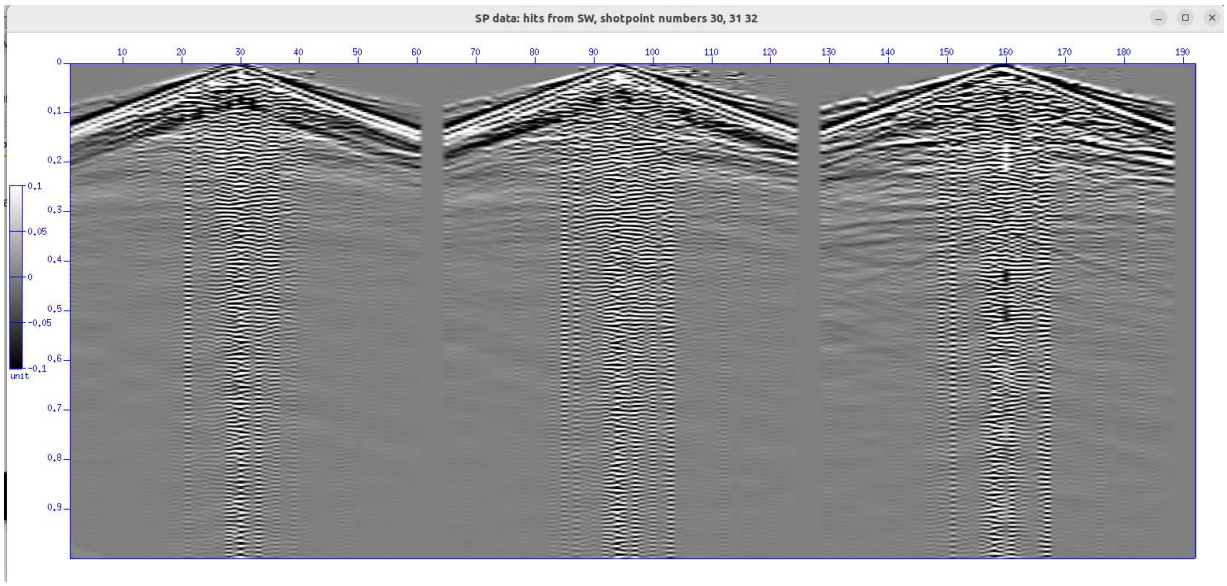


Figure 21. Raw SP gathers produced by hitting source plate from SW at SP 30, 31 and 32 where fldr= 60, 62 and 64. Trace numbers: 1857-1920, 1921-1984, 1985-2048

3.1.4 STEP 4. Subtraction of opposite-polarity stacks

<MB 1> Flow->Open

Select file: **3A.sudiff.pl** (from GUI)

Purpose: Enhance SH wave arrivals and remove potential S-to-P converted modes

Input: /home/**user**/Sevil-
leta/seimics/data/loma_blanca//053018/H/1/su/**user**/hit_fromNE.su and hit_fromSW.su

Output:/home/**user**/Sevilleta/seimics/data/loma_blanca//053018/H/1/su/**user**/All_SH.su

Equivalent command-line instruction:

% perl /home/**user**/Servilleta/seismics/pl/loma_blanca/053018/H/1/**user**/**3A.sudiff.pl**

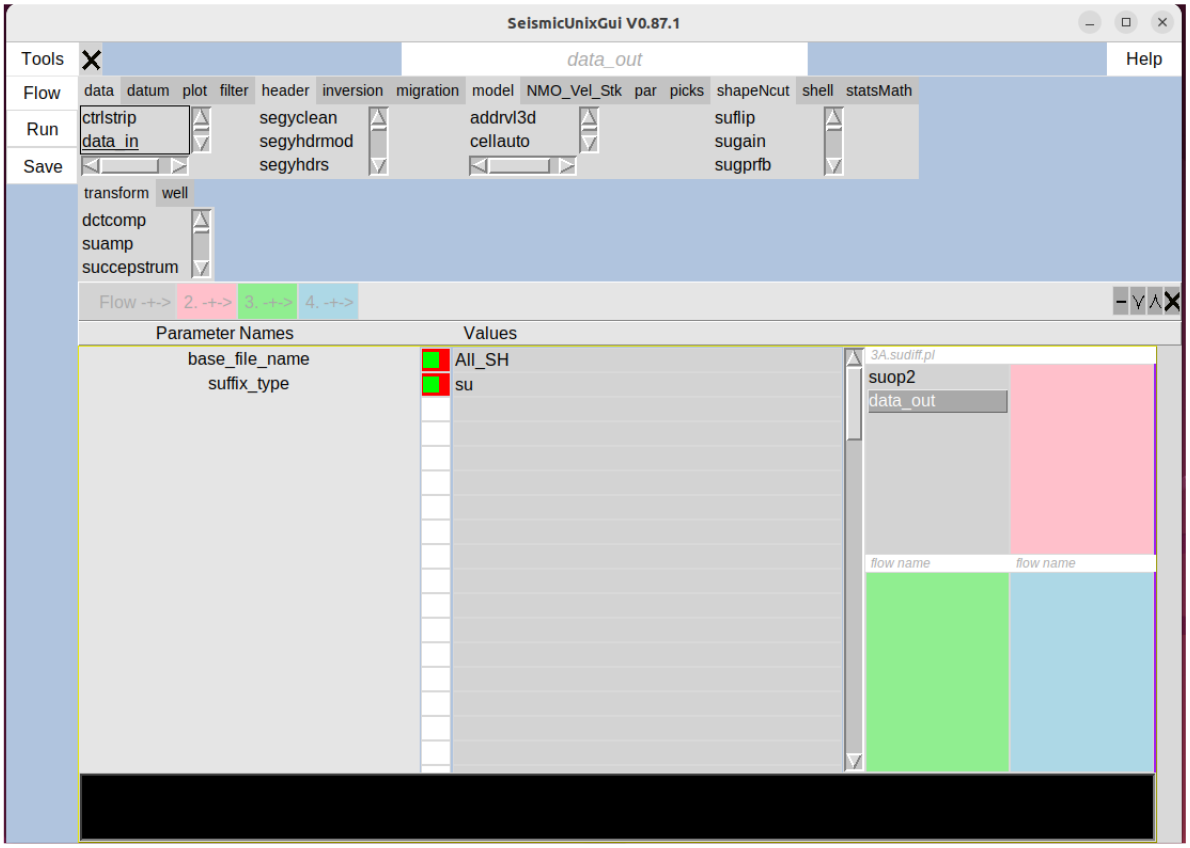


Figure 22. Output from subtracting stacks is sent to file All_SH.su (file-type =“su”).

3.1.4.1.1 View and confirm quality of concatenated data

<MB 1> Flow->Open

Select file: 3B.view_All_SH.pl (from GUI)

Purpose: View close-up three SP gathers of previous combined data set

Input /home/**user**/Sevilleta/seimics/data/loma_blanca//053018/H/1/su/**user**/**All_SH.su**

Output: image (Figure 23)

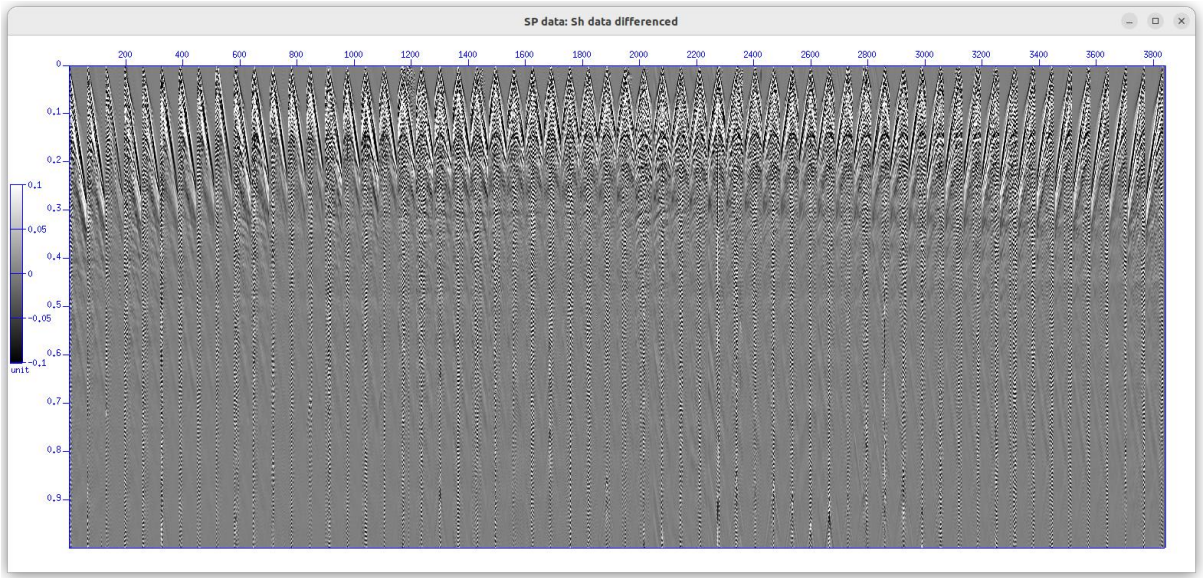


Figure 23. SP gathers after differencing gathers with opposing data polarities

Equivalent command-line instruction:

% perl /home/**user**/Servilleta/seismics/pl/loma_blanca/053018/H/1/**user**/3B.view_All_SH.pl

3.1.5 STEP 5. Change of header geometries to prepare the data for additional processing

<MB 1> Flow->Open

Select file: **4A.sx_gx_header.pl** (from GUI)

Purpose: Build the following header values: sx (shotpoint x-value location) and gx (geophone x-value location

Input /home/**user**/Sevilleta/seimics/data/loma_blanca//053018/H/1/su/**user**/**All_SH.su**

Output:/home/**user**/Sevilleta/seimics/data/loma_blanca//053018/H/1/su/**user**/**All_SH_geom1.su**

Equivalent command-line instruction:

% perl /home/**user**/Servilleta/seismics/pl/loma_blanca/053018/H/1/**user**/**4A.sx_gx_header.pl**

