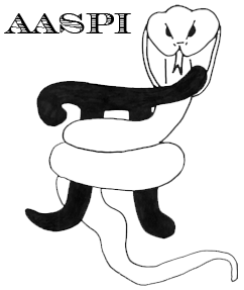


COMPUTING MULTIAZIMUTH OR MULTIOFFSET COHERENCE – PROGRAM similarity_prestack

Contents

Overview

Program **similarity_prestack** provides superior results when applied to a data volume that has been properly migrated using an azimuthally varying velocity model. Coherence is an edge-detection attribute that measures lateral changes in the seismic waveform and amplitude. The multi-azimuth coherence algorithm is based on an energy ratio coherence algorithm, which computes the ratio of coherent energy of seismic trace and total energy of seismic trace. Migrating seismic traces into bins depending on the source-receiver orientation provides azimuthally limited seismic amplitude volumes. Using a migration isotropic velocity may give rise to imaging misalignments in high azimuthally anisotropic reservoirs. Stacking those seismic gathers along offset domains results in azimuthally limited seismic amplitude volumes. Coherence computed from the poststack volume that stacks all azimuthally limited seismic amplitude volumes exhibits less geologic details and lower lateral resolution of migrated seismic images than coherence that computed from azimuthally limited seismic amplitude volumes (Chopra and Marfurt, 2007). Stacking these azimuthally limited amplitude volumes can suppress random noise.

Similarity_prestack computation flow chart

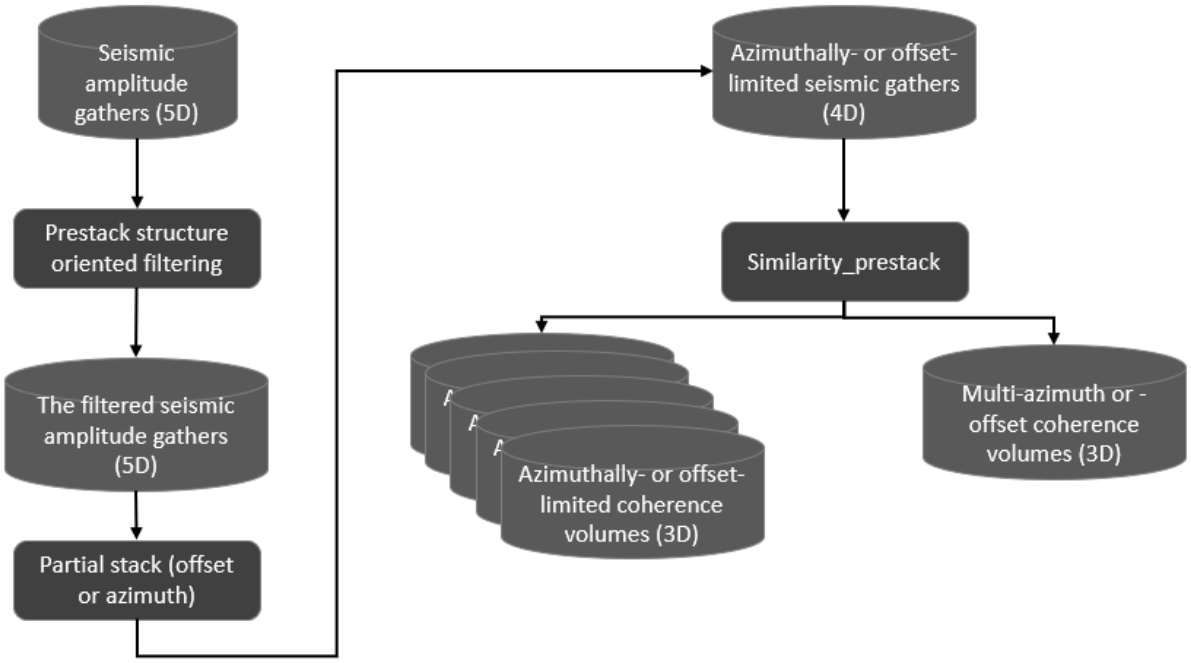


Figure 1.

Output file naming convention

Program **similarity_prestack** will always generate the following output files:

Output file description	File name syntax
Program log information	similarity_prestack_*unique_project_name_suffix*.log
Program error/completion information	similarity_prestack_*unique_project_name_suffix*.err

where the values in red are defined by the program GUI. The errors we anticipated will be written to the **.err* file and be displayed in a pop-up window upon program termination. These errors, much of the input information, a description of intermediate variables, and any software trace-back errors will be contained in the **.log* file.

In general, each trace of a seismic gather is called a sector. A sector may be an offset-azimuth sector, and offset vector tile, or if the data have been partially stacked, simply an azimuth or an offset. The name of the sector will be contained in the output files where the name variable sector below may take on values of *sector, azimuth,* or *offset*.

If you check the *Output attributes for each azimuth or offset component* box, you will generate files with the following names, where the variable *number* takes on values of 1, 2, 3, to the total number of sectors in the input data volume:

Output file description	**File name syntax**
energy-ratio similarity of each sector	energy_ratio_similarity_*unique_project_name_suffix_sector_number*.H
outer-product similarity (semblance) of each sector	outer_product_similarity_*unique_project_name_suffix_sector_number*.H
Sobel-filter similarity of each sector	sobel_filter_similarity_*unique_project_name_suffix sector_number*.H
inline component of the energy gradient of each sector	inline_energy_gradient_*unique_project_name_suffix_sector_number*.H
crossline component of the energy gradient of each sector	crossline_energy_gradient_*unique_project_name_suffix_sector_number*.H

If you check the *Output multiazimuth or multioffset attributes* box the program will generate the following files where *multi_sector* may take on the values of *multi_sector, multi_azimuth,* or *multi_offset*:

Outputfile description	**File name syntax**
energy-ratio similarity of stacked data	energy_ratio_similarity_*unique_project_name_suffix*_multi_*sector*_stack.H
outer-product similarity (semblance) of stacked data	outer_product_similarity_*unique_project_name_suffix*_multi_*sector*_stack.H
Sobel-filter similarity of stacked data	sobel_filter_similarity_*unique_project_name_suffix*_multi_*sector*_stack.H
inline component of the energy gradient of stacked data	inline_energy_gradient_*unique_project_name_suffix*_multi_*sector*_stack.H
crossline component of the energy gradient of stacked data	crossline_energy_gradient_*unique_project_name_suffix*_multi_*sector*_stack.H

