# COMPUTING SPECTRAL COMPONENTS USING A COMPLEX MATCHING PURSUITE ALGORITHM – PROGRAM spec_cmp

## Alternative Spectral Decomposition Algorithms

Spectral decomposition methods can be divided into three classes: those that use quadratic forms, those that use linear forms and those that use atom decomposition. Quadratic forms are based on Wigner-Ville distribution, which can be easily designed but lose the phase component of the data and therefore cannot be used in reconstruction. Linear forms are based on short time Fourier transform (STFT), and include the S transform and the continuous wavelet transform (or CWT) as in program **spec_cwt**. Program **spec_clssa** is a nonlinear implementation of the short time Fourier transform. Atomic decomposition reconstructs the signal by using small “atom-sized” signals (in our case wavelets), such as complex matching pursuit (program **spec_cmp**) and the Hilbert-Huang transform.

## spec_cmp computation flow chart

While there is only one input file to program **spec_cmp**, there are many output files which can easily fill your disk drives. Detailed spectral analysis will typically be done about a reservoir or other zone of interest, such that the input data volume may be windowed. If this is not the case, it is best to first run several of the statistical summary volumes including the peak magnitude, peak frequency, and peak phase, spectral bandwidth, spectral slope, and spectral range-trimmed mean for the entire volume. These spectra may be sufficient for your analysis. If not, the peak frequency volume will serve as a guide as to which spectral components are well tuned.

If the data have not previously been spectrally whitened, it may be beneficial to reconstruct the spectrally balanced amplitude. In either case, it is best to ask for spectrally balanced spectral components and statistical measures, since this will suppress the effect of the dominant frequency of the source wavelet, leaving the tuning effects of the geology in place.

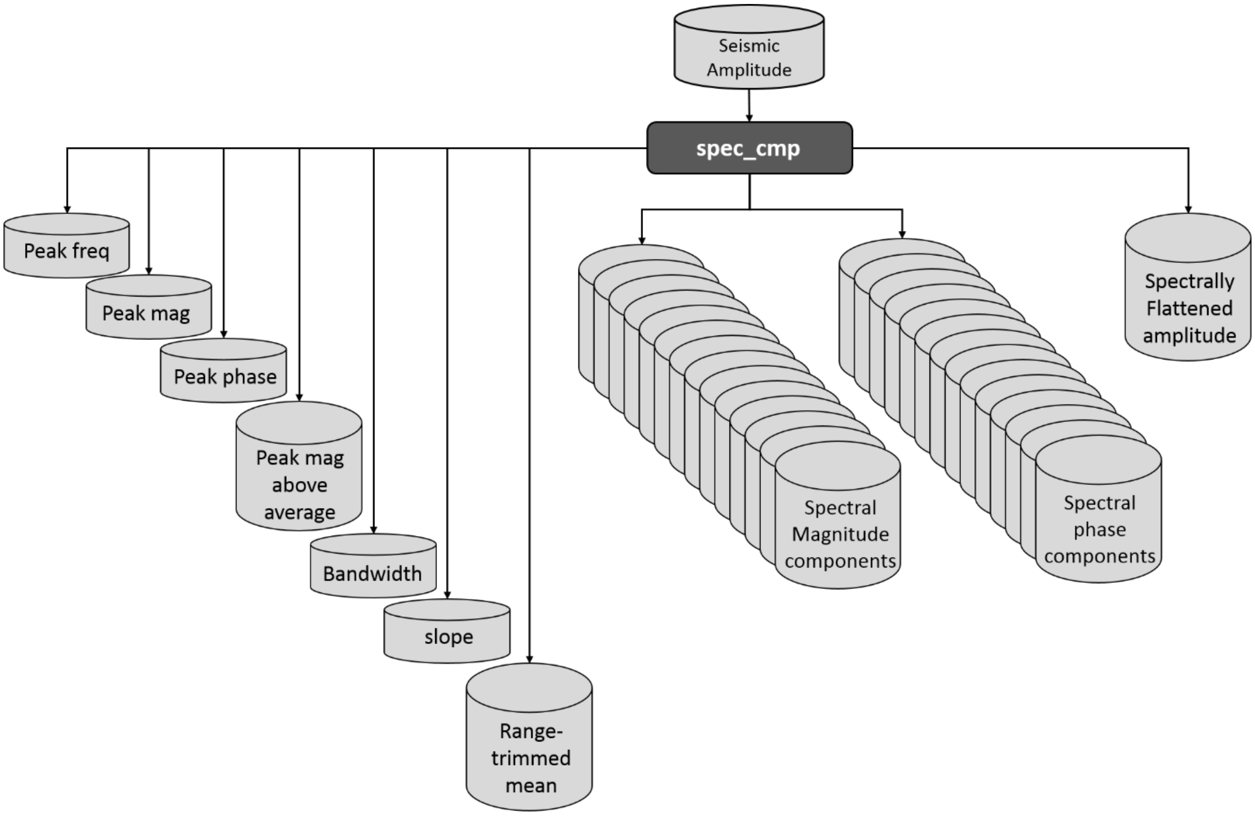


Figure 1: Computation flow chart for program **spec_cmp**, spectral decomposition using a complex matching pursuit algorithm.

## Complex Matching Pursuit Estimate of Spectral Components

Matching pursuit is commonly used in many geophysical processes, including "high resolution" Radon transforms. In its simplest implementation, so-called "greedy matching pursuit", the single model parameter (in **spec_cmp** a single wavelet at a fixed time that best represents the energy of the seismic trace) is least-squares fit to the data, whereby modeled and residual traces are constructed. This process continues until the residual has reached an acceptably low energy. A somewhat more efficient approach is to allow multiple wavelets (in **spec_cmp**, all those events that exceed a given percent of the peak envelope of the residual trace) are fit to the data simultaneously. The known complex spectra of the modeled Ricker or Morlet wavelets are then accumulated, giving rise to a complex time-frequency distribution.

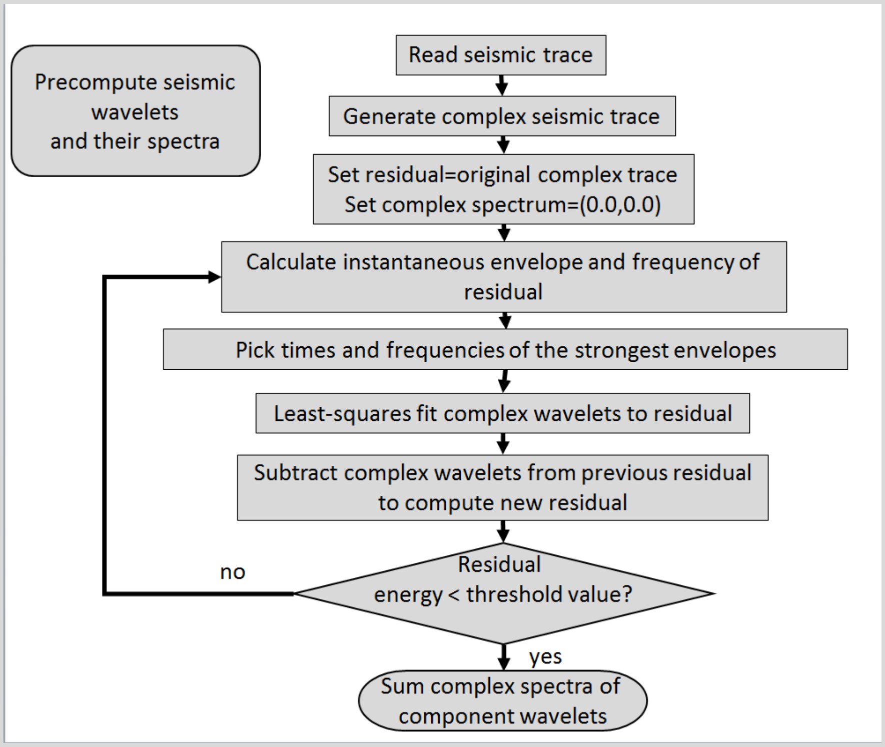

Precompute seismic wavelets and their spectra
Read seismic trace
Generate complex seismic trace
Set residual=original complex trace
Set complex spectrum=(0.0,0.0)
Calculate instantaneous envelope and frequency of residual
Pick times and frequencies of the strongest envelopes
Least-squares fit complex wavelets to residual
Subtract complex wavelets from previous residual to compute new residual
Residual energy < threshold value?
no
yes
Sum complex spectra of component wavelets

Figure A1. The matching pursuit workflow as implemented in program **spec_cmp.** (After Liu and Marfurt, 2005).