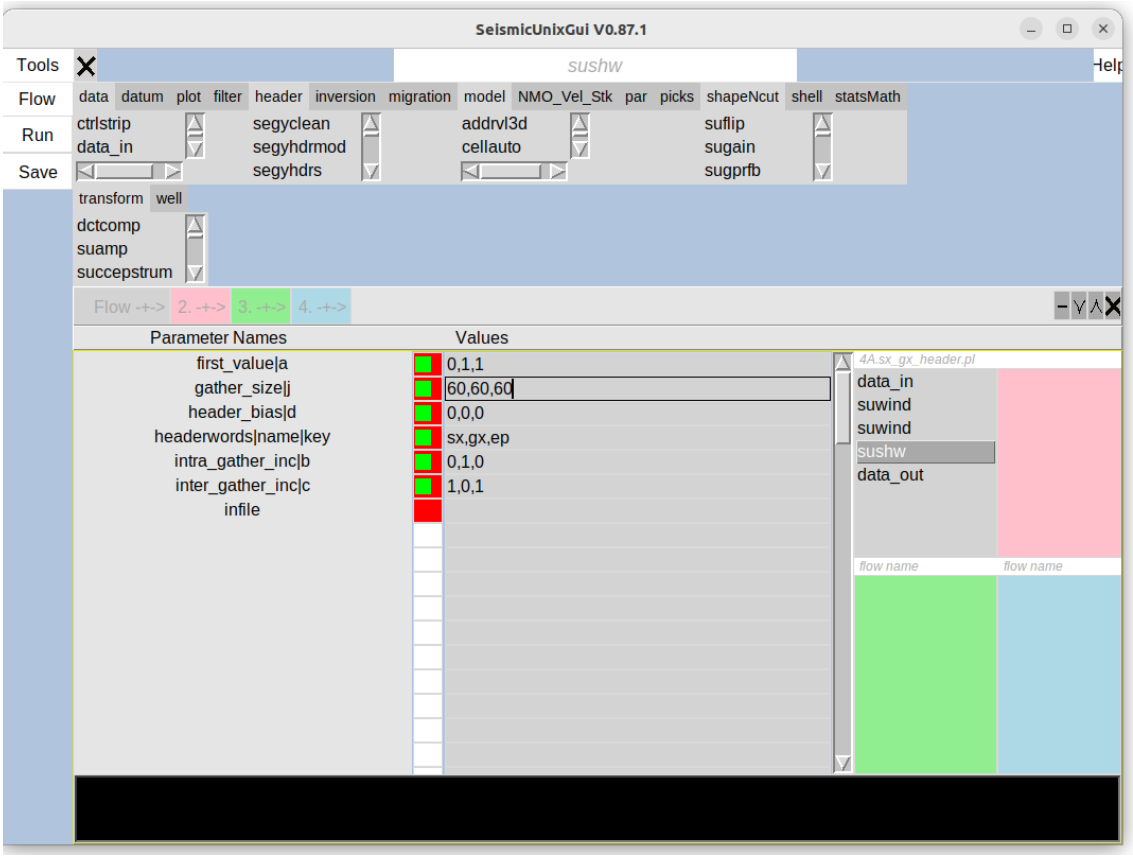


Figure 24. Parameter values for sushw to build and load geometry values into the header

3.1.6 STEP 6. Modification of Header files--offsets

<MB 1> Flow->Open

Select file: **4B.set_offset.pl** (from GUI)

Purpose:
Calculate offsets in the header:
offfset = shotpoint poisition (sx) - geophone position (gx). We use the convention that geophones to the right of the shotpoint are negative and geophones located to the left of the shotpoint have positive offset values.

Input /home/**user**/Sevilleta/seimics/data/loma_blanca//053018/H/1/su/**user**/**All_SH_geom1.su**

Output:/home/**user**/Sevil-leta/seimics/data/loma_blanca//053018/H/1/su/**user**/**All_SH_geom2.su**

Equivalent command-line instruction:

% perl /home/**user**/Servilleta/seismics/pl/loma_blanca/053018/H/1/**user**/**4B.set_offset.pl**

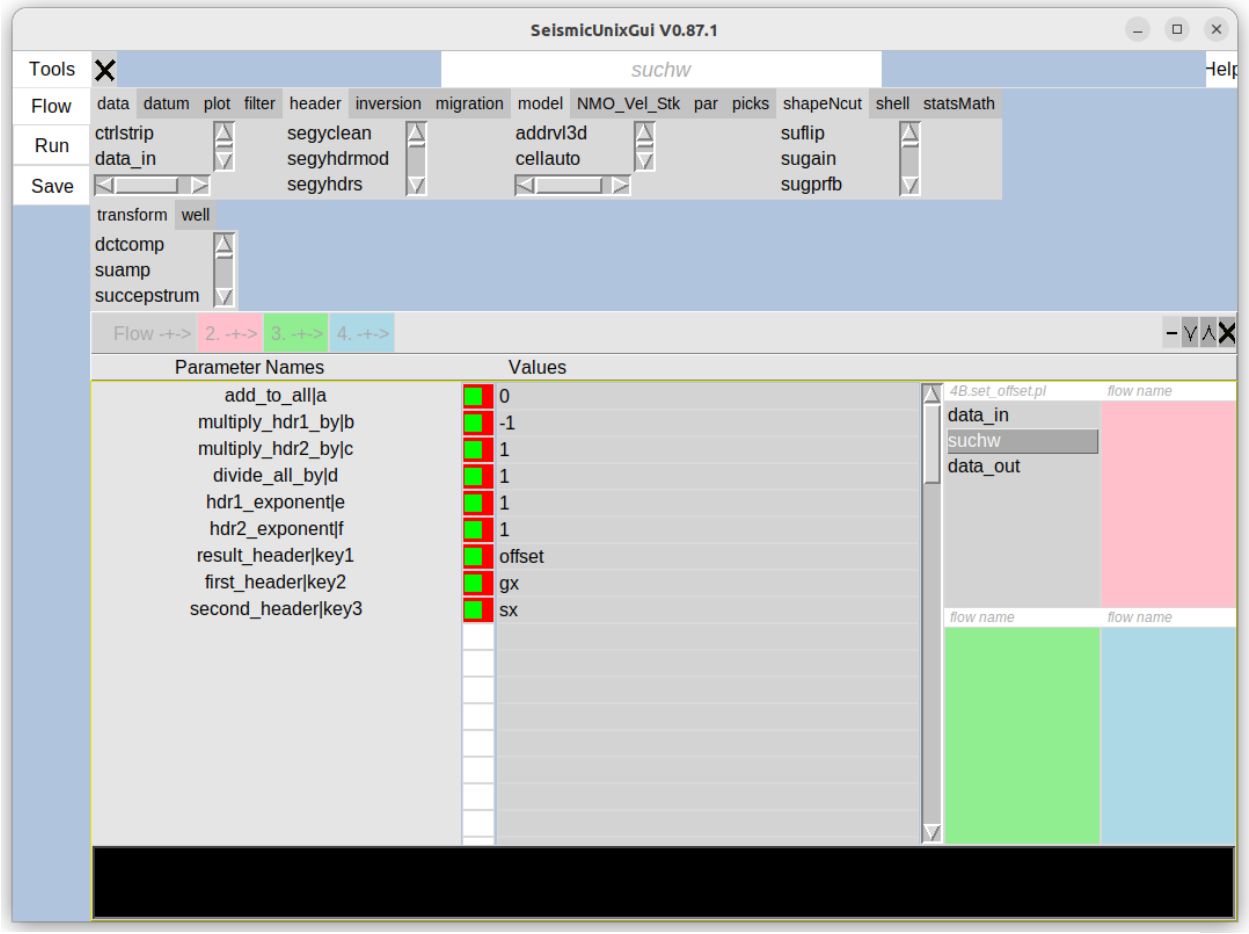


Figure 25. Parameter values for offset, gx, and sx, used to calculate offsets in the headers

3.1.6.1 Numerical verification of offset calculations

Confirm the result by examining the new values for the offset:

% cd /home/**user**/Servilleta/seismics/data/loma_blanca//053018/H/1/su/**user**

And then when you are in the correct data directory:

% **suxedit** All_SH_geom2.su

Automatically, user is placed at the end of the file, at trace number = 3600, and see the following:

tracr=60 fldr=119 tracf=60 ep=60 trid=1 offset=-1
sx=59 gx=60 ns=1001 dt=1000 year=2018 day=151
hour=18 minute=32 sec=1

3.1.6.2 Graphical verification of offset calculations

<MB 1> Flow->Open

Select file: **4C.view_header_geom.pl** (from GUI)

Purpose:
Plot the new header parameters

Input /home/**user**/Sevilleta/seimics/data/loma_blanca//053018/H/1/su/**user**/**All_SH_geom2.su**

Output: Plot in window (Figure 26)

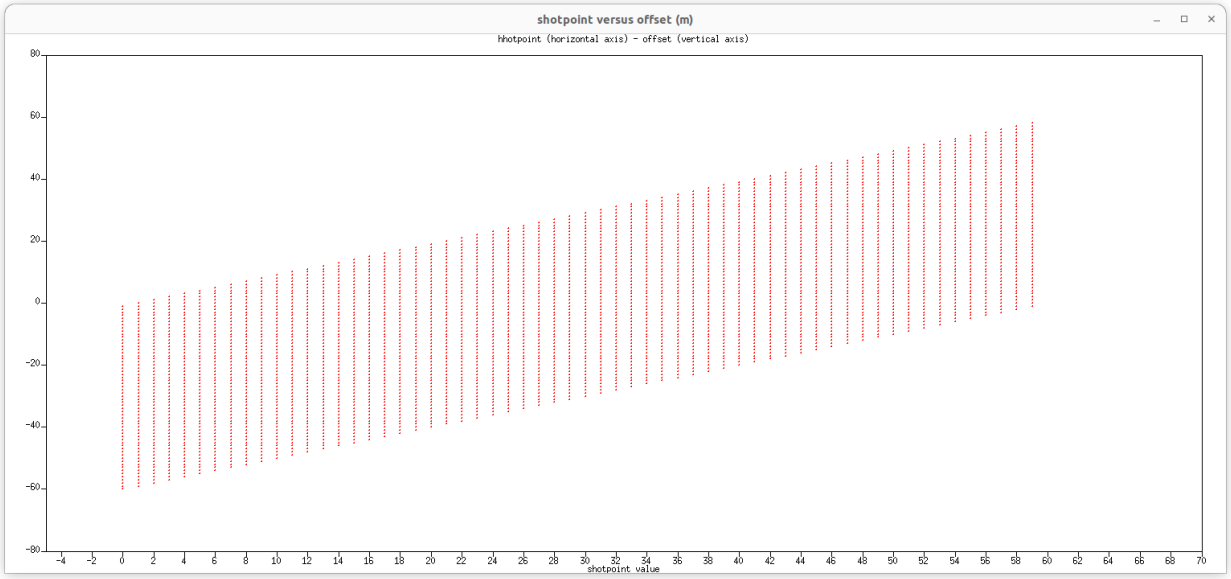


Figure 26. Plotted header values of sx (horizontal axis) versus offset display (vertical axis) a geometric pattern that reflects the regular acquisition geometry used in the experiment. There is one red filled circle for each trace in the data set (3600 points). Our convention results in negative offset values represent geophones that lie north of the shotpoint location (sx).