For reference, you will also obtain the attributes computed from the stacked data. No compensation for mutes are taken into account, so there may be some lower amplitudes in the shallow section. The names of these files will be of the form:

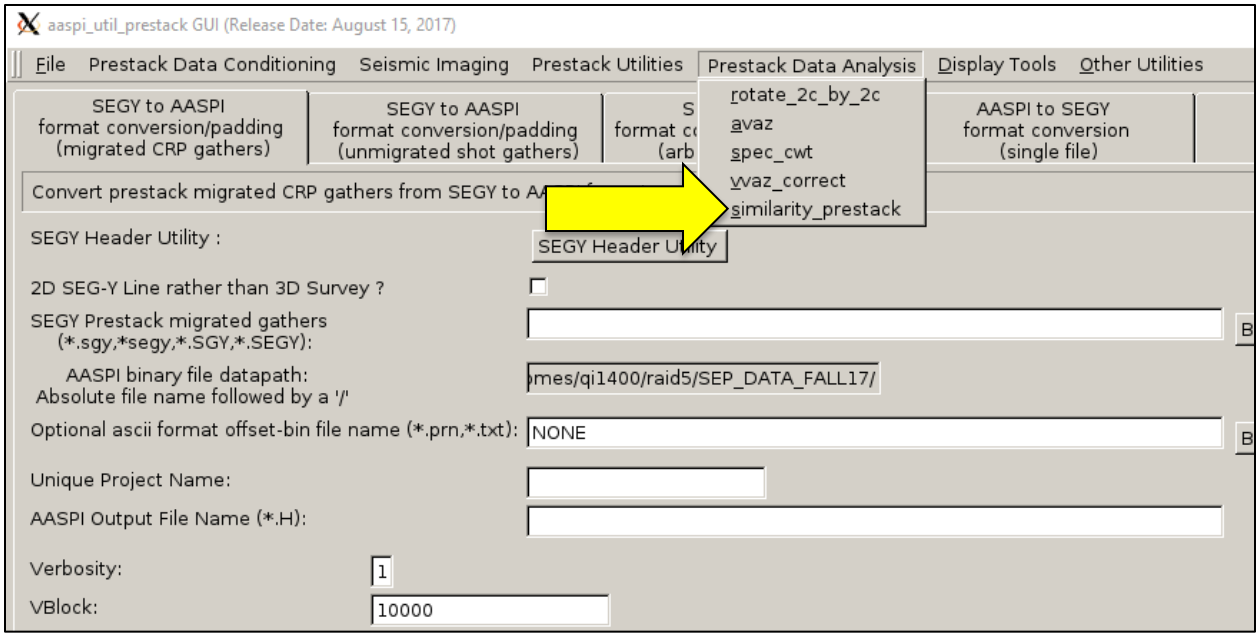
Output file description	File name syntax
energy-ratio similarity of stacked data	energy_ratio_similarity_*unique_project_name_suffix_sector_*stack.H
outer-product similarity (semblance) of stacked data	outer_product_similarity_*unique_project_name_suffix_sector_*stack.H
Sobel-filter similarity of stacked data	sobel_filter_similarity_*unique_project_name_suffix sector_*stack.H
inline component of the energy gradient of stacked data	inline_energy_gradient_*unique_project_name_suffix_sector_*stack.H
crossline component of the energy gradient of stacked data	crossline_energy_gradient_*unique_project_name_suffix_sector_*stack.H

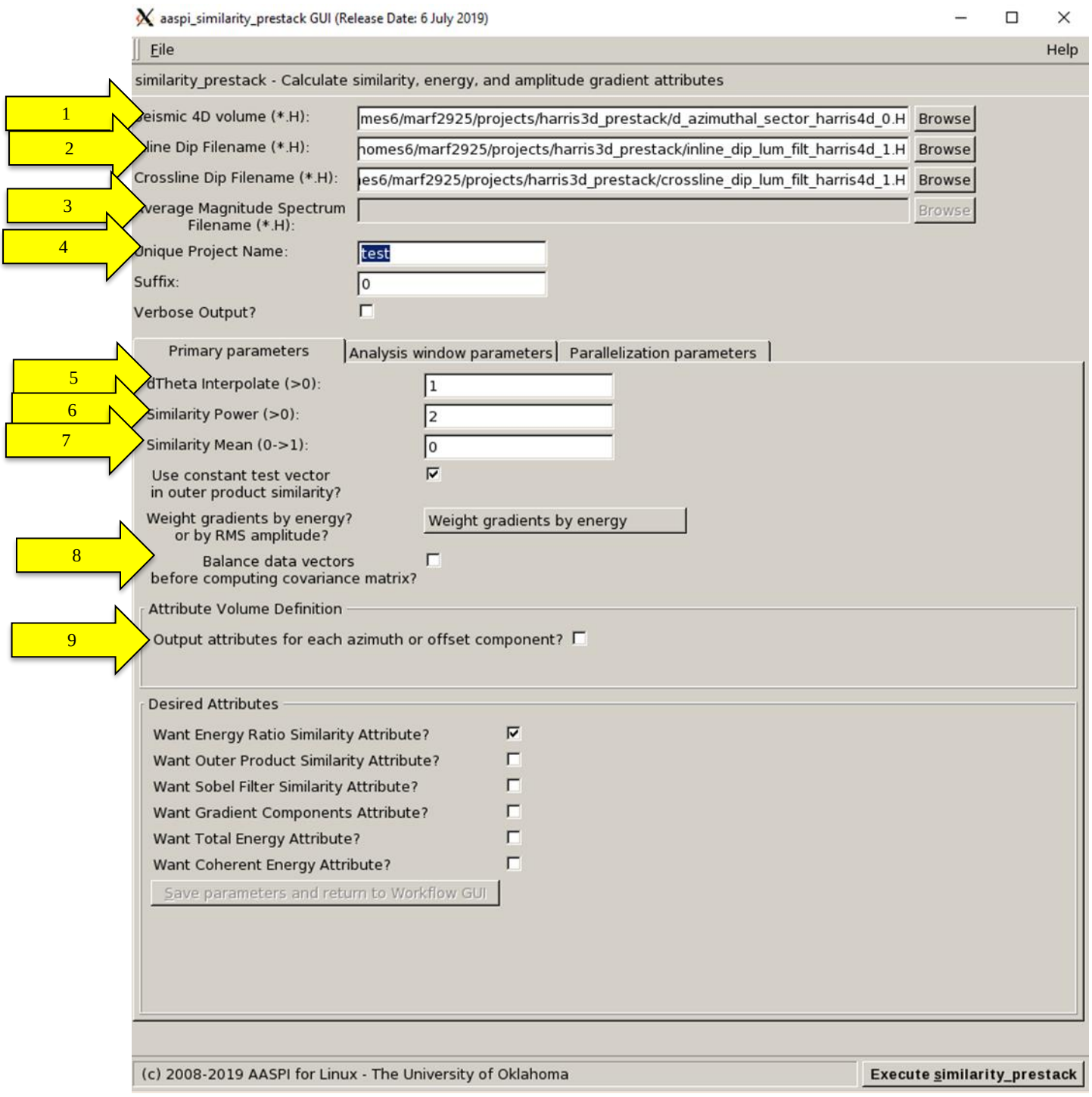
We can also generate attributes that are either functions of the energy or of the RMS amplitude within the analysis window. If we choose to *Output energy and weight gradients by coherent energy*, the following files will be called:

Output file description	File name syntax
The total energy in the analysis window	total_energy_*unique_project_name_suffix*.H
The coherent energy in the analysis window	coherent_energy_*unique_project_name_suffix*.H

Parameter description

To begin, at the top of the **aaspi_util_prestack** GUI (the first image below), click the *Prestack_Data_Analysis* tab, located to the top tab. A downdrop menu will appear containing the major AASPI programs. We choose **similarity_prestack**, and the **similarity_prestack** window will popup, which is shown like the second image below:


aaspi_util_prestack GUI (Release Date: August 15, 2017)
File
Prestack Data Conditioning
Seismic Imaging
Prestack Utilities
Prestack Data Analysis
Display Tools
Other Utilities
rotate_2c_by_2c
avaz
spec_cwt
vvaz_correct
similarity_prestack
SEGY to AASPI format conversion/padding (migrated CRP gathers)
SEGY to AASPI format conversion/padding (unmigrated shot gathers)
AASPI to SEGY format conversion (single file)
Convert prestack migrated CRP gathers from SEGY to AA
SEGY Header Utility :
SEGY Header Utility
2D SEG-Y Line rather than 3D Survey ?
SEGY Prestack migrated gathers (*.sgy,*segy,*.SGY,*.SEGY):
AASPI binary file datapath: Absolute file name followed by a '/'
mes/qi1400/raid5/SEP_DATA_FALL17/
Optional ascii format offset-bin file name (*.prn,*.txt):
NONE
Unique Project Name:
AASPI Output File Name (*.H):
Verbosity:
1
VBlock:
10000



Click (1) *Browse,* and select *d_lum_filt_gathers_justin3d_0.H* as your input seismic data set. The data is offset partial stacked data that means this is a 4D data after stacked from a 5D data. In general, seismic data with offset is one 4D volume, however seismic data with multiazimuth may be either one 4D volume or a few azimuthally limited 3D volumes. If you have a few azimuthally limited seismic volumes, you need to combine these to be one 4D volume. We do have a program to combine seismic volumes together, and as such will be shown after. You also need to (2)

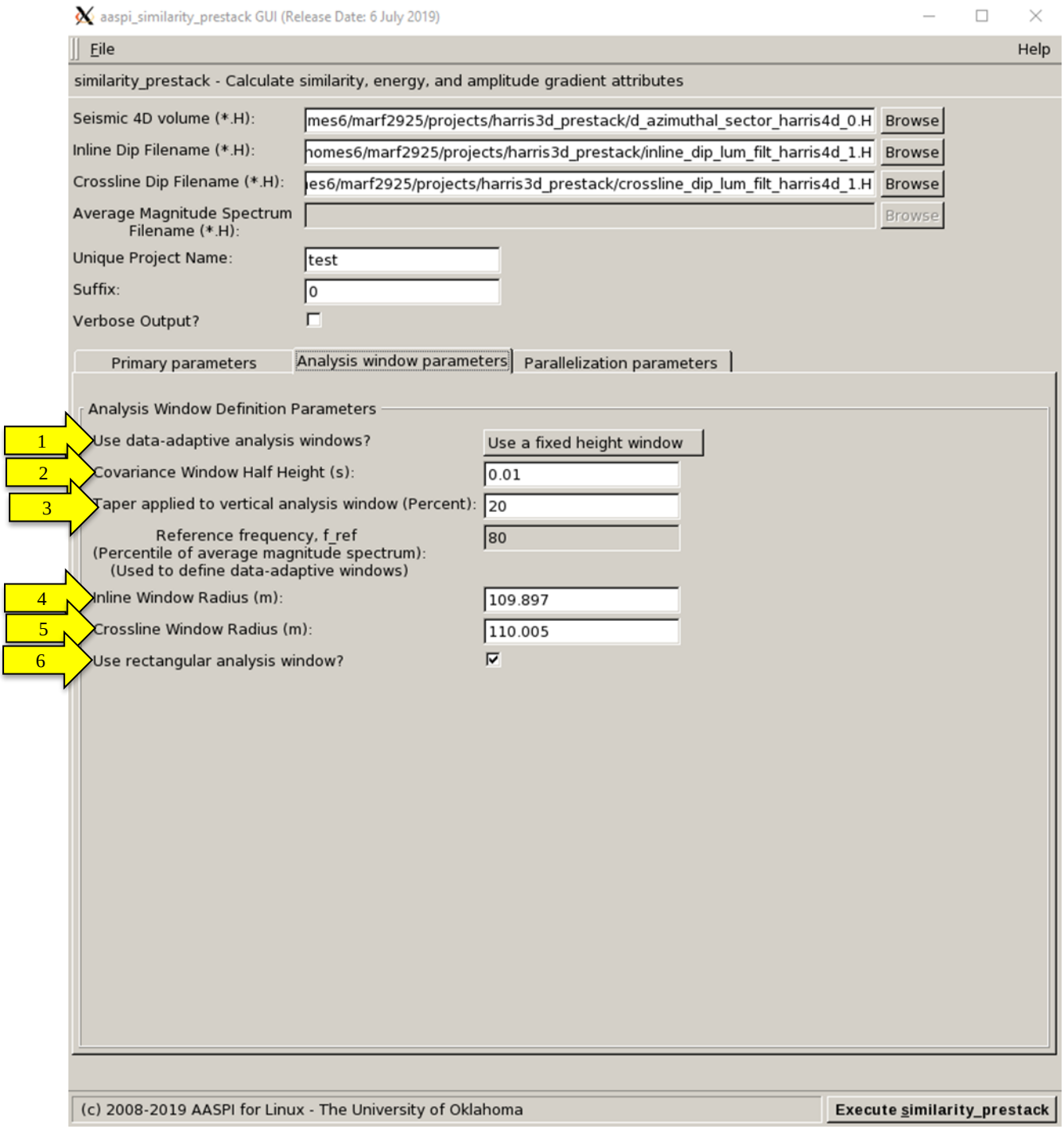
choose inline and crossline dip components, which can be computed from the **dip3d** program. We can focus on using the dip components computed from poststack data in multiazimuth or multioffset coherence algorithm. Then (3) select if using the average power spectrum that is an advanced option that allows the definition of a data adaptive analysis window, which may be useful for depth-migrated data where the dominant wavelength changes significantly from shallow to deeper depths. For now, use a fixed window and proceed to (4) select *Unique Project Name* and *Suffix*