## Matching Pursuit Iterations

Matching pursuit models the seismic data iteratively, starting by modeling the most energetic events of the original data, followed by modeling the most energetic events of the residual, until the energy of the residual is acceptably low. The center time of a wavelet (or “atom”) is estimated by the peak of the trace envelope, while its frequency is estimated by the instantaneous frequency. The magnitude and phase of the event are estimated by fitting complex wavelets to a complex (or analytic) trace.

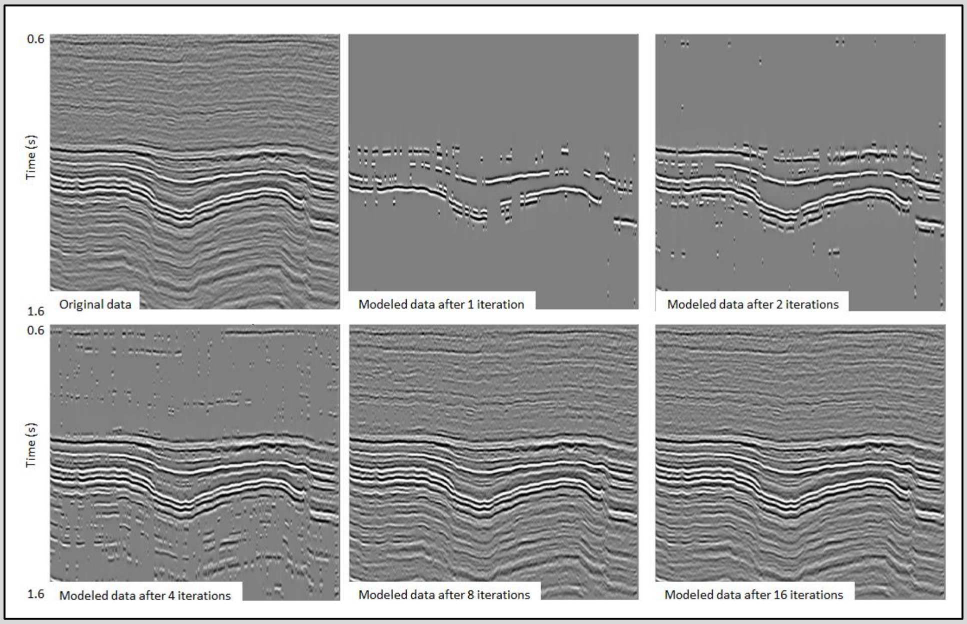


Figure A2. The original data (upper left) and successive approximations using a Ricker wavelet basis function. In this example all events whose envelope exceeded 80% of the envelope of the maximum event on each trace was fit simultaneously to the data. (After Liu and Marfurt, 2005).

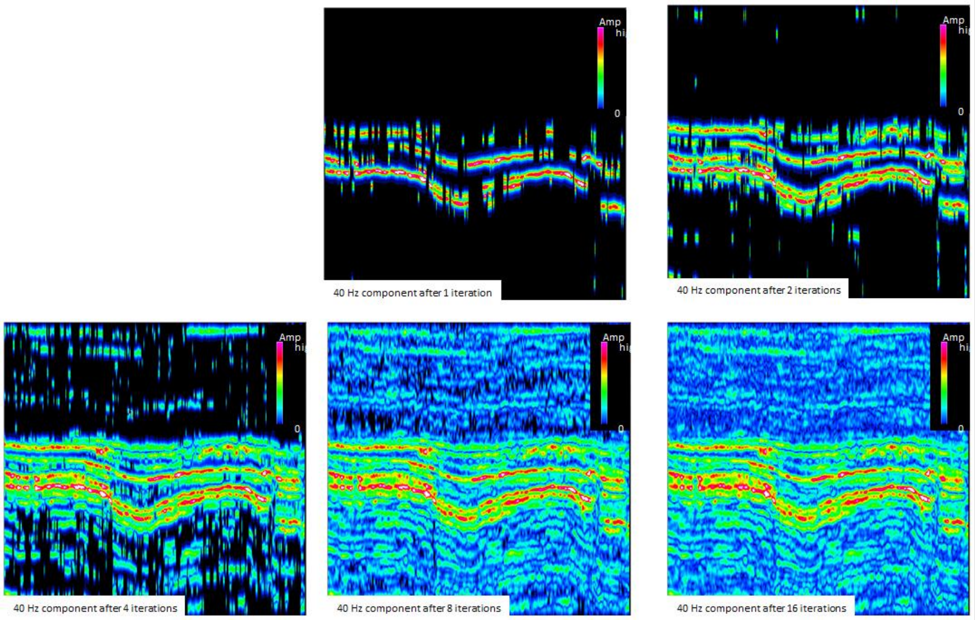
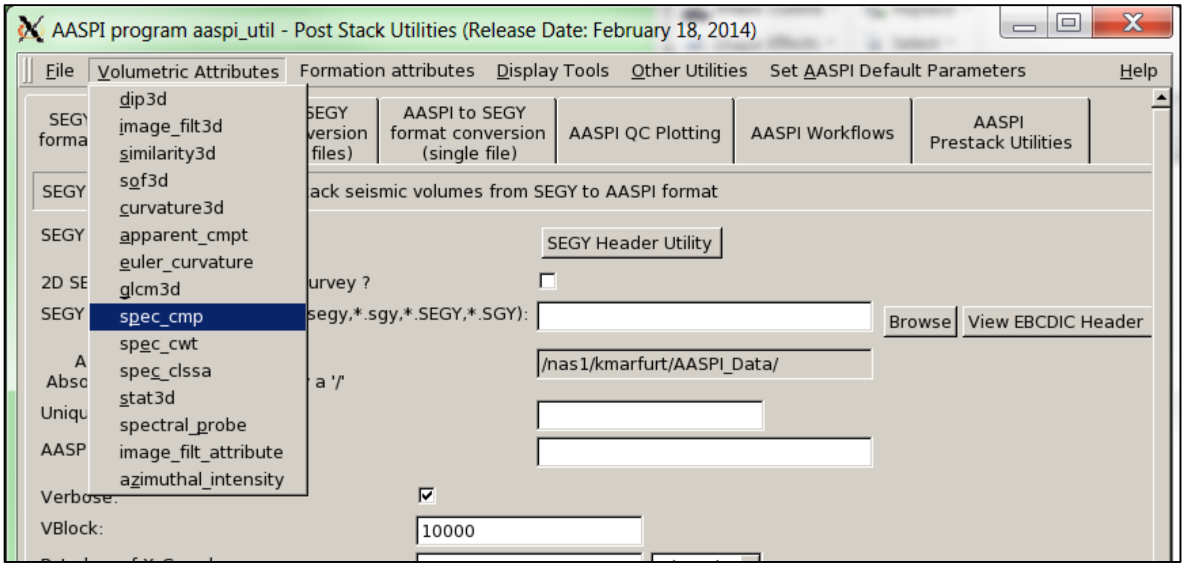


Figure A3. Accumulation of the 40 Hz spectral component corresponding to the iterations shown in Figure A2. Although this display shows the spectral magnitude, it is the complex spectra that are accumulated.