Equivalent command-line instruction:

% perl /home/**user**/Servilleta/seismics/pl/loma_blanca/053018/H/1/**user/4C.view_header_geom.pl**

3.1.7 STEP 7. Creation of CDP's/CMP's

<MB 1> Flow->Open

Select file: **5A.make_cmp.pl** (from GUI)

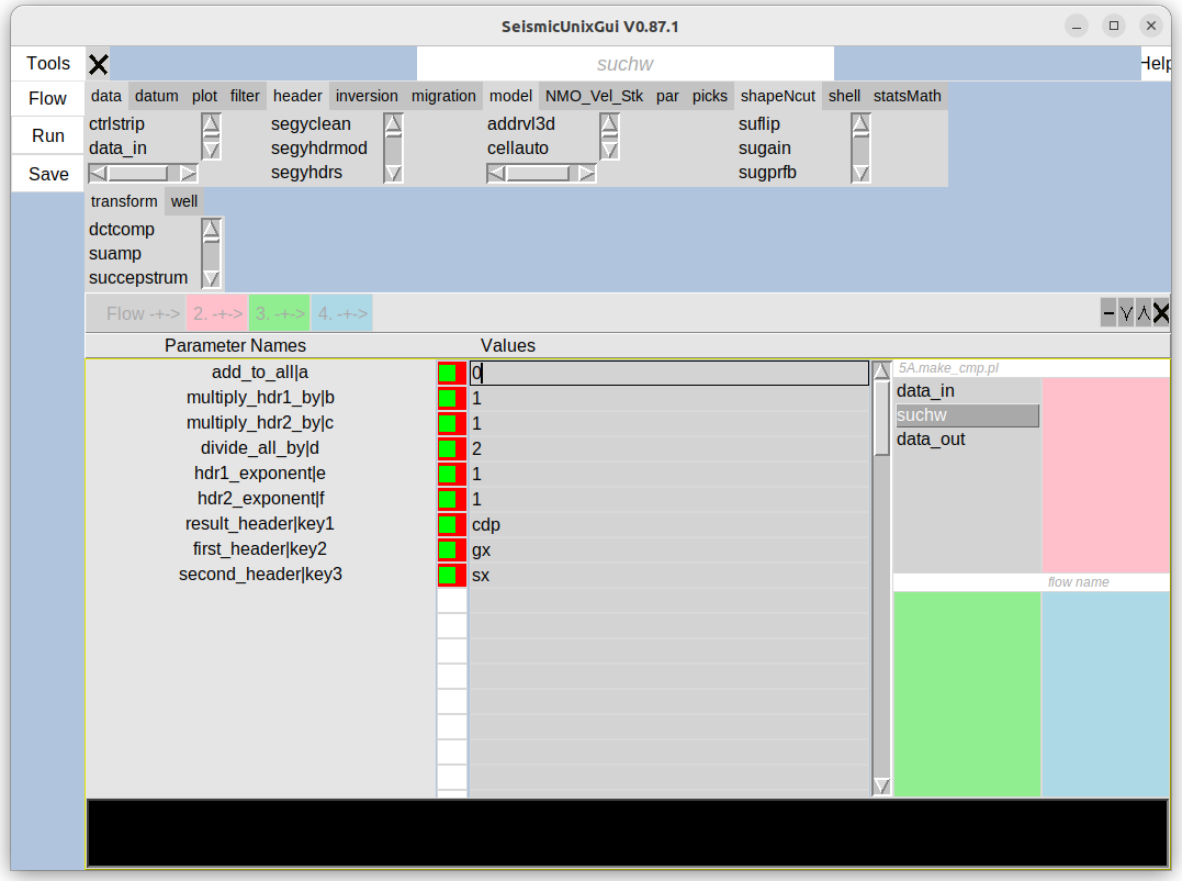


Figure 27. Values of arithmetic operators (6 first lines) used to calculate cdp values from header parameters gx (geophone location) and sx (shotpoint location). Results are saved into the headers of an output file (All_SH_geom3.su).

Purpose: Calculate cmp/cdp values and place them in the corresponding data header ("cdp")

Input: /home/**user**/Servilleta/seimics/data/loma_blanca//053018/H/1/su/**user/All_SH_geom2.su**

Output: /home/**user**/Sevil-leta/seimics/data/loma_blanca//053018/H/1/su/**user**/**All_SH_geom3.su**

Equivalent command-line instruction:

% perl /home/**user**/Servilleta/seismics/pl/loma_blanca/053018/H/1/**user**/**5A.make_cmp.pl**

3.1.7.1 Numerical verification of cmp/cdp calculations

Confirm the result by examining the new values for the "cdp" header parameter:

% cd /home/**user**/Servilleta/seismics/data/loma_blanca//053018/H/1/su/**user**/

And then when you are in the correct data directory:

% **suxedit** All_SH_geom3.su

Automatically, you will be placed at the end of the file, at trace number = 3600, and see the following:

tracr=60 fldr=119 tracf=60 ep=60 cdp=60 trid=1
offset=-1 sx=59 gx=60 ns=1001 dt=1000 year=2018
day=151 hour=18 minute=32 sec=1

3.1.7.2 Graphical verification of cmp/cdp calculations

<MB 1> Flow->Open

Select file: **5B.view_cmp_hdr.pl** (from GUI)

Purpose:

Display plot of the newly estimated header parameter "cdp" versus offset.

Input /home/**user**/Sevil-leta/seimics/data/loma_blanca//053018/H/1/su/**user**/**All_SH_geom3.su**

Output: Plot in window (Figure 28)

Equivalent command-line instruction:

% perl /home/**user**/Servilleta/seismics/pl/loma_blanca/053018/H/1/**user**/**5B.view_cmp_hdr.pl**

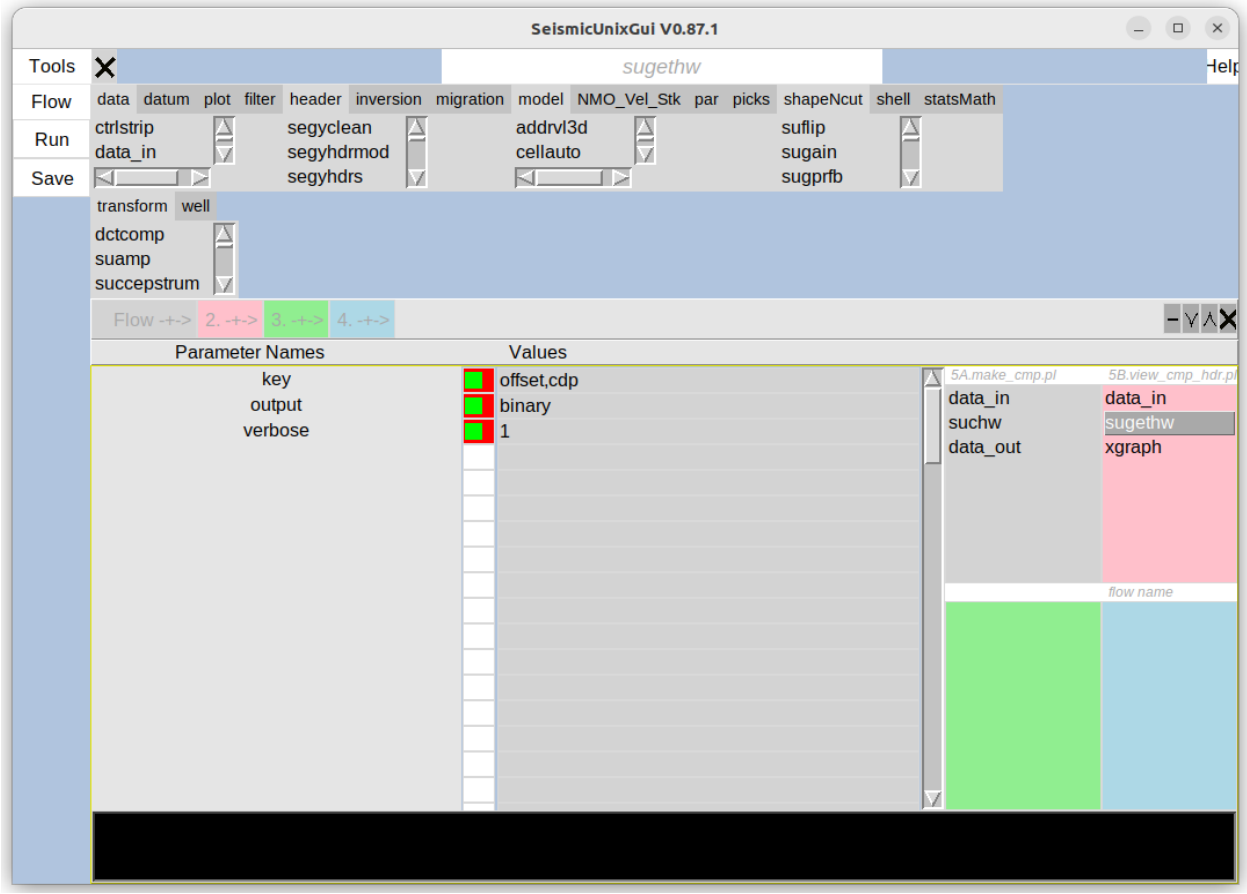


Figure 28. Values used by sugethw to help display offset and cdp (Figure X).