The primary Parameters Tab

Similarity values will range between 0 and 1, with most of the values biased towards 1. (5) dtheta interpolation is 1. If one (6) takes the power of the similarity, the distribution moves towards zero, providing improved contrast in the resulting image. This option remains from a time when many workstation software programs had only limited color bar manipulation capabilities. In a modern system it makes no difference whether one maps nonlinearly (powers) the similarity and plots it against a linear gray scale color bar, or if one plots the unscaled similarity against a nonlinear gray scale color bar. (8) Whether or not you wish to balance the covariance matrix sample vectors prior to constructing it (see the appropriate Theory box below). (9) Whether or not you wish to output each azimuthally-limited or offset-limited coherence volumes.

The Analysis window parameter Tab

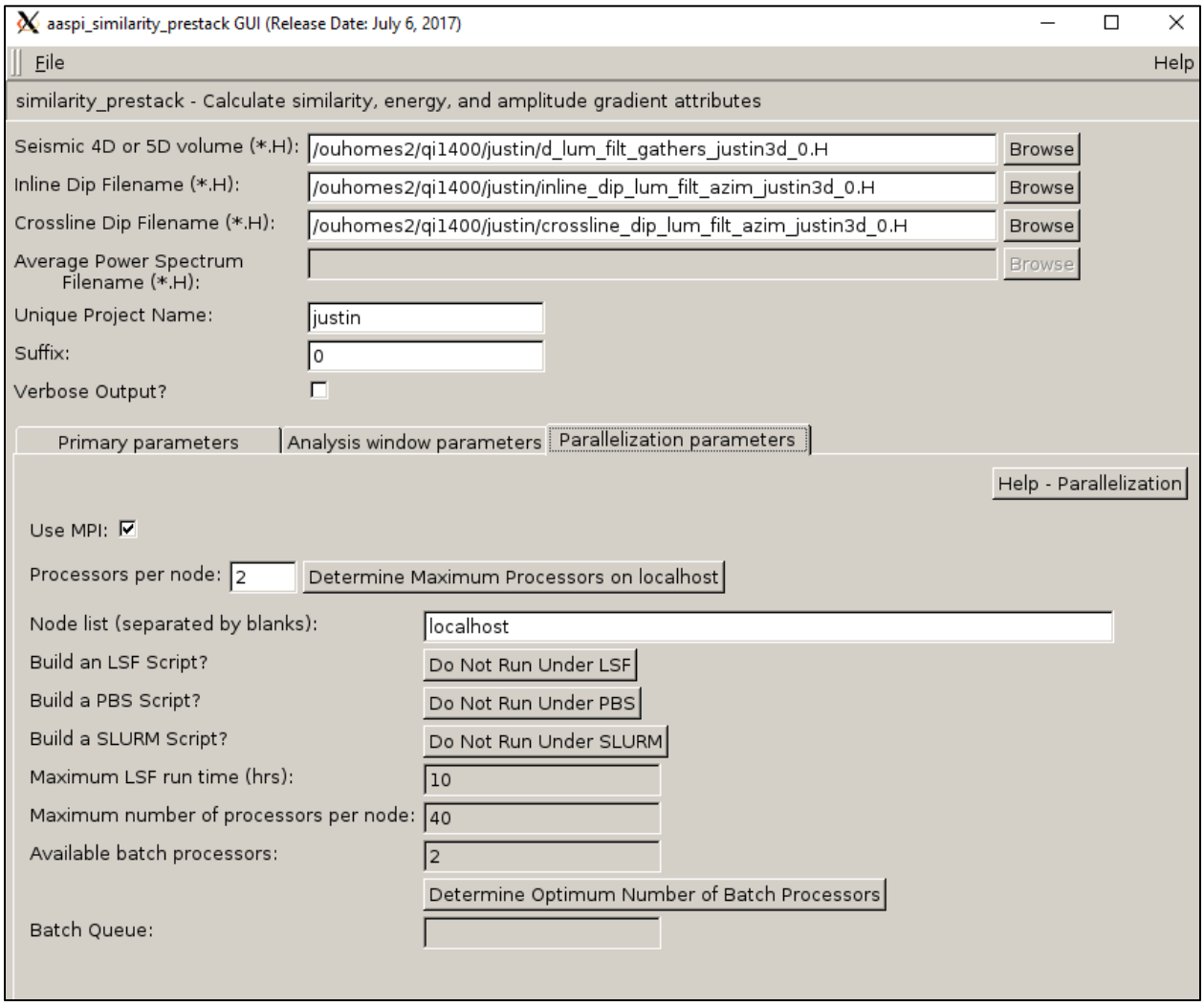
The image below shows the GUI when the Analysis window parameters tab has been selected. By default, **similarity3d** will use (1) a fixed-size analysis window. We will investigate what happens if we toggle this button shortly. For now, define the (2) covariance window half height and (3) percent tapers applied to the samples vertically. Drop down to (4) the inline and (5) crossline window radii. The default will be ±1 (one) trace in each direction, which for the harrs survey are 110 ft and 110 ft. (6) a rectangular vs. an elliptical analysis window. For small windows like this, the rectangular window provides more robust estimates of inline and crossline coherent energy gradient components. For larger elliptical or circular windows, this is less important.