## Computing spectral components

To begin, click the Attributes calculation tab in the AASPI-util window and select program **spec_cmp**:

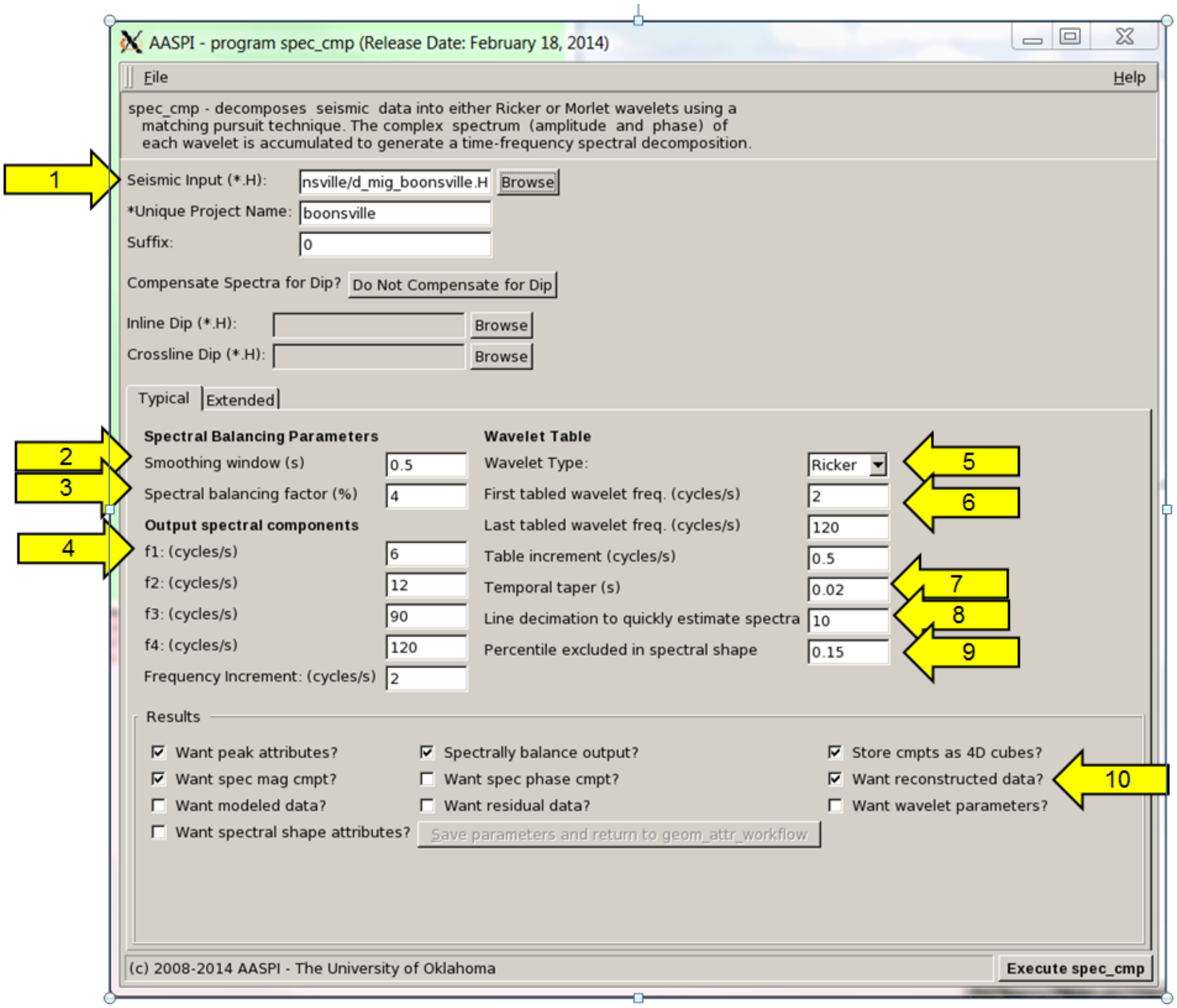


/nas1/kmarfurt/AASPI_Data/

Verbose: ☑

VBlock: 10000

Program **spec_cmp** performs spectral decomposition by least-squares fitting either complex Ricker or Morlet wavelets to the analytic (complex) seismic trace using a matching pursuit method. The following window appears:



AASPI - program spec_cmp (Release Date: February 18, 2014)

spec_cmp - decomposes seismic data into either Ricker or Morlet wavelets using a matching pursuit technique. The complex spectrum (amplitude and phase) of each wavelet is accumulated to generate a time-frequency spectral decomposition.

Seismic Input (*.H): nsville/d_mig_boonsville.H

*Unique Project Name: boonsville

Suffix: 0

Compensate Spectra for Dip? Do Not Compensate for Dip

Inline Dip (*.H):

Crossline Dip (*.H):

**Spectral Balancing Parameters**

Smoothing window (s) 0.5

Spectral balancing factor (%) 4

**Output spectral components**

f1: (cycles/s) 6

f2: (cycles/s) 12

f3: (cycles/s) 90

f4: (cycles/s) 120

Frequency Increment: (cycles/s) 2

**Wavelet Table**

Wavelet Type: Ricker

First tabled wavelet freq. (cycles/s) 2

Last tabled wavelet freq. (cycles/s) 120

Table increment (cycles/s) 0.5

Temporal taper (s) 0.02

Line decimation to quickly estimate spectra 10

Percentile excluded in spectral shape 0.15

Results

☑ Want peak attributes?

☑ Want spec mag cmpt?

☐ Want modeled data?

☐ Want spectral shape attributes?

☑ Spectrally balance output?

☐ Want spec phase cmpt?

☐ Want residual data?

☑ Store cmpts as 4D cubes?

☑ Want reconstructed data?

☐ Want wavelet parameters?

(c) 2008-2014 AASPI - The University of Oklahoma

First, enter the (1) name of the *Seismic Input (*.H)* file you wish to decompose, as well as a *Unique Project Name* and *Suffix* as you have done for other AASPI programs.

## Spectral Balancing

Moving down the panel, we enter (2) a *Smoothing window* = 0.5 s that smoothes the spectra vertically before trying to estimate a spectral balancing operator. The (3) *Pct. for spectral balancing* is set to be 1%. (4% would be a more conservative value and produce data closer to the original.

### The arithmetic of spectral balancing and bluing

Flattened spectra are obtained by balancing the power. The power of the $j^{th}$ trace is simply the spectral magnitude squared:

$$P_j(t,f) = a_j^{\,2}(t,f)\,. \tag{A1}$$

This spectral magnitude is averaged over all traces *j=1,…,J* and a *2K+1* sample vertical analysis window to obtain the average power for each time slice *t*

$$P_{avg}(t,f) = \frac{1}{J(2K+1)} \sum_{k=-K}^{K} \sum_{j=1}^{J} P(t+k\Delta t, f)\,. \tag{A2}$$

The peak of the average power spectrum at time *t* is defined as

$$P_{peak}(t) = \underset{f}{MAX}\left\lfloor P_{avg}(t,f)\right\rfloor. \tag{A3}$$

With these definitions and a prewhitening value of $\varepsilon$=0.02 (2%) the flattened magnitude spectrum is computed as

$$a_j^{flat}(t,f) = \left[\frac{P_{peak}(t)}{P_{avg}(t,f)+\varepsilon P_{peak}(t)}\right]^{1/2} a_j(t,f)\,. \tag{A4}$$

Traditionally, the goal of seismic processing was to produce a flat spectrum. However, Neep (2007) and others built on earlier “colored inversion” work that showed the reflectivity spectrum derived from well logs behaves as $f^{\beta}$ where $0.0 < \beta < 0.4$. A more general spectral bluing filter is then

$$a_j^{blue}(t,f) = \left[\frac{P_{peak}(t)}{P_{avg}(t,f)+\varepsilon P_{peak}(t)}\right]^{1/2} f^{\beta} a_j(t,f)\,. \tag{A5}$$

The *Output spectral components parameters* (4) indicates the four corner frequencies $f_1$, $f_2$, $f_3$, and $f_4$ used to reconstruct the seismic data after flattening. A raised-cosine taper is applied between $f_1$and $f_2$ as well as between $f_3$ and $f_4$. The generated spectral components will range between $f_1$and $f_4$ with a *Frequency increment* defined in this example as 2 cycles/s.