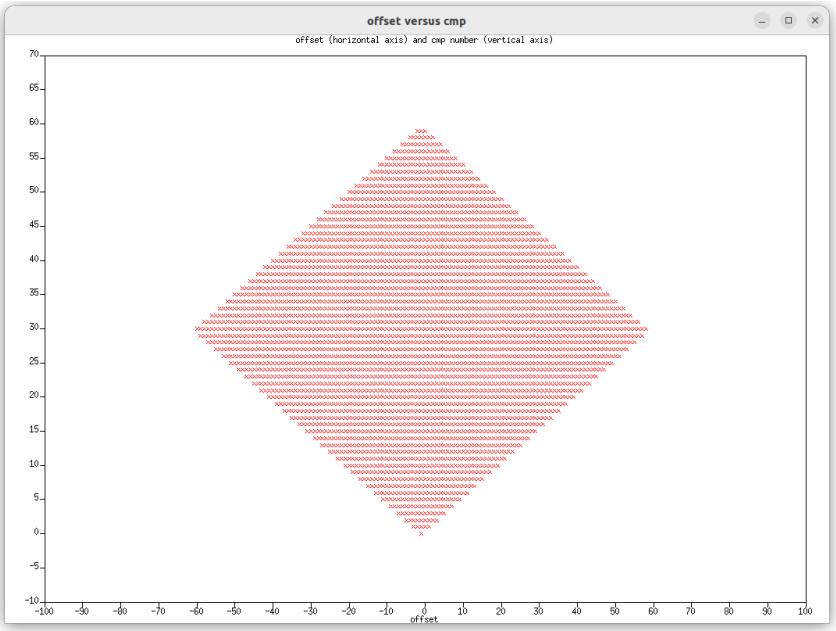


Figure 29. Plotted header values of offset (horizontal axis) versus cmp/cdp display (vertical axis) a geometric pattern that reflects the regular acquisition geometry used in the experiment. There is one red "x" for each calculated "cdp" value in the data set (3600 points). Our convention results in negative offset values represent geophones that lie north of the shotpoint locations.

3.1.8 STEP 8. Selection of top-mute values to suppress Love wave noise.

3.1.8.1 Interactive picking of time-distance pairs for top muting

<MB 1> Tools-> iTopMute

Purpose: To move through between the 7th and 60th shotpoint gather (segy header key-word="ep") interactively picking vertices of a polygon (x dimension, and time dimension). These value pairs are saved as text files and serve to identify zones to be zeroed out in the sub-sequent step.

Input /home/**user**/Sevil-leta/seimics/data/loma_blanca//053018/H/1/su/**user**/**All_SH_geom3.su**

Output: Saved files containing x-t pairs in folder: /home/**user**/Sevil-leta/seimics/data/loma_blanca//053018/H/1/txt/**user**/

Default configuration file: /home/**user**/Servilleta/seismics/pl/loma_blanca/053018/H/1/**user/iTopMute.config**

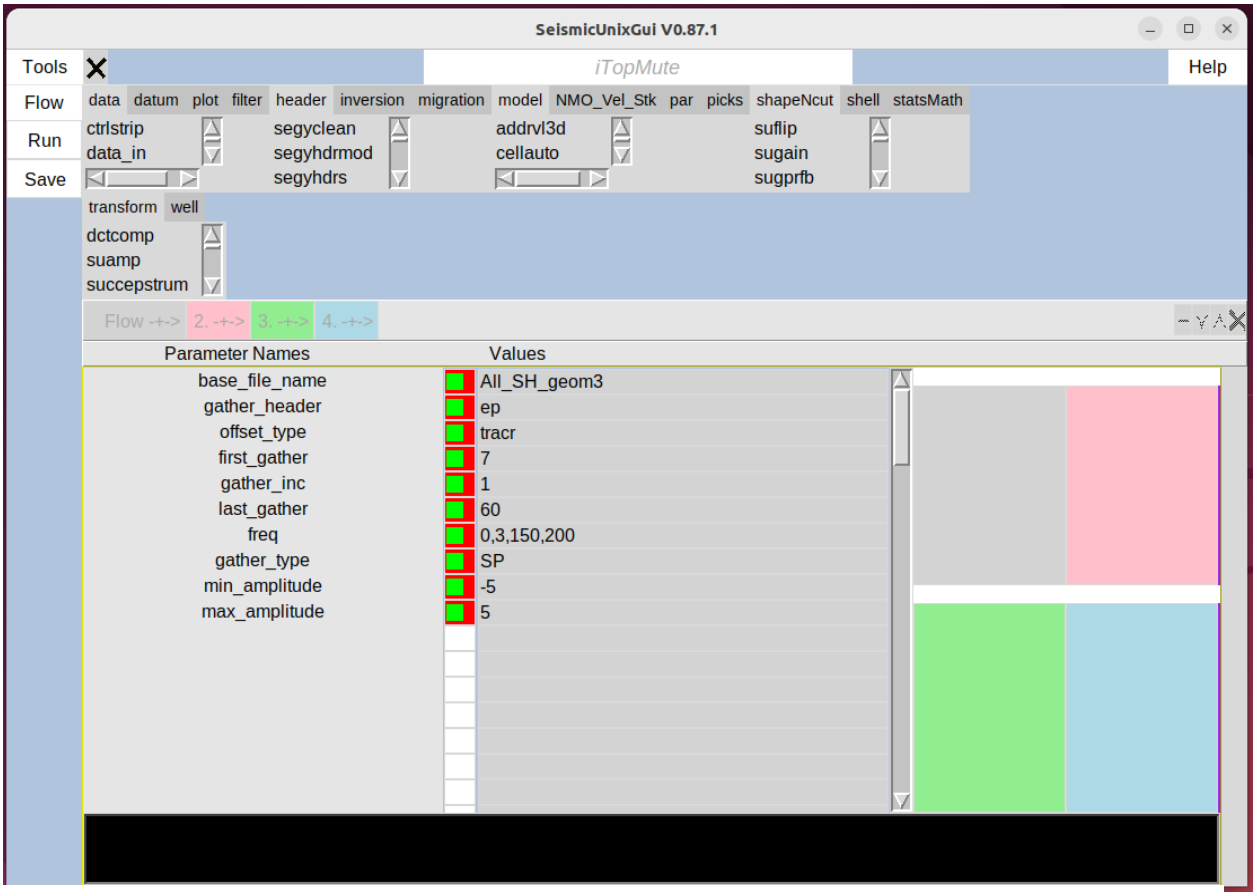


Figure 30. Display of parameters used by iTopMute. Remember to click: Save (<MB1>) and Run (<MB1>)

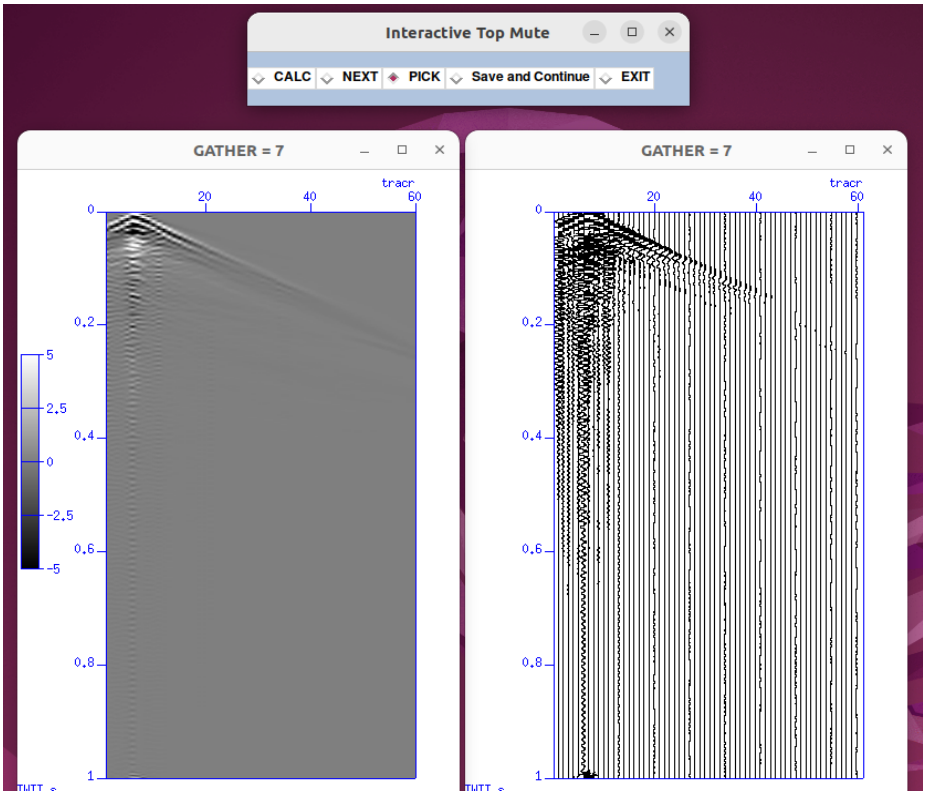


Figure 31 Panels display two versions of the same shotpoint gather; number 7. Top menu selections allow review ("Calc") and change ("pick") of mute pairs before proceeding to the next shotpoint gather. Instructions for picking appear separately in the command window.

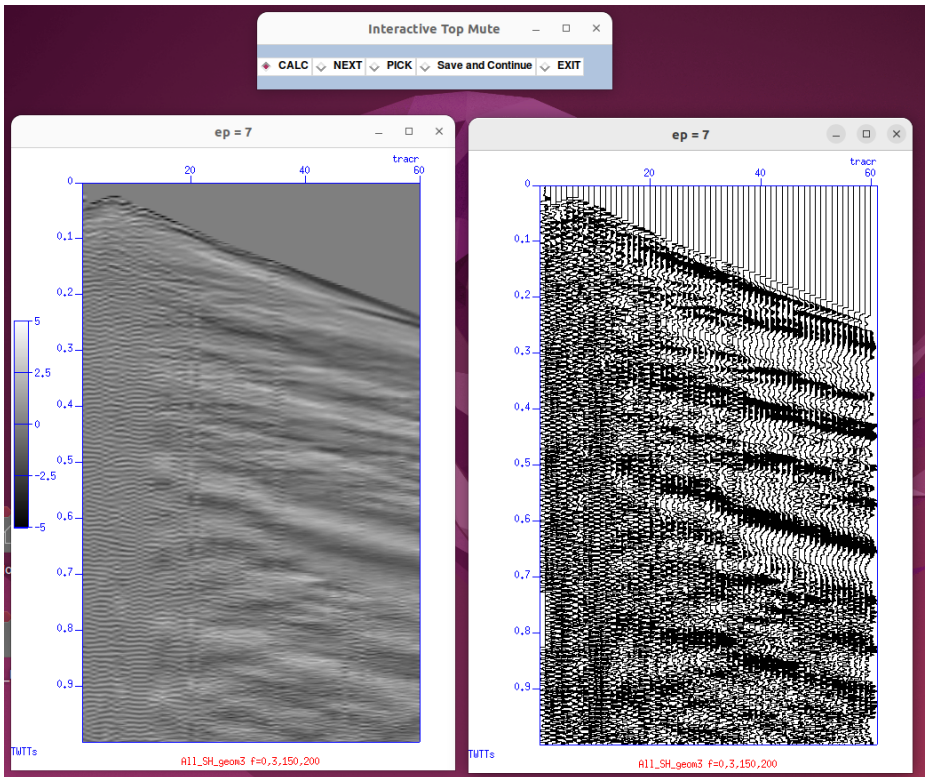


Figure 32 Panels display two versions of the same shotpoint gather, number 7 after application of the mute (not that the upper triangular areas have much lower noise and that earlier portions of the Love wave train has been removed.