The Parallelization Parameters Tab

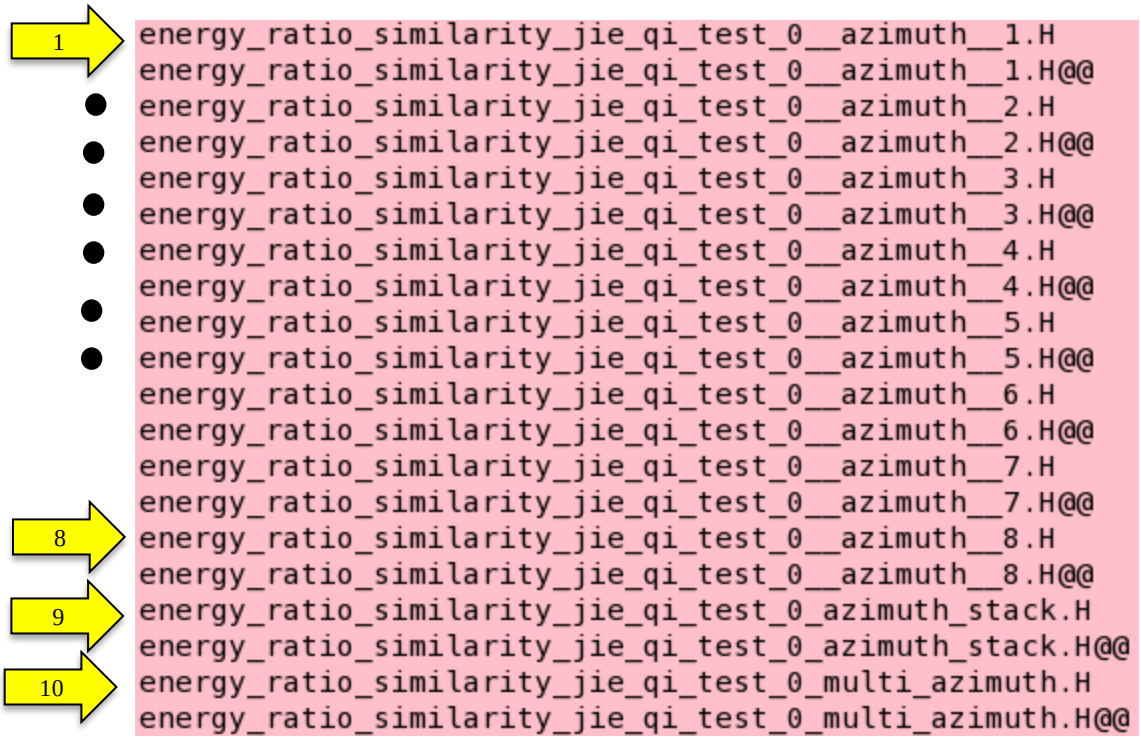
The parallelization parameters tab for **similarity_prestack** is the same for all AASPI applications. One can run across multiple processors and cores on a single node, across nodes if that is allowed by your IT department, or on large batch supercomputers using LSF, PBS, or SLURM scripts.

Details on parallelization can be found in the parallelization section of the documentation Overview: AASPI Software Parallelization. The **similarity_prestack** volume uses a stencil-based parallelization scheme; this results in suboptimal performance for surveys that do not approximate a rectangular shape, where cores assigned to dead or padded traces lie idle.



Execution

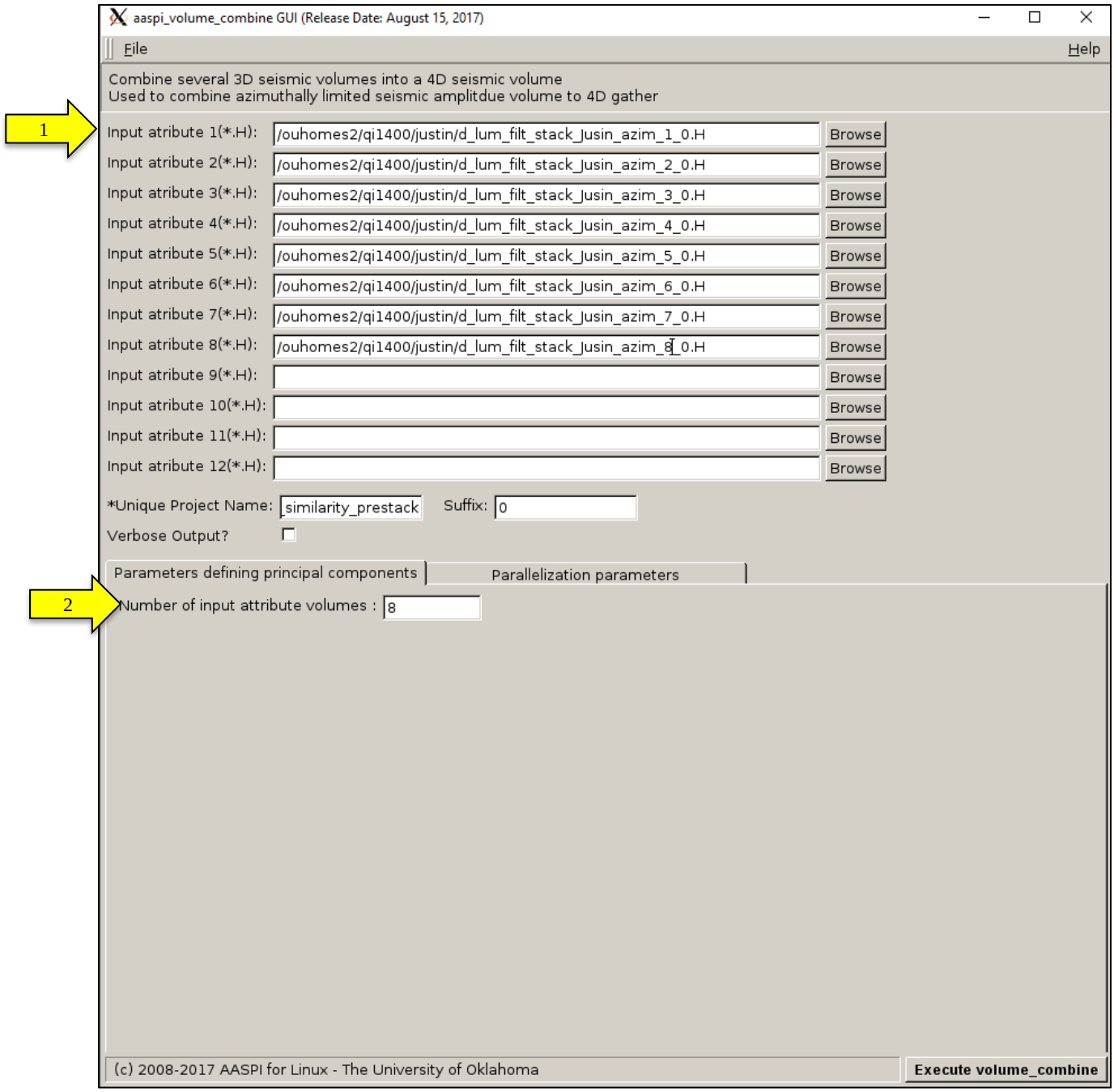
After selecting all the parameters you wish to change, click *Execute similarity_prestack* and generate several output files (see image on next page):