## Spectral Sampling and Data Reconstruction

Unlike the Fast Fourier Transform, most spectral decomposition algorithms, including the three AASPI programs **spec_cmp**, **spec_cwt**, and **spec_clssa** are non-orthogonal transforms. The interpreter is free to either undersample or oversample the seismic spectrum. In the case of oversampling, all three algorithms will accurately reconstruct the original data, and if requested, perform accurate spectrally balanced results. Undersampling is more problematic. In principle, the proper spectral sampling is defined by the analysis window. In the case of a fixed length window, such as the default 40 ms window in program **spec_clssa**, Nyquist sampling tells us that we cannot sample frequencies less than $f_{min}$=1/0.040 s = 25 Hz. The least-squares approximation used in this implementation helps us go below this cutoff. Both **spec_cmp** and **spec_cwt** use variable length windows, with low frequencies having longer windows and high frequencies having shorter windows. We do not know of an exact spectral sampling criterion. Coarse sampling with a value of $\Delta f$=5 Hz or greater may give rise to poor data reconstruction, even though the spectral analysis at those frequencies (the "forward" problem) will still be very accurate. While this may seem to be suspicious, the seismic data in **spec_cmp** are being modeled not by spectral components, but rather by Morlet or Ricker "atoms" or wavelets. These wavelets are then sampled at a user-defined frequency increment, say at 10 Hz. The choice of frequency increment and range does not significantly impact the decomposition time, only the output size of the data if you choose to output spectral magnitude and spectral phase components. Our experience for seismic data in the 5-120 Hz range indicates that a sample increment of $\Delta f$=2 Hz almost always gives good reconstruction results.

### Avoiding Pitfalls in the Decomposition of Flattened Data

One of the more common applications of spectral decomposition is to analyze stratigraphy about an interpreted horizon. The simplest, but perhaps least efficient approach is to compute the spectral components for the entire seismic volume. If you request 50 to 100 spectral components you may quickly fill your disk drive. The statistical estimates of the spectrum such as the peak frequency and peak magnitude help avoid this surplus of attribute volumes.

A second approach is to window the seismic data about a horizon of interest. In general, one would generate a "flattened" subvolume using either your commercial interpretation package or AASPI program **flatten**. Computation of spectral components about a horizon of interest requires a sufficiently large flattened volume. To accurately model a 5 Hz component, one would need a time window of $T=1/f_1==1/5\ Hz=0.2\ s$ , plus a taper that is at least twice this size, or a window of about 0.6 s. If one wishes to evaluate spectral components about a reservoir of finite thickness, the maximum thickness of the reservoir also needs to be added to the data flattening parameters.

The (5) *Wavelet type* can be either *Ricker* or *Morlet*. Ricker wavelets tend to represent seismic data a little better, so convergence may be somewhat quicker even though the resulting spectra will be very similar. The (6) *tabled wavelets* do not impact the computation time. In this case, the wavelets range from 2 cycles/s to 120 cycles/s and are calculated at an increment of 0.5 cycles/s.

A (7) *Temporal taper* is applied to ends of the input seismic data, thereby minimizing potential Gibb's phenomena. If you are applying **spec_cmp** to a flattened data volume, make sure your input data are sufficiently padded to account for both the taper and the lowest frequency wavelet that influences your analysis area.

If the data are to be spectrally balanced we need to first estimate the average magnitude spectrum for the survey. By setting the (8) *Line decimation to estimate* to 10 we compute the average using only every 10th seismic line, thereby increasing the total computation time by only 1/10 or 10%.

Figure 2 provides a representative vertical slice.

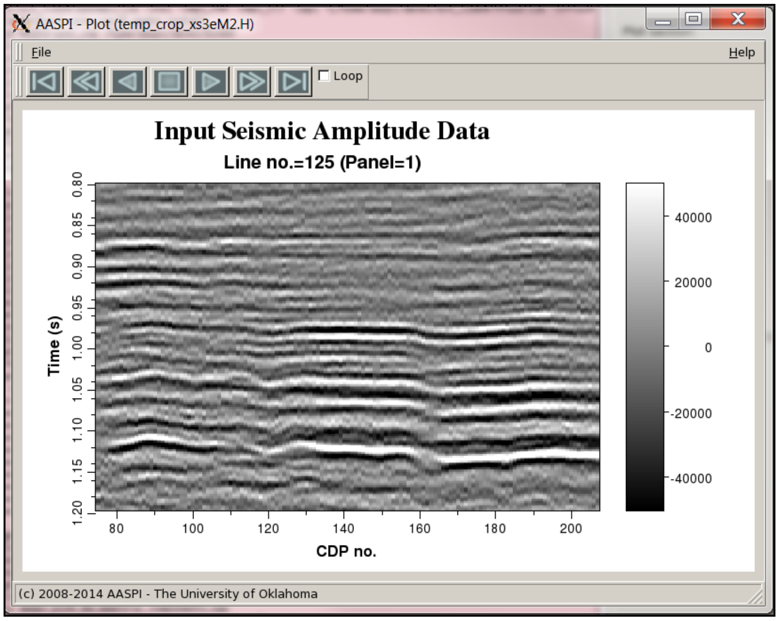


Figure 2: A representative vertical slice through the seismic data.

The modeled data (output only to make sure the program is working correctly) in Figure 3 looks almost identical.

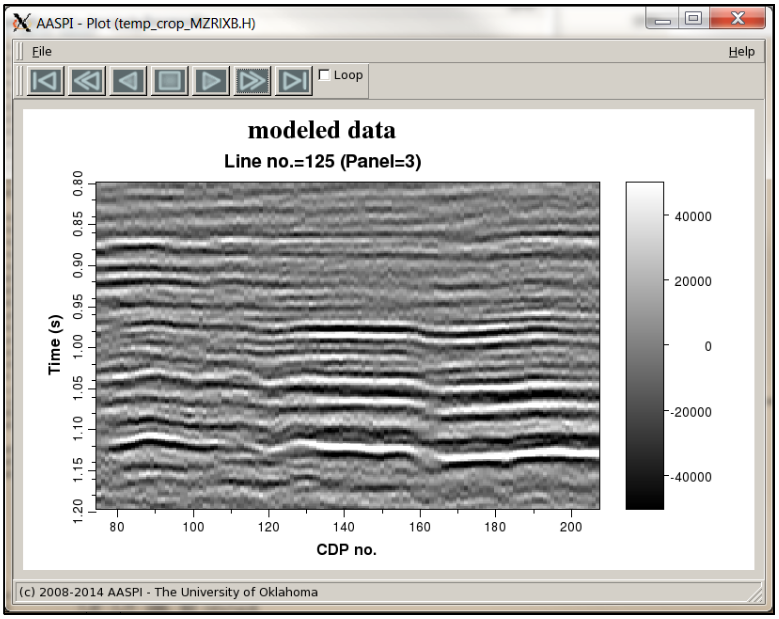


Figure 3: The modeled data from complex matching pursuit algorithm **spec_cmp**. The seismic data for each trace are fit by a suite of wavelets at each iteration, *not* by spectral components.

The residual is the part that is *not* modeled. Figure 4 shows the residual at the same scale as the original vertical slice.