Equivalent command-line instruction, if file **iTopMute**.config has been created:

% **iTopMute**

3.1.8.2 Concatenation of text files containing top-mute picked pairs

<MB 1> Tools->Sucat

Purpose: Concatenate and reformat files containing top-mute picks from individual shot gathers, selected in prior step into a single file .

Input: Uses a list of containing file names of top-mute pair picks from individual shot gathers, i.e., /home/**user**/Sevilleta/seimics/data/loma_blanca//053018/H/1/txt/**user/itop_mute_list.txt**

Output: Single, collated text file containing picked pairs from all shotgathers, i.e., /home/**user**/Sevilleta/seimics/data/loma_blanca//053018/H/1/su/**user/itop_mute_all.txt**

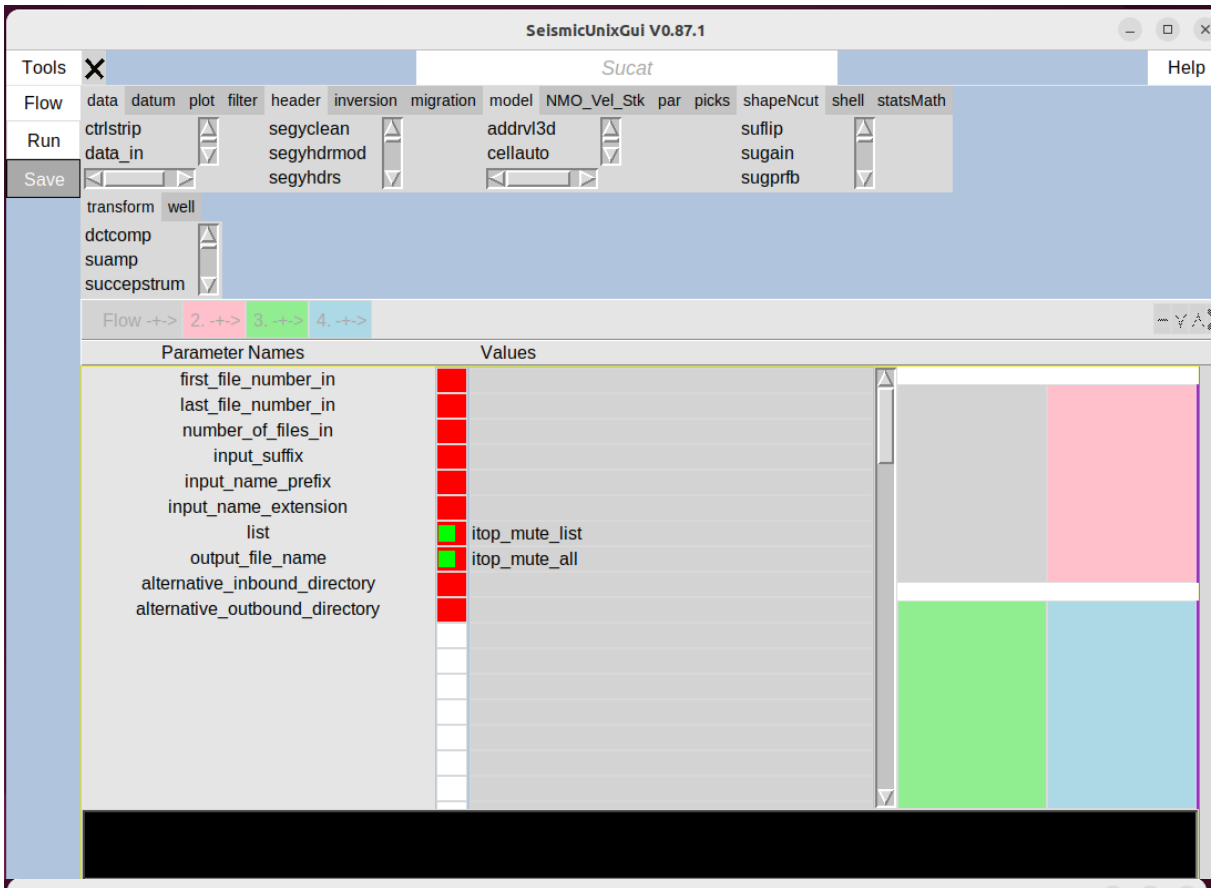


Figure 33. Display of parameters used by Sucat to concatenate individual files generated during interactive picking of itop-mute pairs. Remember to click: Save (<MB1>) and Run (<MB1>)

Default configuration file:
/home/**user**/Servilleta/seismics/pl/loma_blanca/053018/H/1/**user**/**Sucat.config**

Equivalent command-line instruction, if file Sucat.config has been created:
% Sucat
Note: For convenience, a copy of the Sucat.config file created in this step is saved as /home/**user**/Servilleta/seismics/pl/loma_blanca/053018/H/1/**user**/**Sucat.config_4**

3.1.8.3 Application of top-mute picked pairs

<MB 1> Flow->Open

Select file: **5C.sumute.pl** (from GUI)

Purpose: Remove Love wave train by applying picked top-mute pairs.

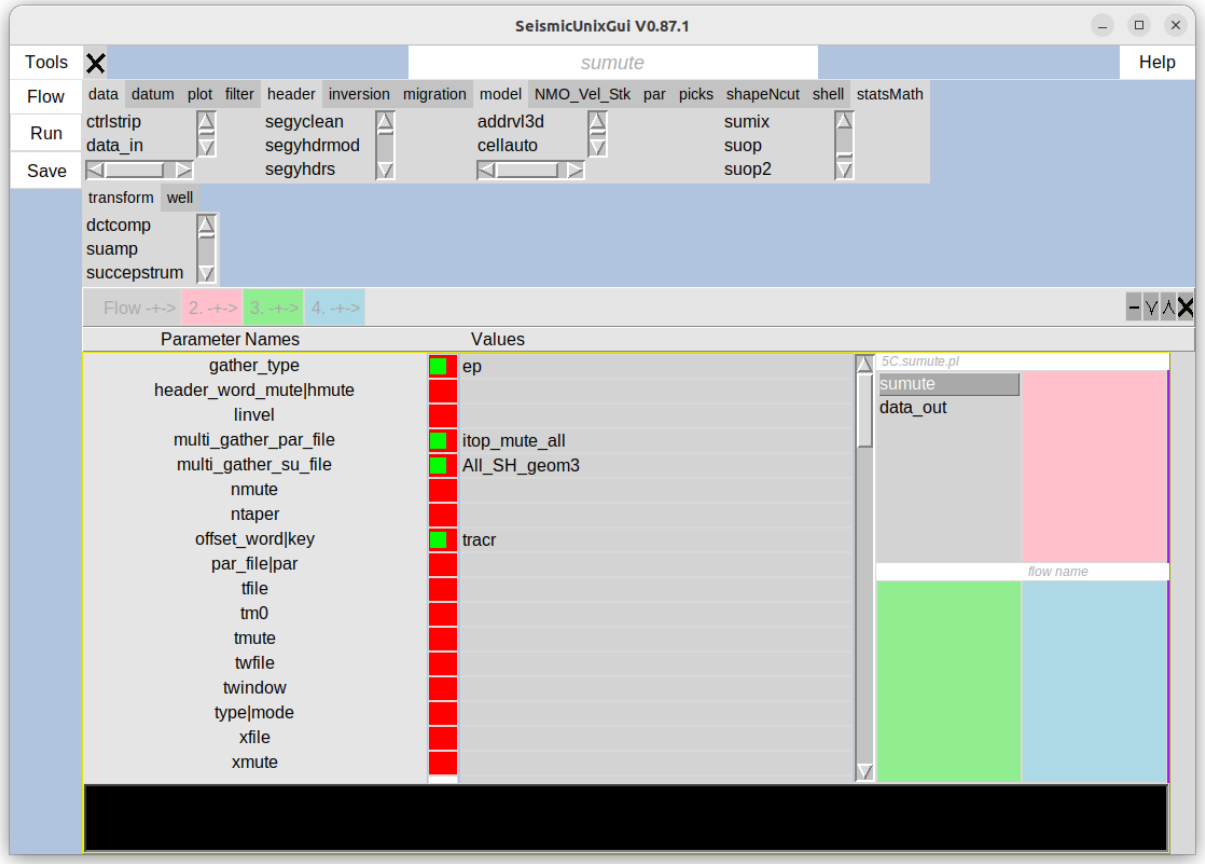


Figure 34. Display of parameters used by sumute to apply interactively picked top-mute, x-time pairs.

Input /home/**user**/Sevil-leta/seimics/data/loma_blanca//053018/H/1/txt/**user**/**itop_mute_all.txt** (applied to /home/**user**/Sevilleta/seimics/data/loma_blanca//053018/H/1/su/**user**/**All_SH_geom3.su**)

Output: /home/**user**/Sevil-leta/seimics/data/loma_blanca//053018/H/1/su/**All_SH_geom3_mute.su**

Note: It is important to use the same keyword (e.g., “tracr”) as was used during the picking of muting pairs.in Tools-> iTopMute.