We only compute **energy_ratio_similarity**, and the program generates 10 output volumes. Volume (1) to volume (8) azimuthally limited coherence volumes computed from the 4D input volume. Note that the suffix at the end of output name is from azim1 to azim8. Each azimuthally limited coherence volume is 3D and represents corresponding coherence in different azimuths. If you compute multioffset coherence, these volumes will be offset limited coherence volumes. The (9) volume is poststack coherence. We have an inner algorithm in **similarity_prestack** program that stacking the 4D input gather to be 3D stacked volume, then compute poststack coherence. Thus, users don't need to run **similarity3d** but still can have **similarity3d** results. The (10) volume is multiazimuth coherence volume that has a suffix *m* at the end of output name. Note that these 10 outputs are all 3D volumes, thus we will use 3D plot to show these volumes. The below box indicates algorithm part of this multiazimuth coherence. The details of multiazimuth coherence algorithm can be found in publication by Qi et al. (2017).

Combining multiple 3D volumes into a single 4D volume

Let's go back to talk about how to combine 6 or 8 azimuthally limited seismic volumes into 1 4D seismic amplitude volume. We use program **volume_combine** to do azimuthally limited seismic volume combination, which is under *Prestack_Utilities*. Using Browse button to select input volumes (1), and set parameter (2) to be number of input volume. The output is a four 4D volume named as *d_volume_combine_'nattr'_'unique_project_name'_'suffix'.H*


aaspi_volume_combine GUI (Release Date: August 15, 2017)
File
Help
Combine several 3D seismic volumes into a 4D seismic volume
Used to combine azimuthally limited seismic amplitdue volume to 4D gather
1
Input atribute 1(*.H): /ouhomes2/qi1400/justin/d_lum_filt_stack_Jusin_azim_1_0.H Browse
Input atribute 2(*.H): /ouhomes2/qi1400/justin/d_lum_filt_stack_Jusin_azim_2_0.H Browse
Input atribute 3(*.H): /ouhomes2/qi1400/justin/d_lum_filt_stack_Jusin_azim_3_0.H Browse
Input atribute 4(*.H): /ouhomes2/qi1400/justin/d_lum_filt_stack_Jusin_azim_4_0.H Browse
Input atribute 5(*.H): /ouhomes2/qi1400/justin/d_lum_filt_stack_Jusin_azim_5_0.H Browse
Input atribute 6(*.H): /ouhomes2/qi1400/justin/d_lum_filt_stack_Jusin_azim_6_0.H Browse
Input atribute 7(*.H): /ouhomes2/qi1400/justin/d_lum_filt_stack_Jusin_azim_7_0.H Browse
Input atribute 8(*.H): /ouhomes2/qi1400/justin/d_lum_filt_stack_Jusin_azim_8_0.H Browse
Input atribute 9(*.H): Browse
Input atribute 10(*.H): Browse
Input atribute 11(*.H): Browse
Input atribute 12(*.H): Browse
*Unique Project Name: similarity_prestack
Suffix: 0
Verbose Output?
Parameters defining principal components
Parallelization parameters
2
Number of input attribute volumes : 8
(c) 2008-2017 AASPI for Linux - The University of Oklahoma
Execute volume_combine

Implementation details: Multiazimuth coherence description in the AASPI software

The covariance matrix is constructed from a suite of sample vectors that are parallel to structural dip. The covariance matrix for this analysis window is,

$$C_{mn} = \sum_{k=-K}^{+K} (d_{km} d_{kn} + d_{km}^{H} d_{kn}^{H}),$$

where the superscript H denotes the Hilbert transform along traces, and the subscripts m and n are indices of input traces (1, 2, ..., M). For example, the element C_{23} in the covariance matrix C_{mn} is $\sum_{k=-K}^{+K}(d_{k2}d_{k3} + d_{k2}^{H}d_{k3}^{H})$. The Hilbert transform (90o phase rotated) version of the data does not modify the vertical resolution but improves areas of low SNR about zero crossing (Marfurt, 2006). The first eigenvector v(1) of the covariance matrix C best represents the lateral variation of each sample vector of the constituent.

We generalize the concept of energy-ratio coherence by summing J covariance matrices $C(\phi_j)$ computed from each of the J azimuthally sectored data volumes:

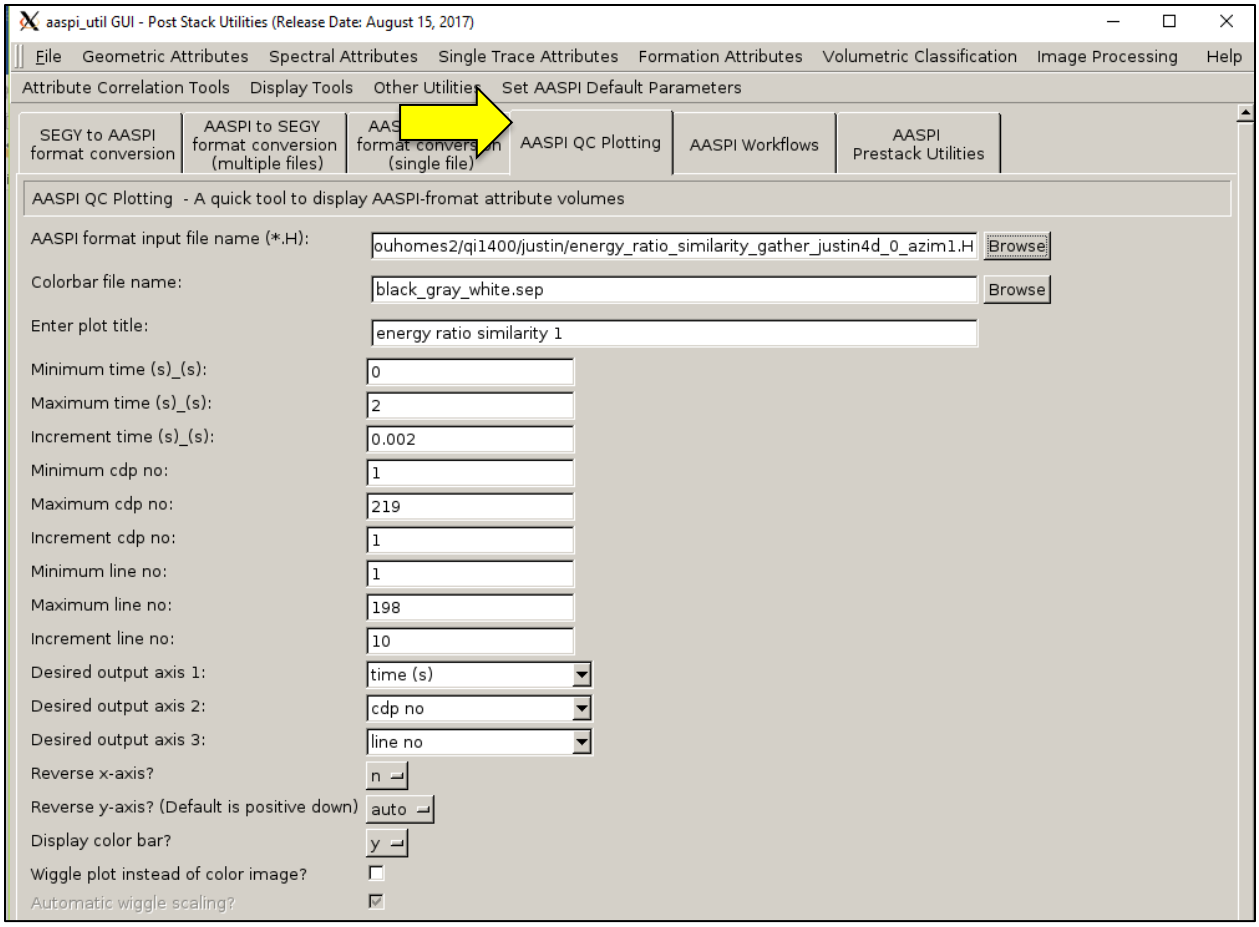
$$\mathbf{C}_{multi-\varphi} = \sum_{j=1}^{J} \mathbf{C}(\varphi_j).$$

The summed covariance matrix is of the same M by M size as the original single-azimuth covariance matrix but is now composed of J times as many sample vectors. As the conventional covariance matrix, the multi-azimuth covariance matrix is a symmetric positive definite matrix. Eigen-decomposition of the multi-azimuth covariance matrix is a nonlinear process, such that the first eigenvector of the summed covariance matrix is not a linear combination of the first eigenvectors computed for the azimuthally limited covariance matrices, in which case the resulting coherence would be the average of the azimuthally limited coherence computations.

Geologic details in each azimuthally seismic images are transferred into sample vectors. Summing sample vectors provides a means of summing geologic anomalies into the multi-azimuth covariance matrix, such as stacking up azimuthally limited coherence. This nonlinear Eigen-decomposition of the multi-azimuth covariance matrix has advantages in suppressing random noise that would help deal with random noise in azimuthally limited seismic volumes. To lessen computation cost, azimuths are commonly binned into six 30° or eight 22.5° sectors, although finer binning is common in large processing shops.

Example:

We can plot our outputs to QC the results by clicking (1) the AASPI QC Plotting tab in the **aaspi_util** GUI (see next page):



Time slices at t=0.74 s through the six different azimuthally limited seismic amplitude volumes of the test survey looks like