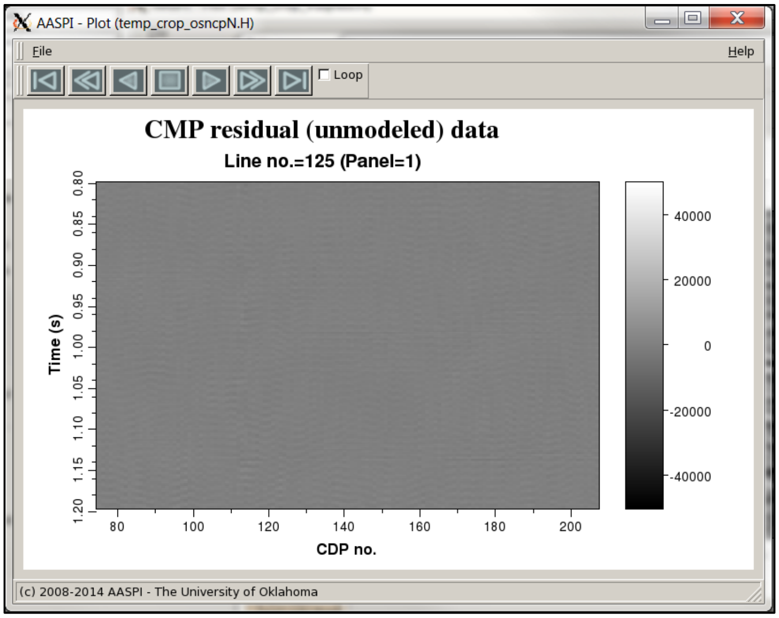


Figure 4: The residual data that was *not* modeled by the matching pursuit algorithm.

Some very low frequency migration artifacts and some very high frequency features outside the desired bandwidth have not been modeled. This result is excellent and shows that all events in the seismic amplitude data have been represented by Ricker or Morlet wavelet, which in turn can be represented by complex spectral components.

This is a 3D survey. Use of the spectral balancing using a percentage of 1% in **spec_cmp** results in the image shown in Figure 5.

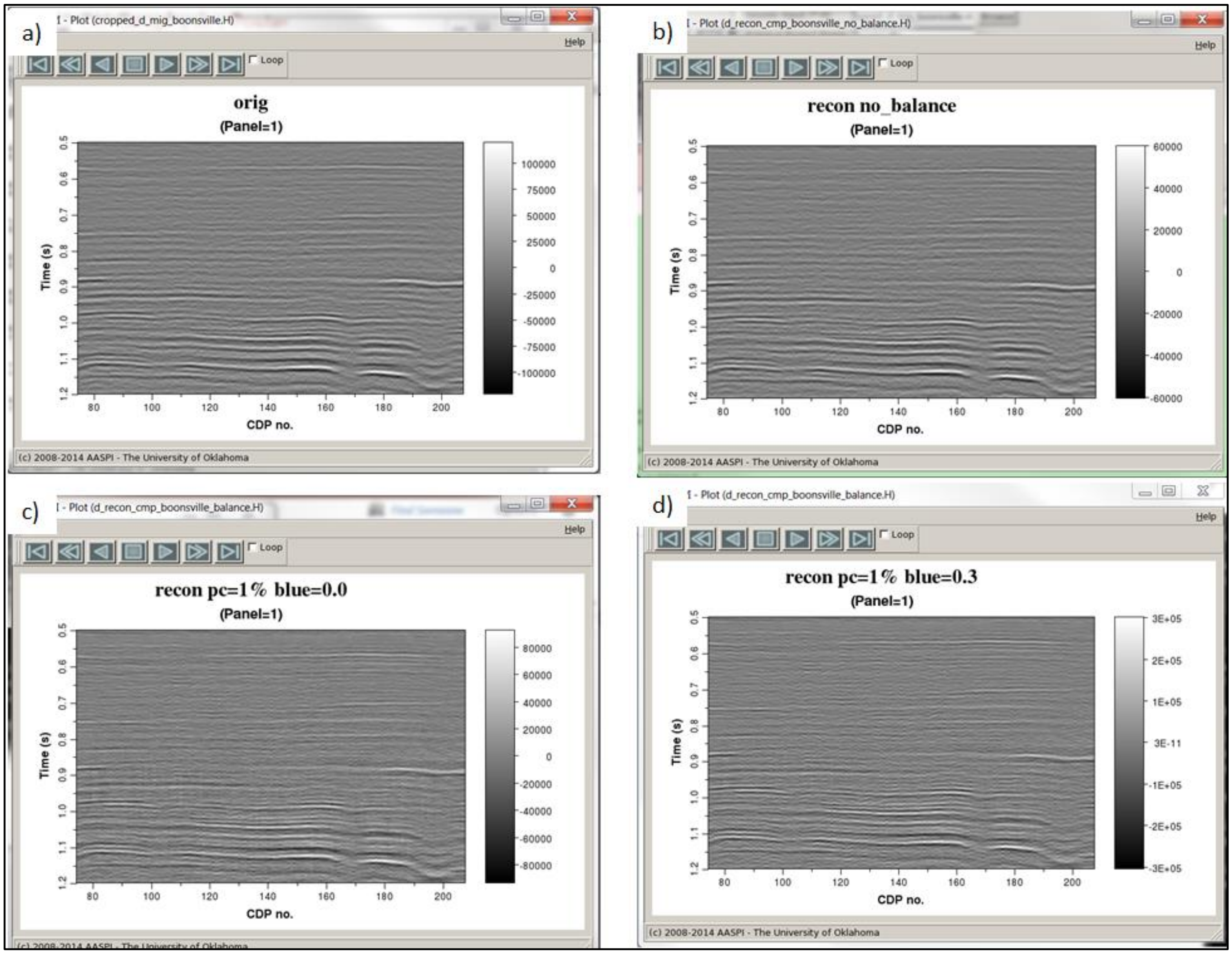


Figure 5: Vertical slices through (a) original data, (b) reconstructed data without spectral balancing, (c) reconstructed data with spectral balancing but no bluing ($\beta$=0.0), and (d) reconstructed data with spectral balancing and bluing ($\beta$=0.3).

The vertical resolution has been improved. In this application the average spectrum is computed from a vertical window over the entire survey. If the reflectivity is random (which usually occurs if there are lateral changes in lithology and dip across the survey within the vertical averaging window) this algorithm provides a stable, time-variant spectral balancing filter where every event at a given two-way travel time has been modified in a consistent manner. There is an alternative spectral-balancing workflow in program **sof3d** which estimates the spectrum of the signal (coherent part of the data) trace by trace in somewhat longer vertical windows.

The averaged spectra before and after spectral balancing can are saved in a file called *avg_spec_power_cmp_boonsville_1_percent.H* for this particular job.

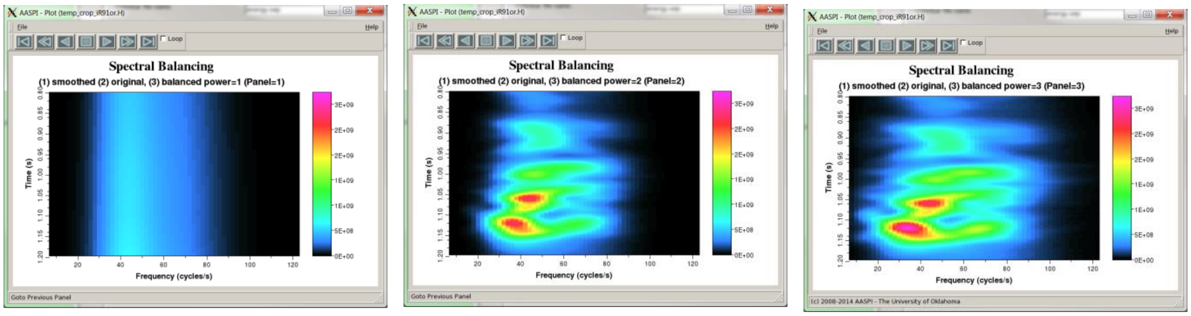

AASPI - Plot (temp_crop_iR91or.H)
File
Help
Loop
Spectral Balancing
(1) smoothed (2) original, (3) balanced power=1 (Panel=1)
(1) smoothed (2) original, (3) balanced power=2 (Panel=2)
(1) smoothed (2) original, (3) balanced power=3 (Panel=3)
Time (s)
0.80
0.85
0.90
0.95
1.00
1.05
1.10
1.15
1.20
Frequency (cycles/s)
20
40
60
80
100
120
3E+09
2E+09
2E+09
1E+09
1E+09
5E+08
0E+00
Goto Previous Panel
(c) 2008-2014 AASPI - The University of Oklahoma

Figure 6: (Center) The average time-frequency spectrum of the entire survey. (Left) The vertically smoothed (±0.5 s) spectrum used to compute the scale factors in equation A4 in the gray box. (Right) the spectrally balanced average spectrum for the entire survey.