Equivalent command-line instruction:

% perl /home/**user**/Servilleta/seismics/pl/loma_blanca/053018/H/1/**user**/**5C.sumute.pl**

3.1.8.4 View sub-portion of “top-muted”data

<MB 1> Flow->Open

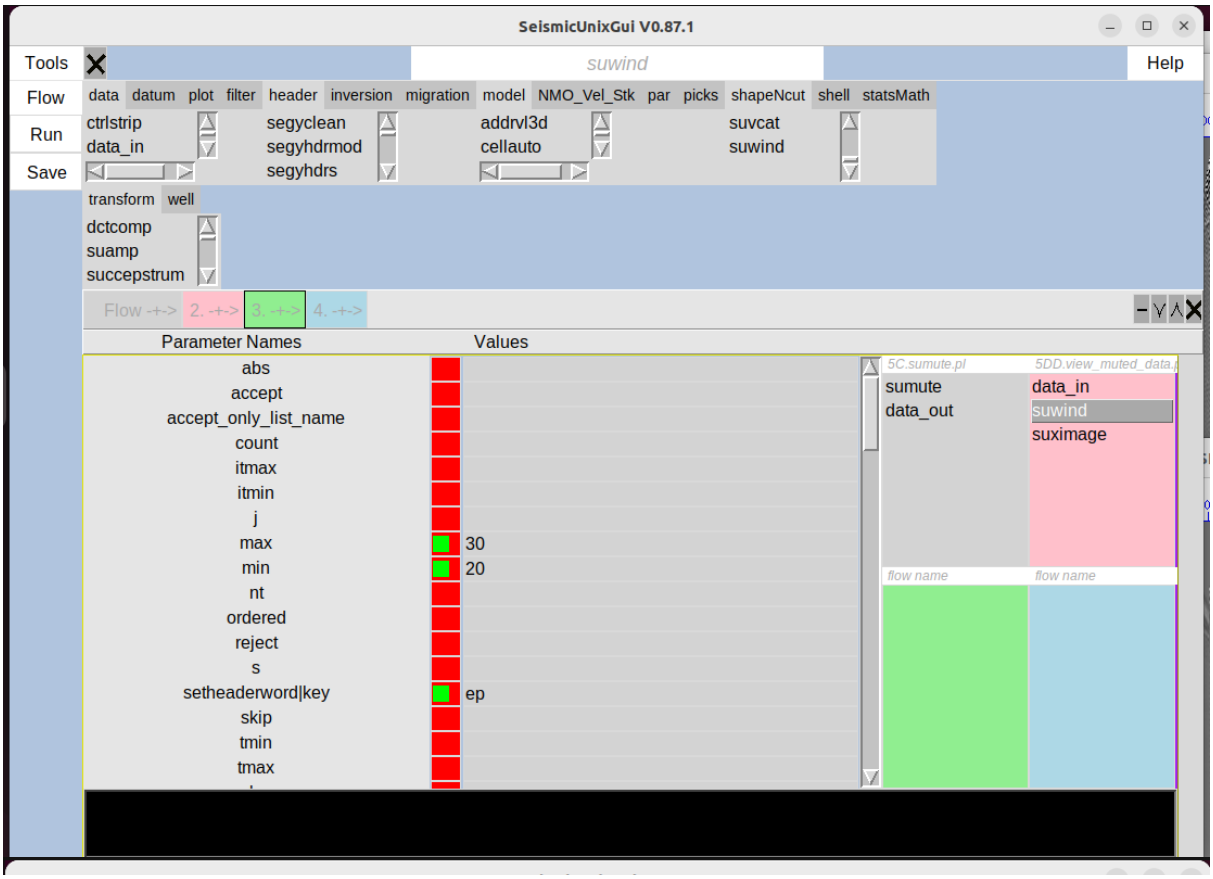


Figure 35. Display of parameters used to display a subsample of the complete data set, i.e. only shotpoint gathers (ëp”) numbers 20 through 30 inclusively are selected.

Select file: **5DD.view_muted_data.pl** (from GUI)

Purpose: Graphical display of a subset of previous data with its upper sections muted.

Input: /home/**user**/Sevil-
leta/seimics/data/loma_blanca//053018/H/1/su/**user**/**All_SH_geom3_mute.su**

Output: Image (Figure 36)

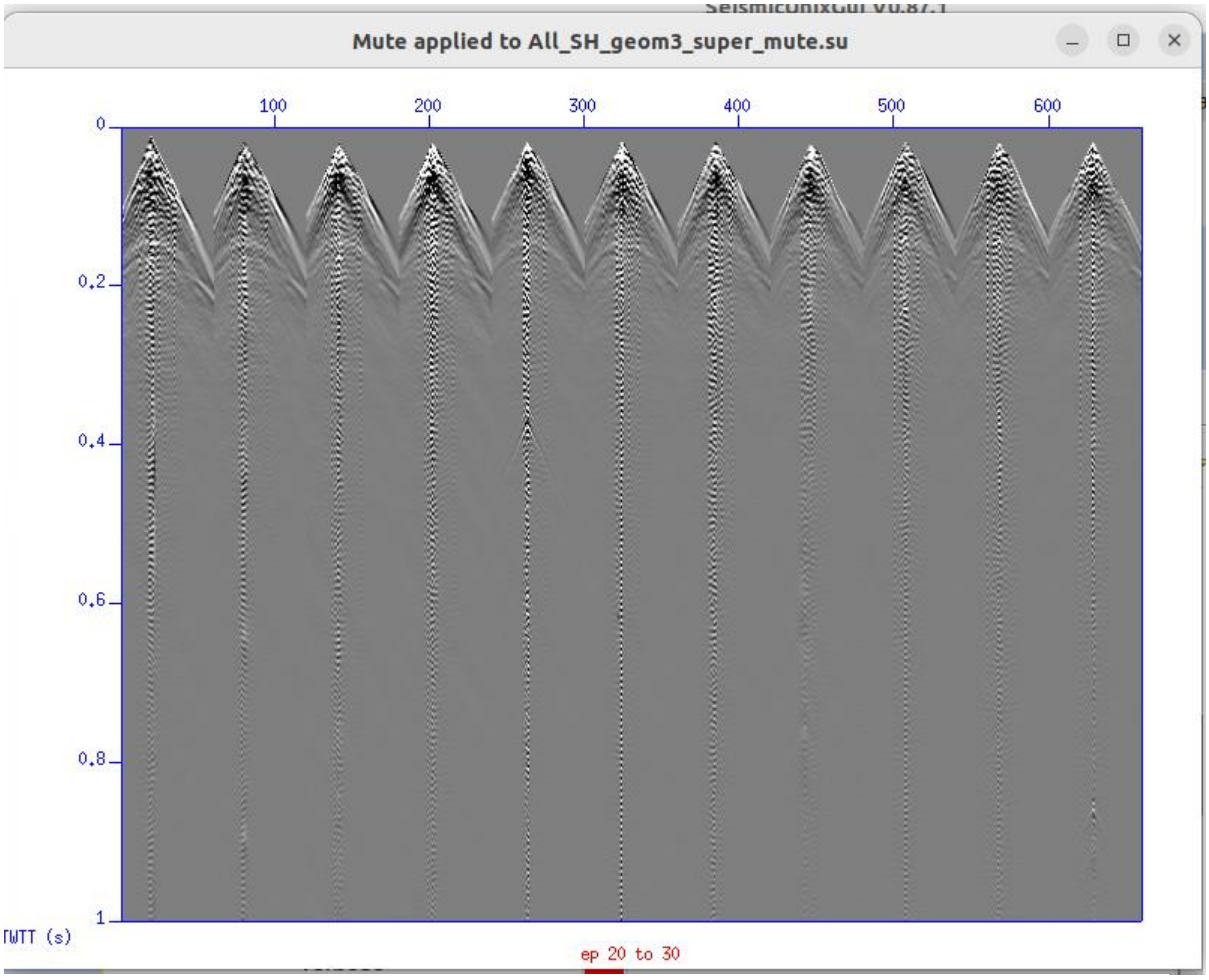


Figure 36. Display of muted SP gathers , where ep values range from 20 -30 (right).

Equivalent command-line instruction:

% perl /home/**user**/Servilleta/seis-
mics/pl/loma_blanca/053018/H/1/**user**/**5DD.view_muted_data.pl**

3.1.9 STEP 9. Preparation and creation of a preliminary stacked section

3.1.9.1 Sorting data as a function of cdp/cmp and offset

<MB 1> Flow->Open

Select file: **6A.sort_cdp.pl** (from GUI)

Purpose: Later steps such as normal moveout requires the data to be rearranged. The following sorting could also be combined into a single flow as part of the following steps.

Input /home/**user**/Sevilleta/seimics/data/loma_blanca//053018/H/1/su/**user**/**All_SH_geom3_mute**

Output: /home/**user**/Sevilleta/seimics/data/loma_blanca//053018/H/1/su/**user**/**All_SH_cdp_offset_sorted.su**

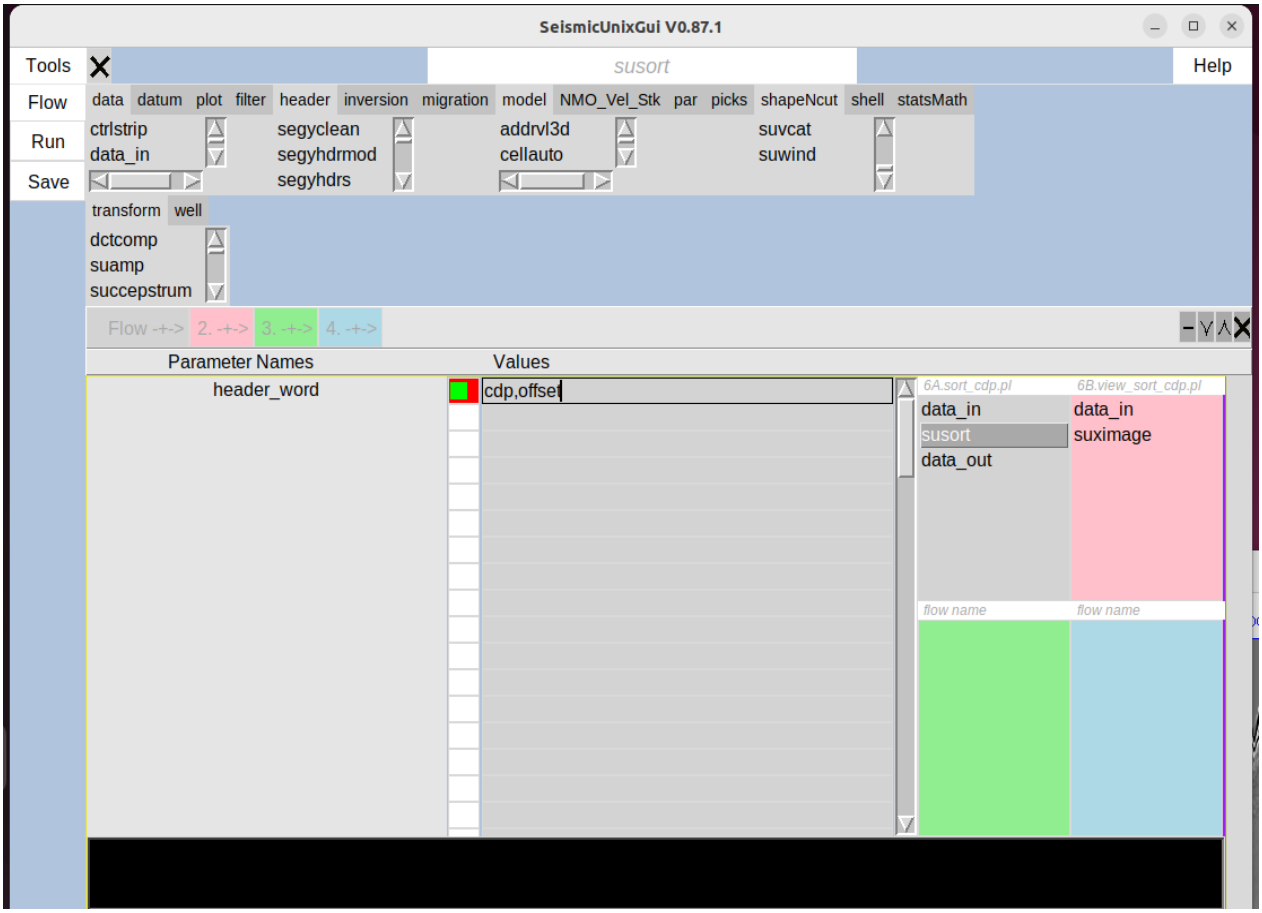


Figure 37. Display of parameters used by susort. Remember to click: Save (<MB1>) and Run (<MB1>)