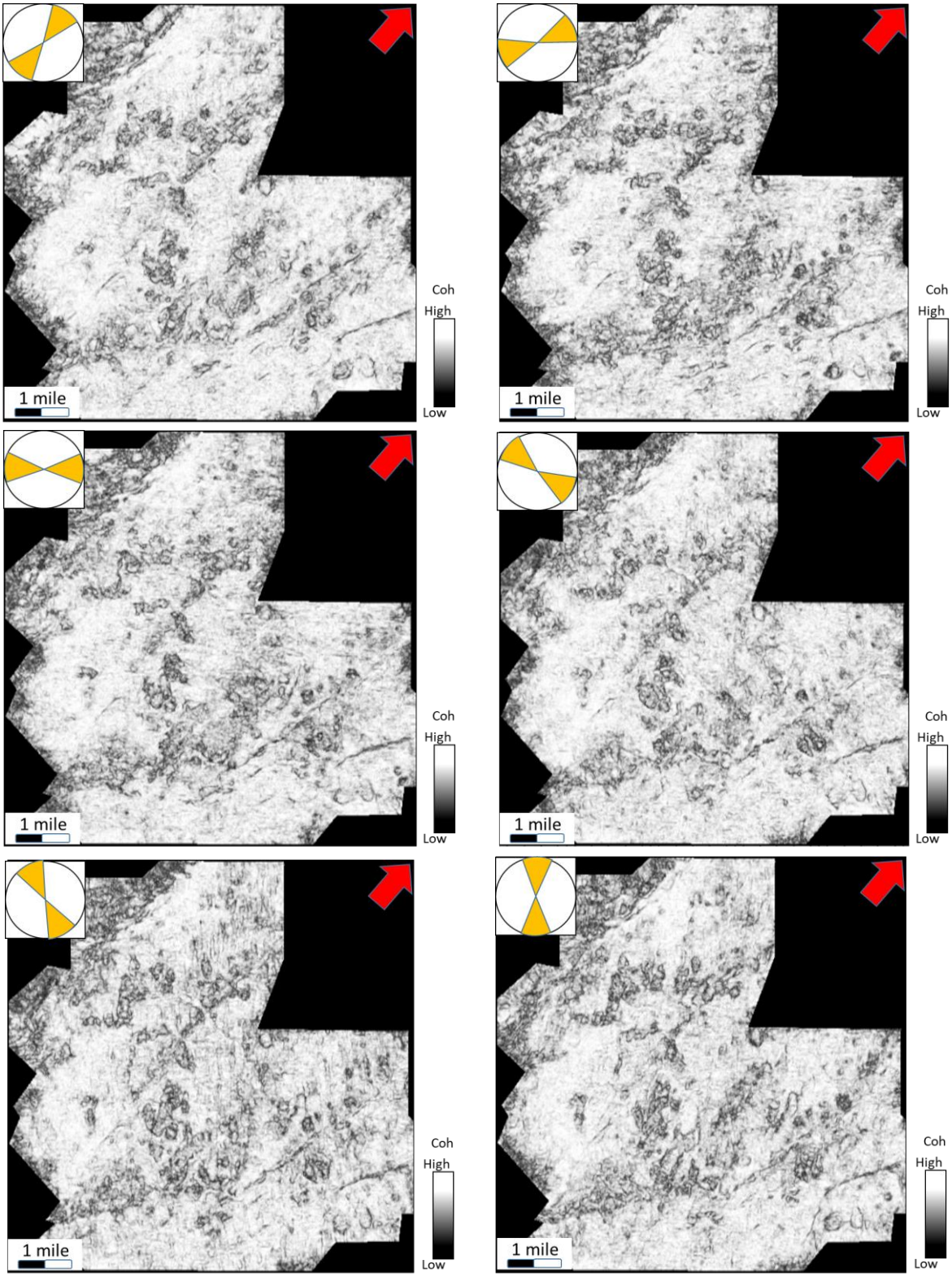

1 mile
Coh
High
Low
1 mile
Coh
High
Low
1 mile
Coh
High
Low
1 mile
Coh
High
Low
1 mile
Coh
High
Low
1 mile
Coh
High
Low

Figure 2.

Stacking the six seismic amplitude volumes and then computing coherence (the conventional analysis workflow) gives the result shown in left top figure. This image shows increased SNR but slightly lower lateral resolution than the azimuthally limited coherence time slices shown in azimuthally limited coherence images. The right top figure shows the result of stacking the six images shown in azimuthally limited coherence images. The resolution on stacked coherence is lower than that of post-stack coherence; however, edges of the karst features appear more pronounced than on the traditional coherence computation. The below, central figure, shows the multi-azimuth coherence result computed using the covariance matrix. Note that the multi-azimuth coherence displays the higher spatial resolution than either traditional coherence or the stacked azimuthal coherence.

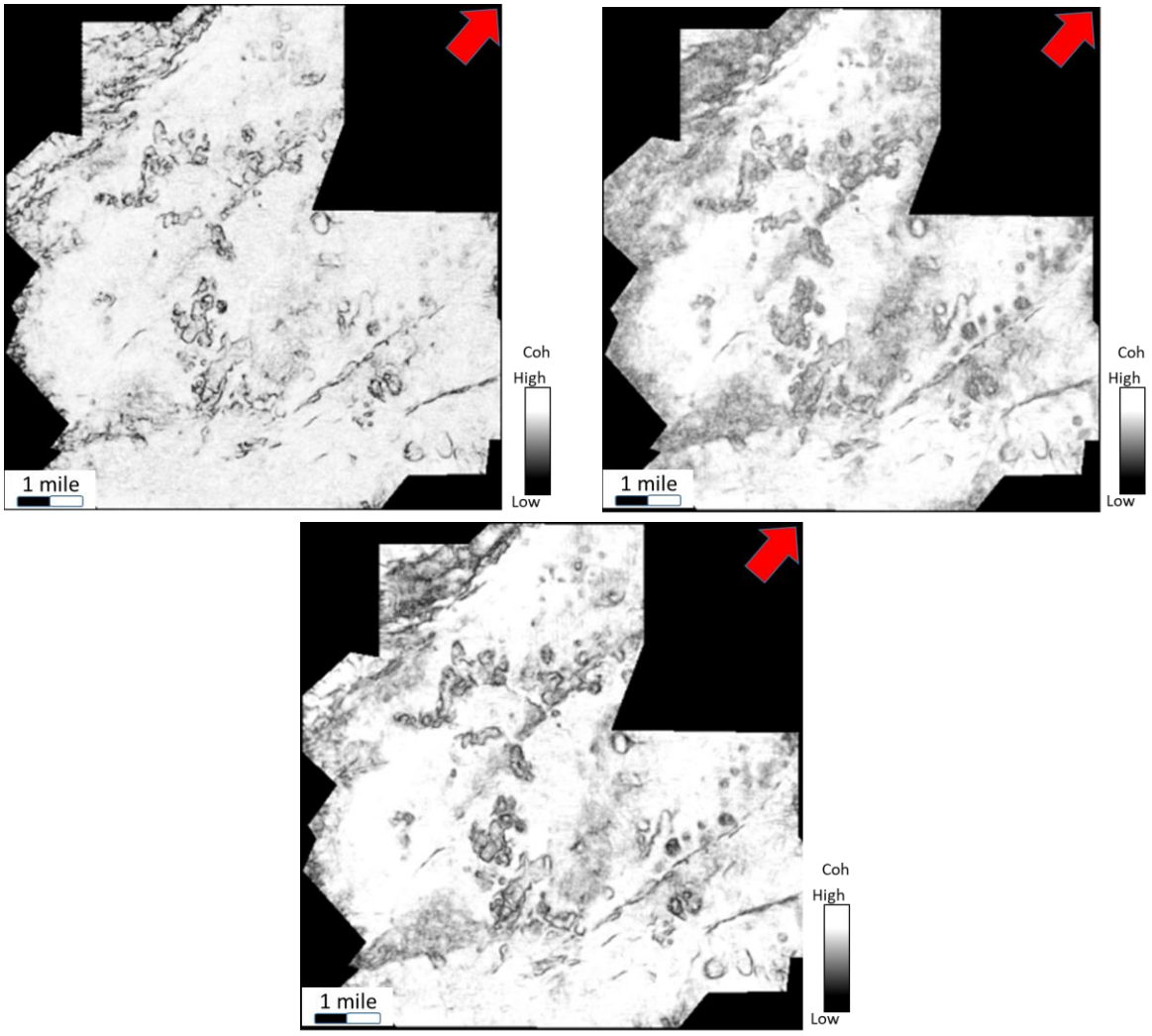


Figure 3.

References

Chopra, S., and K. J. Marfurt, 2007, Seismic attributes for prospect identification and reservoir characterization: SEG, Geophysical Development Series.

Chopra, S, and K. J. Marfurt, 2019, Multispectral, multiazimuth, and multioffset coherence attribute applications: Interpretation, **7**, SC21-SC32.

Qi, J., F. Li, and K. J. Marfurt. 2017, Multiazimuth coherence: Geophysics, **82**, no. 6, o83-o89.