The center panel shows the time-variant spectra averaged over the entire survey. The panel on the left vertically averages this spectrum over a *Smoothing window* of +/-0.5 s, which looks near constant in this display where the time axis ranges between 0.8 and 1.2 s. It is the peak value at each time sample, $P_{peak}(t)$, of this image that is used in the equation 4 described in the gray box above. The panel on the right shows the average power spectrum after spectral balancing. Note that is extended to both lower and higher frequencies within the $f_1$, $f_2$, $f_3$, $f_4$ constraints.

A best practice is apply structure-oriented filtering prior to spectral balancing. This workflow is captured in the chapter called Geometric Attribute Workflows.

## Attribute representation of the seismic spectrum

Program **spec_cmp** provides several statistical measures of the spectrum that can be used in addition to or in place of the full 4D spectral components. The peak spectral magnitude, peak spectral frequency, and peak spectral phase are easy to understand. You obtain these by placing a checkmark in front of *Want peak attributes?* A checkmark in front of *Want spectral shape attributes?* will generate the spectral bandwidth, range-trimmed mean spectrum, spectral slope, and spectral roughness attributes described by Zhang et al. (2008) and Zhang (2010). Each of these attributes are computed using a user-defined (8) *Percentile excluded in spectral shape* value (default =0.15) that excludes the tails of the spectra that make up 15% of the energy at either end. In this manner, the spectral bandwidth provides a more accurate estimate for flattened spectra than the more common definition that assumes the spectra have a Gaussian shape. The image graph below from Zhang et al. (2008) describes the spectral bandwidth, slope, and roughness attributes as used in program **spec_cmp.**

## Statistical measures of the spectrum

Gaussian statistics such as the mean, standard deviation, kurtosis, and skewness are sometimes used to represent a seismic spectrum, with the mean representing the average spectrum, the standard deviation the bandwidth, and kurtosis and skewness deviations from the  Gaussian spectrum model. Unfortunately, seismic processors try as hard as they can to make the spectra flat, which is decidedly non-Gaussian. Zhang (2010) therefore constructed a suite of attributes that better define these kinds of spectra. The local bandwidth is defined as the difference between user-defined percentiles. The range-trimmed mean is simply the average frequency within these percentiles. The slope is a measure of how the spectrum changes with frequency – e.g. increasing, flat, or decreasing. Finally, the roughness is a measure of local smoothness of the spectrum.

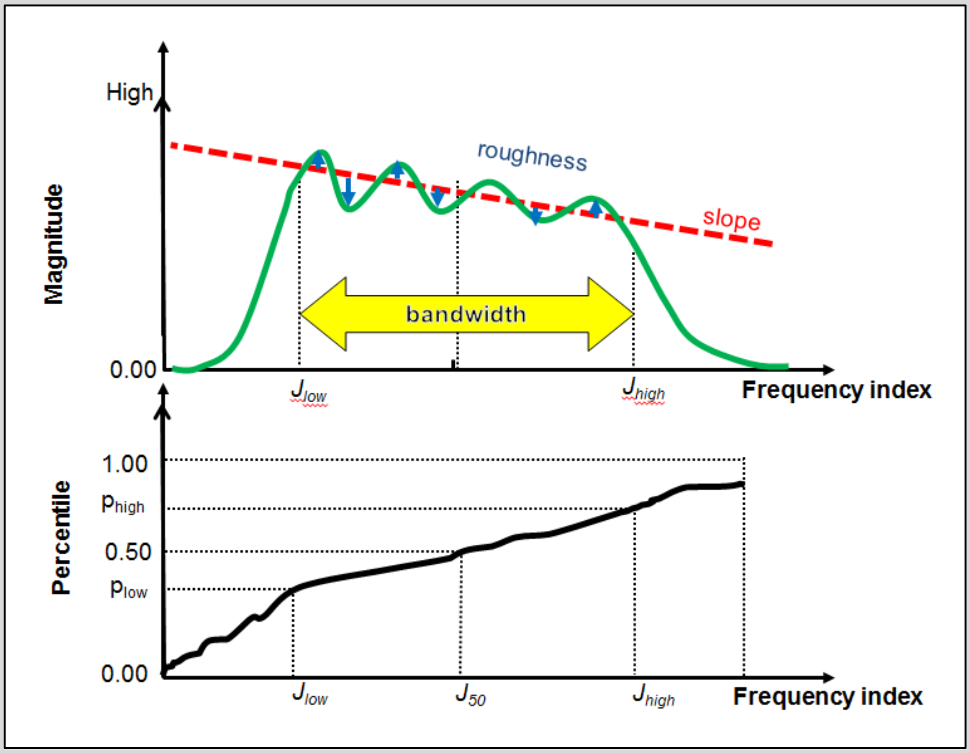
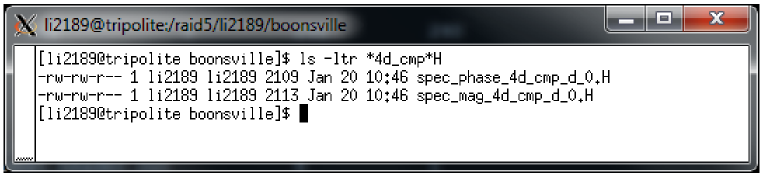


Figure B1. The shape of a spectrum (in green) containing 50 or more components approximated by average frequency,  bandwidth,  slope, and   roughness attributes.

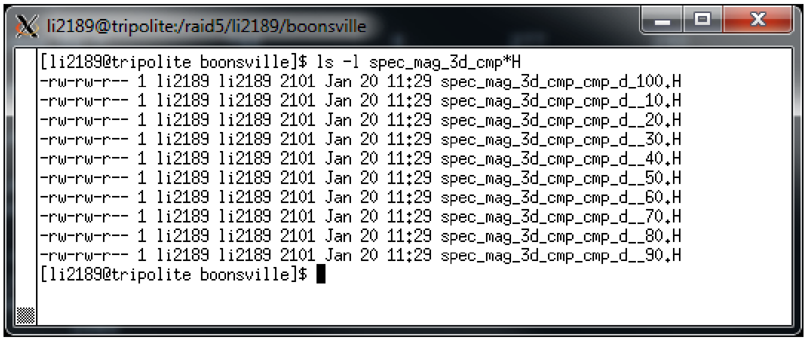
## Individual spectral components

If you place a checkmark in front of *Want spec mag cmpt?* or *Want spec phase cmpt?*, you will obtain each of the spectral components that range between $f_1$ and $f_4$ with the desired *Frequency Increment.* For interpretation of the components on most interpretation workstations, it may be easier to load these components separately. If you place a check mark in front of  (9) *Store cmpts as 4D cubes?,* you obtain spectral gathers that are ordered with the time axis running fastest, followed by the frequency axis, (such that the first two indices represent a time-frequency distribution) followed by the CDP numbers (inline axis) and line numbers (crossline axis).  The 4D volumes will have the following names for this job:



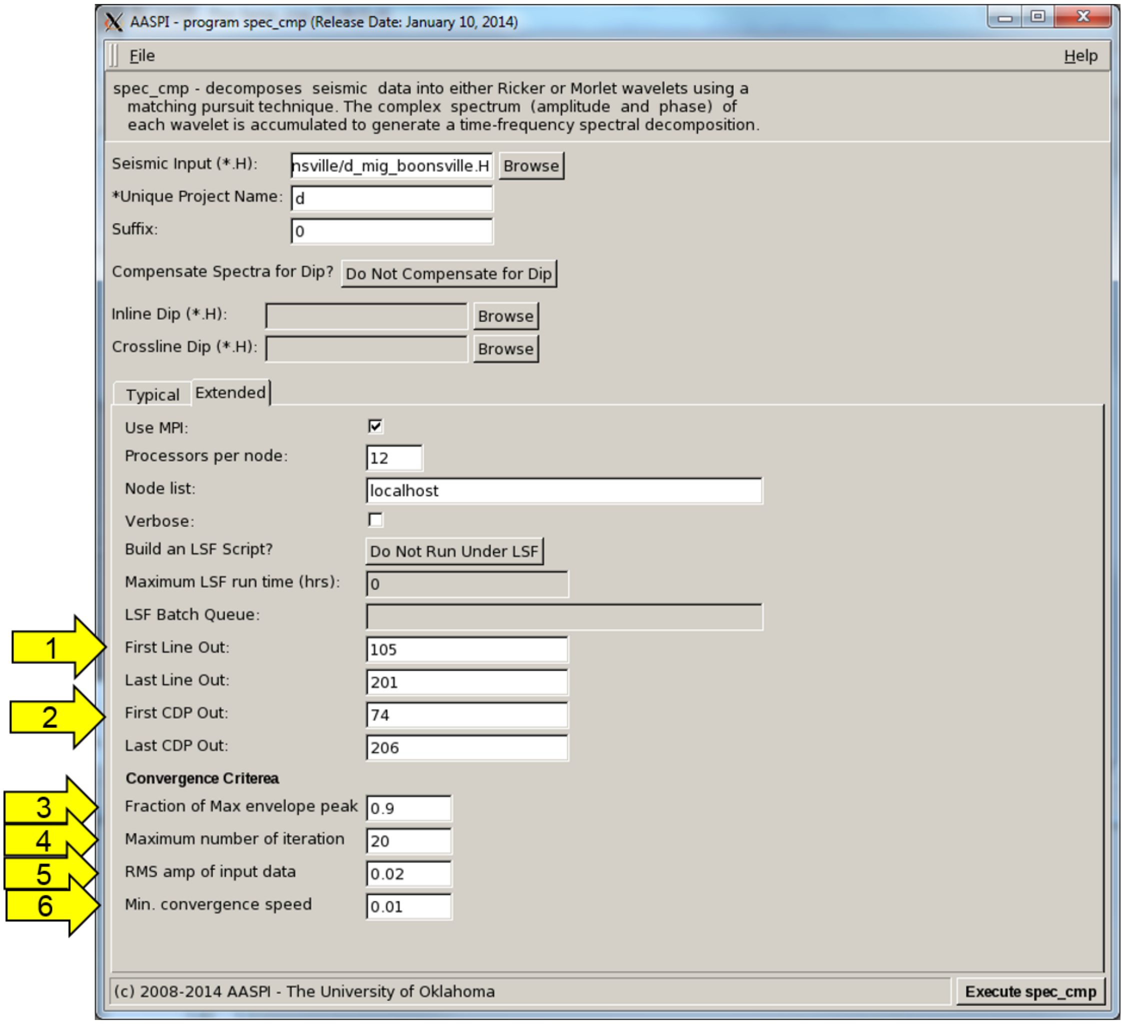
```
[li2189@tripolite boonsville]$ ls -ltr *4d_cmp*H
-rw-rw-r-- 1 li2189 li2189 2109 Jan 20 10:46 spec_phase_4d_cmp_d_0.H
-rw-rw-r-- 1 li2189 li2189 2113 Jan 20 10:46 spec_mag_4d_cmp_d_0.H
[li2189@tripolite boonsville]$ 
```

If you ask for spectral components not to be stored as a 4D cube the constant-frequency 3D spectral magnitude and spectral phase volumes will have the frequency value encoded in the file name:



```
[li2189@tripolite boonsville]$ ls -l spec_mag_3d_cmp*H
-rw-rw-r-- 1 li2189 li2189 2101 Jan 20 11:29 spec_mag_3d_cmp_cmp_d_100.H
-rw-rw-r-- 1 li2189 li2189 2101 Jan 20 11:29 spec_mag_3d_cmp_cmp_d__10.H
-rw-rw-r-- 1 li2189 li2189 2101 Jan 20 11:29 spec_mag_3d_cmp_cmp_d__20.H
-rw-rw-r-- 1 li2189 li2189 2101 Jan 20 11:29 spec_mag_3d_cmp_cmp_d__30.H
-rw-rw-r-- 1 li2189 li2189 2101 Jan 20 11:29 spec_mag_3d_cmp_cmp_d__40.H
-rw-rw-r-- 1 li2189 li2189 2101 Jan 20 11:29 spec_mag_3d_cmp_cmp_d__50.H
-rw-rw-r-- 1 li2189 li2189 2101 Jan 20 11:29 spec_mag_3d_cmp_cmp_d__60.H
-rw-rw-r-- 1 li2189 li2189 2101 Jan 20 11:29 spec_mag_3d_cmp_cmp_d__70.H
-rw-rw-r-- 1 li2189 li2189 2101 Jan 20 11:29 spec_mag_3d_cmp_cmp_d__80.H
-rw-rw-r-- 1 li2189 li2189 2101 Jan 20 11:29 spec_mag_3d_cmp_cmp_d__90.H
[li2189@tripolite boonsville]$ 
```

There are several ‘expert’ controls under the *Extended* tab:



spec_cmp - decomposes seismic data into either Ricker or Morlet wavelets using a matching pursuit technique. The complex spectrum (amplitude and phase) of each wavelet is accumulated to generate a time-frequency spectral decomposition.

Seismic Input (*.H): nsville/d_mig_boonsville.H
*Unique Project Name: d
Suffix: 0
Compensate Spectra for Dip? Do Not Compensate for Dip
Inline Dip (*.H):
Crossline Dip (*.H):

Use MPI: ☑
Processors per node: 12
Node list: localhost
Verbose: □
Build an LSF Script? Do Not Run Under LSF
Maximum LSF run time (hrs): 0
LSF Batch Queue:
First Line Out: 105
Last Line Out: 201
First CDP Out: 74
Last CDP Out: 206

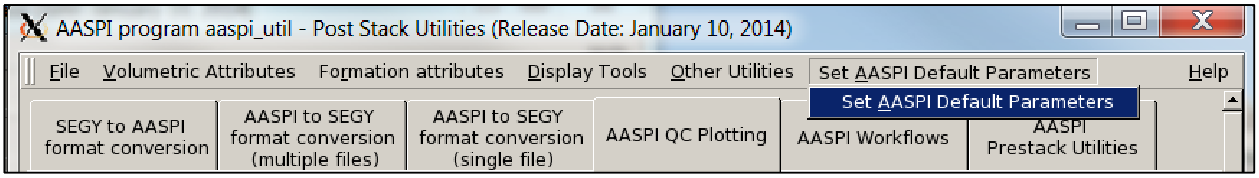
**Convergence Criterea**
Fraction of Max envelope peak 0.9
Maximum number of iteration 20
RMS amp of input data 0.02
Min. convergence speed 0.01