Equivalent command-line instruction:

% perl /home/**user**/Servilleta/seismics/pl/loma_blanca/053018/H/1/**user**/**6A.sort_cdp.pl**

3.1.9.2 View data sorted into cdp/cmp gathers

<MB 1> Flow->Open

Select file: **6B.view_sort_cdp.pl** (from GUI)

Purpose: Visually inspect how the raw shotpoint gather data has been rearranged into cdp/cmp gathers

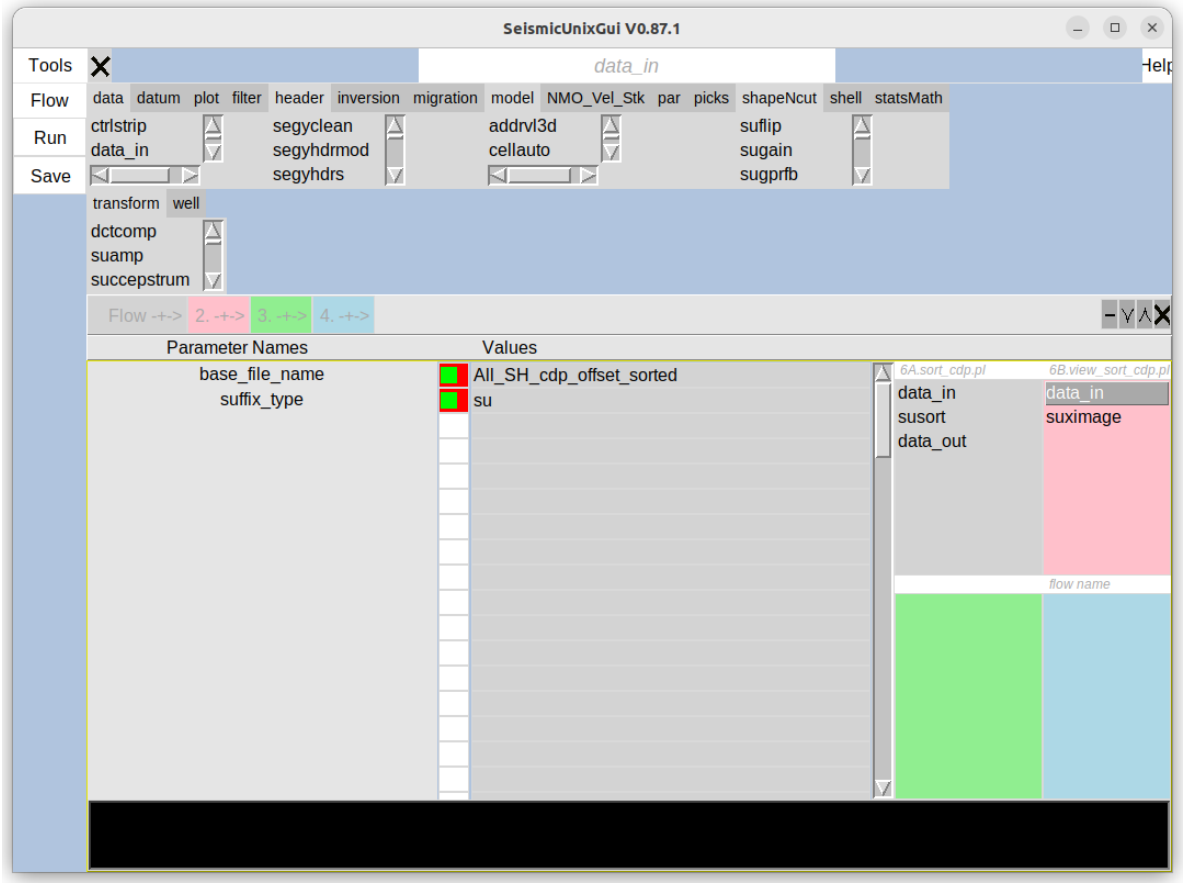


Figure 38. Display of parameters used to display all available field data rearranged into common midpoint/common depth point gathers.

Input: /home/**user**/Sevil-leta/seimics/data/loma_blanca//053018/H/1/su/**user**/**All_SH_cdp_offset_sorted.su**

Output: Image (Figure 39)

Equivalent command-line instruction:

% perl /home/**user**/Servilleta/seismics/pl/loma_blanca/053018/H/1/**user**/

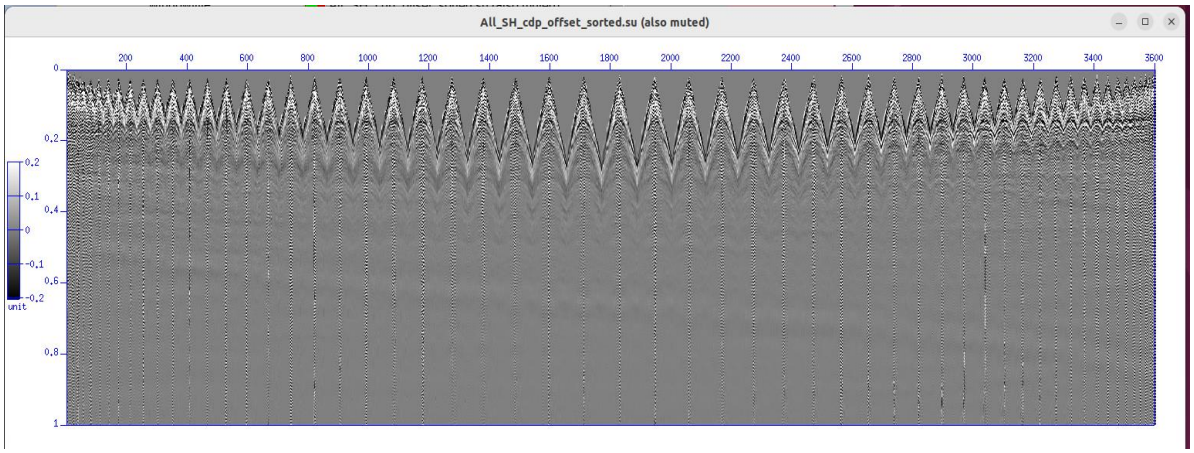


Figure 39. Display of CMP/CDP gathers. Fold increases linearly from edges toward the middle.

3.1.9.3 View of shotpoint gathers after normal moveout

<MB 1> Flow->Open

Select file: **7A.sunmo.pl** (from GUI)

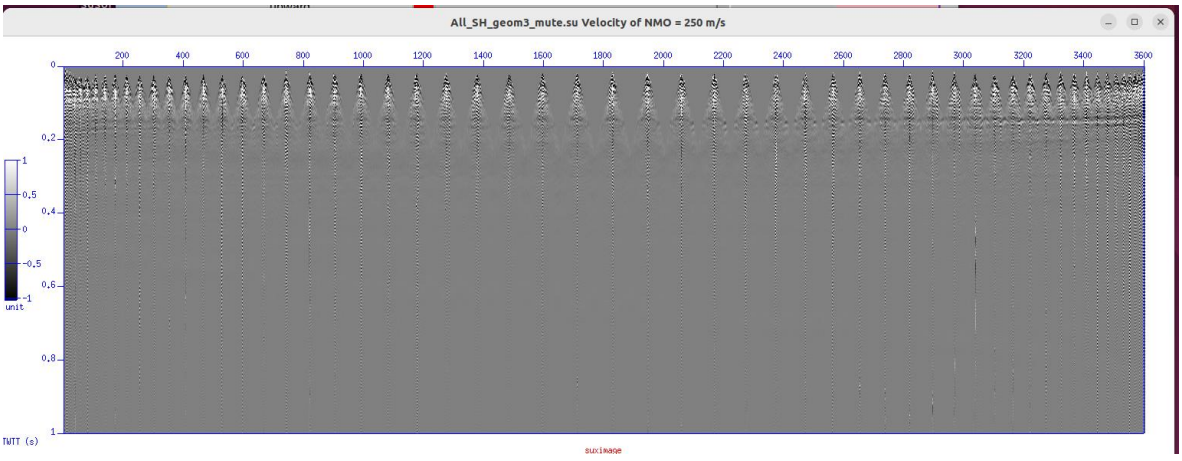


Figure 40. Display of CMP/CDP gathers corrected for normal moveout (Figure 39).

Purpose: Visually inspect effect of using moveout velocity values of 250 m/s and 300 m/s at two-way traveltimes of 0 and 1s, respectively. These values are preliminary for illustrative purposes. The Tool iVA can help determine velocity values interactively, but is not currently described in this tutorial.

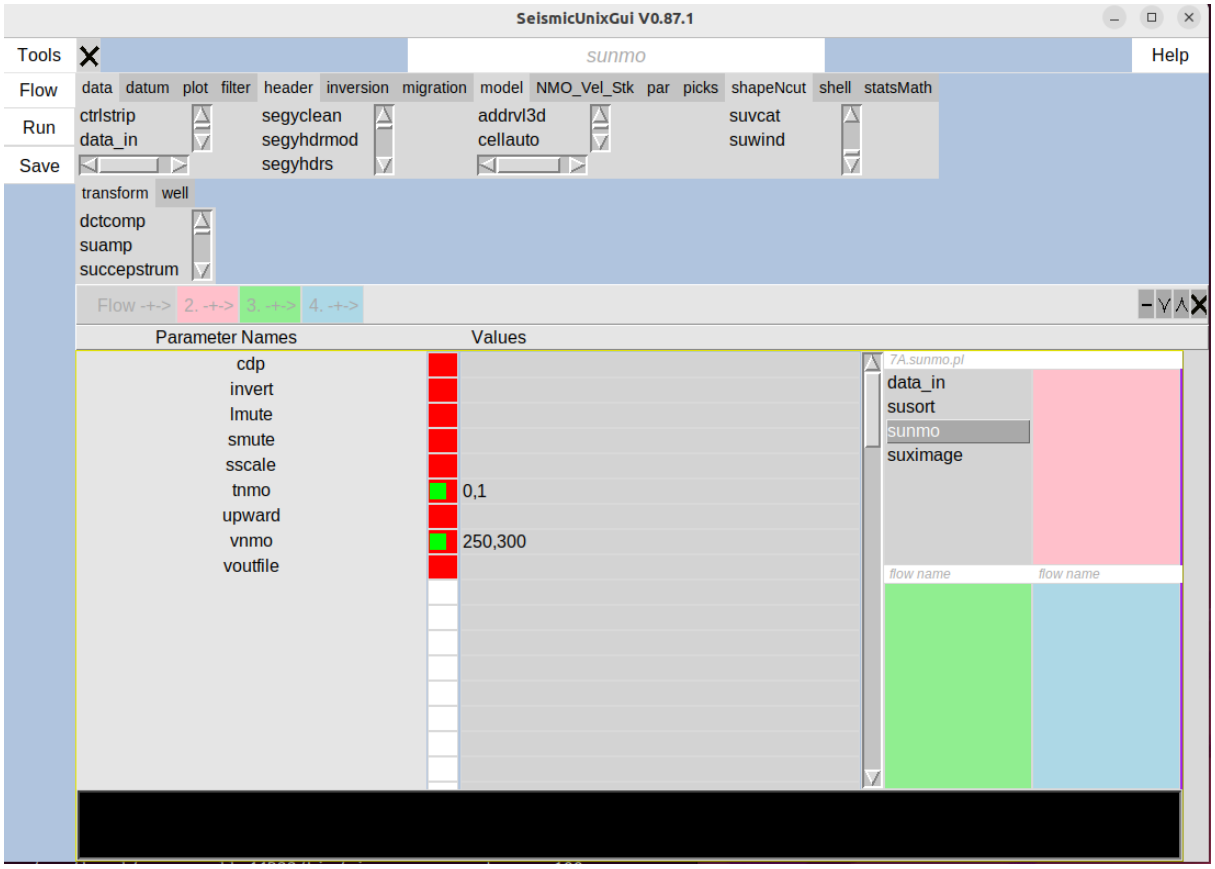


Figure 41. Parameter values used to display all CDP gathers after the application of normal moveout: velocity, time pairs = 250 m/s at 0 s and 300 m/s at 1 s of two-way traveltime.

Input: /home/**user**/Sevilleta/seimics/data/loma_blanca//053018/H/1/su/**user**/**All_SH_cdp_offset_sortedXXX.su**

Output: Image (Figure X)

Equivalent command-line instruction:

% perl /home/**user**/Servilleta/seismics/pl/loma_blanca/053018/H/1/**user**/7A.sunmo.pl

3.1.10 View of preliminary stacked profile

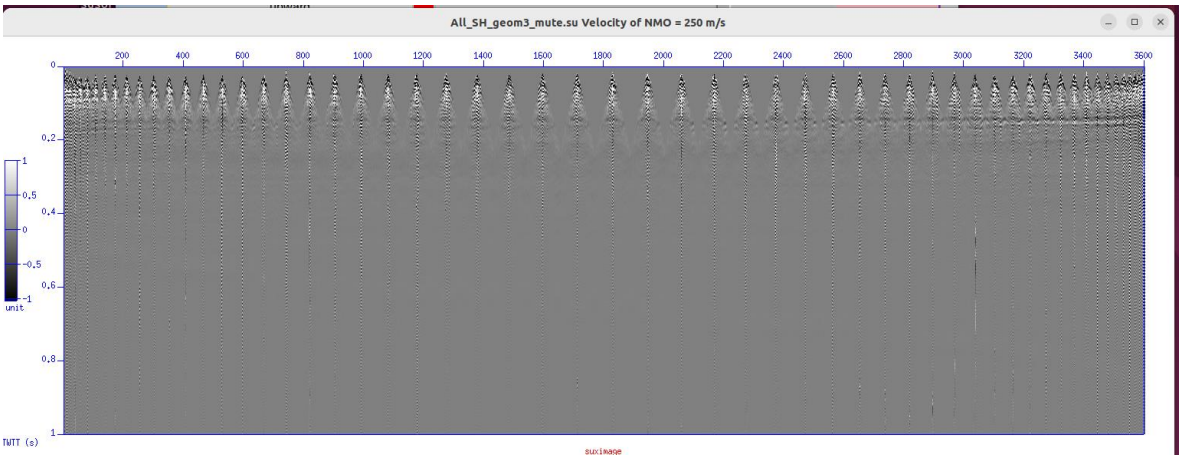


Figure 42. Display of CMP/CDP gathers corrected for normal moveout (Figure X).

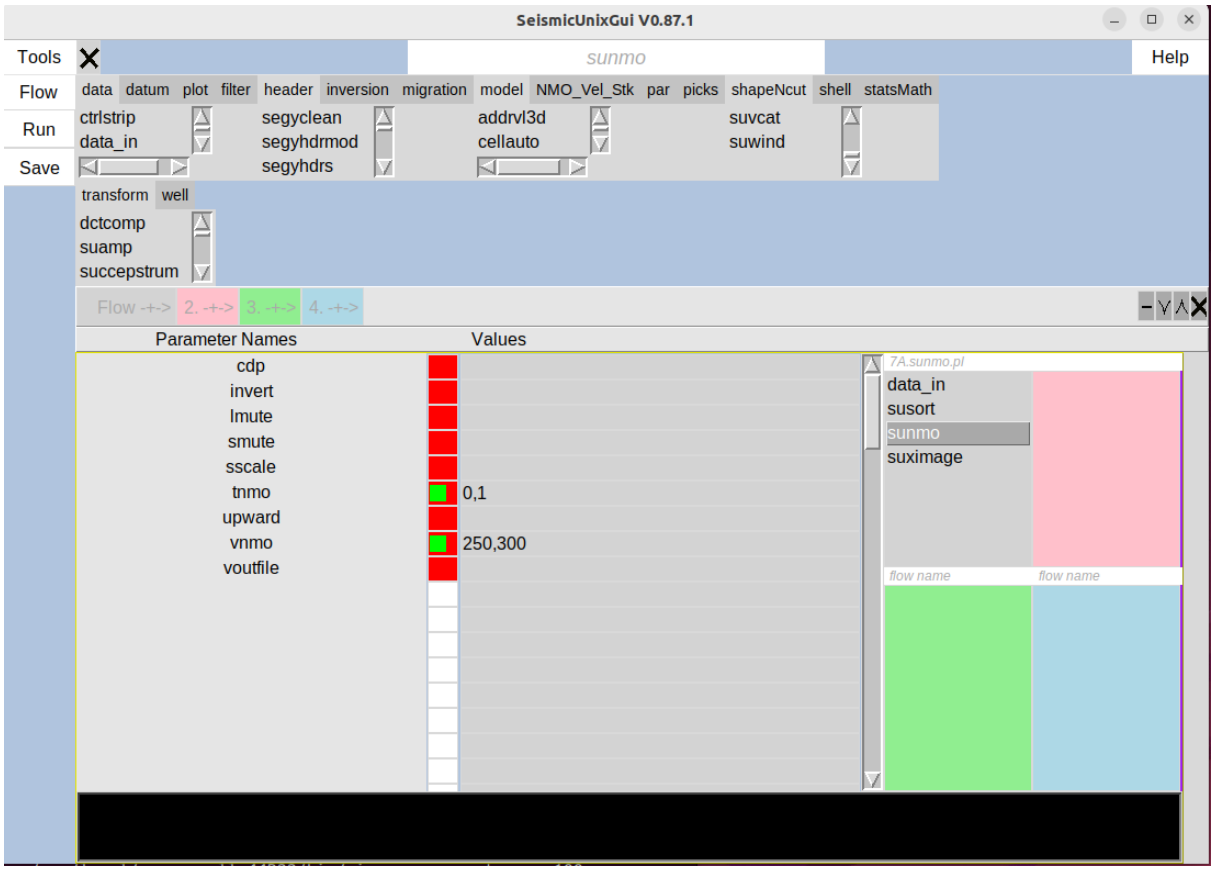


Figure 43. Parameter values used to display all CDP gathers after the application of normal moveout: velocity, time pairs = 250 m/s at 0 s and 300 m/s at 1 s of two-way traveltime

<MB 1> Flow->Open

Select file: **8A.sustack_gain_filter.pl** (from GUI)

Purpose: Create a seismic reflection image from the previous data using only a coarse velocity-depth model.

Input: /home/**user**/Sevilleta/seimics/data/loma_blanca//053018/H/1/su/**user**/**All_SH_cdp_offset_sortedXXX.su**

Output: Image (Figure 44)

Equivalent command-line instruction:

% perl /home/**user**/Servilleta/seismics/pl/loma_blanca/053018/H/1/**user**/**7A.sunmo.pl**

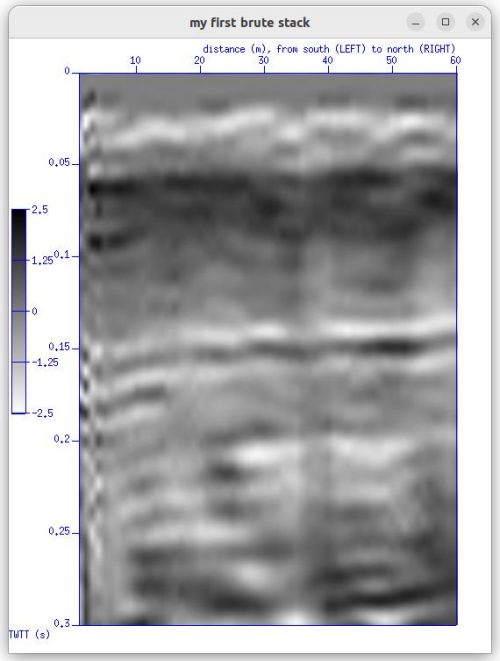


Figure 44. Image of stacked seismic profile. Note an interpreted strong reflective boundary about 150 ms.