(c) 2008-2014 AASPI - The University of Oklahoma

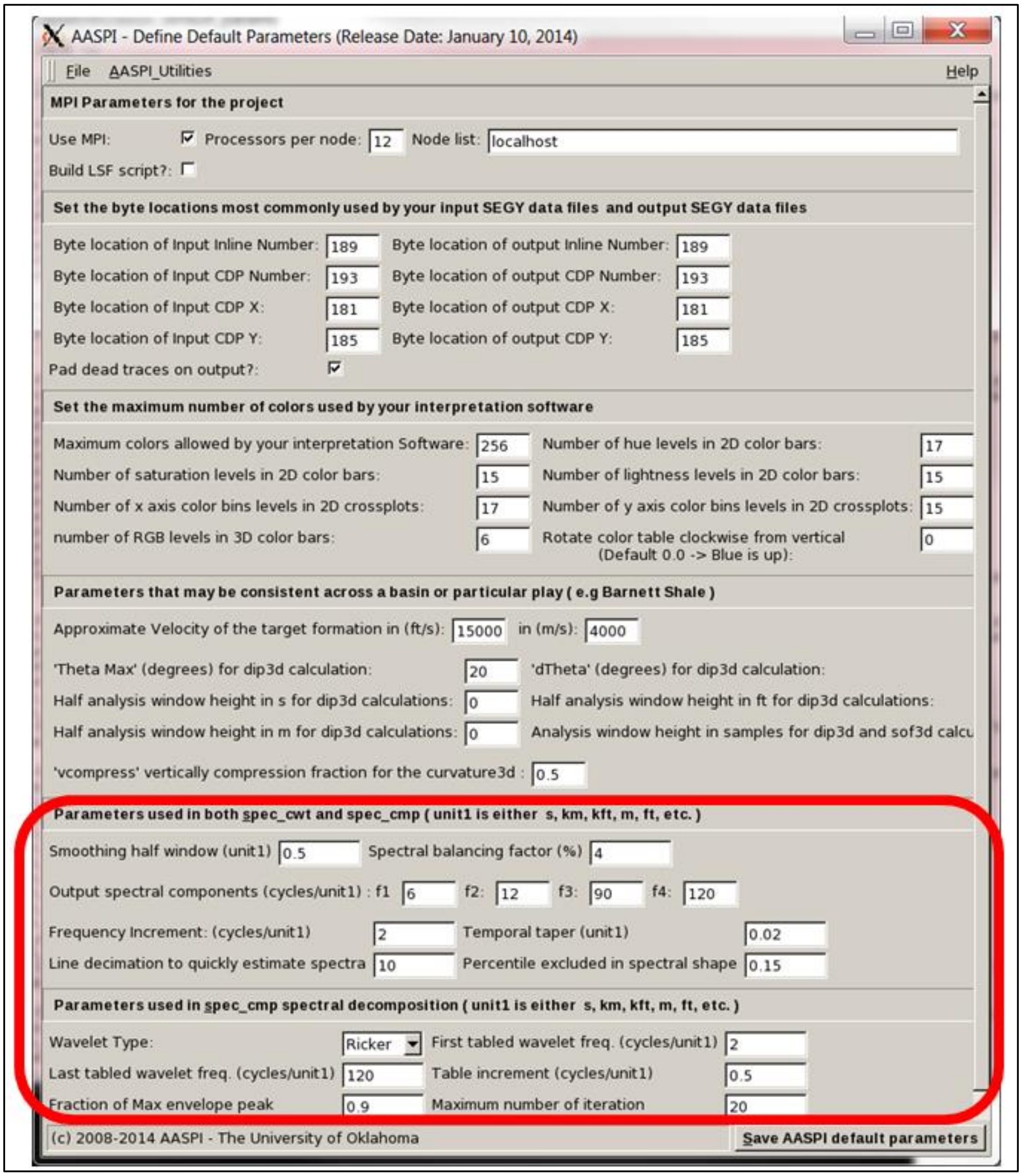
Since the amount of output can be quite large, it may be useful to run **spec_cmp** on only a limited range of (1) inlines and (2) crosslines.

The spectral decomposition is performed using a matching-pursuit algorithm described by Liu and Marfurt (2007).The matching-pursuit algorithm is an iterative process. Before the first iteration, nothing has been done, such that the residual trace is equivalent to the residual trace. At each iteration, the algorithm computes the envelope of the analytic trace and the maximum envelope is detected. Then a subset of all the envelope peaks along the trace are selected that exceed a user-defined (3) *Fraction of Max envelope peak*. Analytic Ricker or Morlet wavelets from a precomputed wavelet library are least-squares fit to the current residual, subtracted, and a new residual is generated. The complex spectral components of these wavelets are multiplied by the phase corresponding to the wavelet time, $t_0$, or exp($i2\pi ft_0$), and accumulated, thereby building up the spectral components of the entire trace. The iteration loop stops either after the (4) *Maximum number of iterations* is reached, if the RMS amplitude of the current residual trace is less than a user-defined fraction (here 0.02) of the (5) *RMS amp of* *input data*, or if the speed in which the RMS decreases between iterations falls below a (6) *Min. convergence speed*. The sensitivity of these values can be evaluated by placing a checkmark in front of *Want modeled data?* And *Want residual data?* on the *Typical* tab and examining the images.

You will probably want to experiment with these parameters a bit to calibrate them for the kind of data you encounter. It is reasonable to expect that surveys of a similar vintage from the same basin will have similar spectra and signal-to-noise ratios. In order to simplify parameter choices, you can use the Set AASPI Default Parameters tab to define parameters you find suitable for spectral decomposition.


AASPI program aaspi_util - Post Stack Utilities (Release Date: January 10, 2014)
File
Volumetric Attributes
Formation attributes
Display Tools
Other Utilities
Set AASPI Default Parameters
Help
Set AASPI Default Parameters
SEGY to AASPI format conversion
AASPI to SEGY format conversion (multiple files)
AASPI to SEGY format conversion (single file)
AASPI QC Plotting
AASPI Workflows
AASPI Prestack Utilities

The default parameters for spec_cmp are found towards the bottom (circled in red):



AASPI - Define Default Parameters (Release Date: January 10, 2014)

**MPI Parameters for the project**

Use MPI: ☑ Processors per node: 12 Node list: localhost

Build LSF script?: □

**Set the byte locations most commonly used by your input SEGY data files and output SEGY data files**

| | | | |
|---|---|---|---|
| Byte location of Input Inline Number: | 189 | Byte location of output Inline Number: | 189 |
| Byte location of Input CDP Number: | 193 | Byte location of output CDP Number: | 193 |
| Byte location of Input CDP X: | 181 | Byte location of output CDP X: | 181 |
| Byte location of Input CDP Y: | 185 | Byte location of output CDP Y: | 185 |
| Pad dead traces on output?: | ☑ | | |

**Set the maximum number of colors used by your interpretation software**

| | | | |
|---|---|---|---|
| Maximum colors allowed by your interpretation Software: | 256 | Number of hue levels in 2D color bars: | 17 |
| Number of saturation levels in 2D color bars: | 15 | Number of lightness levels in 2D color bars: | 15 |
| Number of x axis color bins levels in 2D crossplots: | 17 | Number of y axis color bins levels in 2D crossplots: | 15 |
| number of RGB levels in 3D color bars: | 6 | Rotate color table clockwise from vertical (Default 0.0 -> Blue is up): | 0 |

**Parameters that may be consistent across a basin or particular play ( e.g Barnett Shale )**

Approximate Velocity of the target formation in (ft/s): 15000 in (m/s): 4000

| | | |
|---|---|---|
| 'Theta Max' (degrees) for dip3d calculation: | 20 | 'dTheta' (degrees) for dip3d calculation: |
| Half analysis window height in s for dip3d calculations: | 0 | Half analysis window height in ft for dip3d calculations: |
| Half analysis window height in m for dip3d calculations: | 0 | Analysis window height in samples for dip3d and sof3d calcu |

'vcompress' vertically compression fraction for the curvature3d : 0.5

**Parameters used in both spec_cwt and spec_cmp ( unit1 is either s, km, kft, m, ft, etc. )**

Smoothing half window (unit1) 0.5 Spectral balancing factor (%) 4

Output spectral components (cycles/unit1) : f1 6 f2: 12 f3: 90 f4: 120

| | | | |
|---|---|---|---|
| Frequency Increment: (cycles/unit1) | 2 | Temporal taper (unit1) | 0.02 |
| Line decimation to quickly estimate spectra | 10 | Percentile excluded in spectral shape | 0.15 |

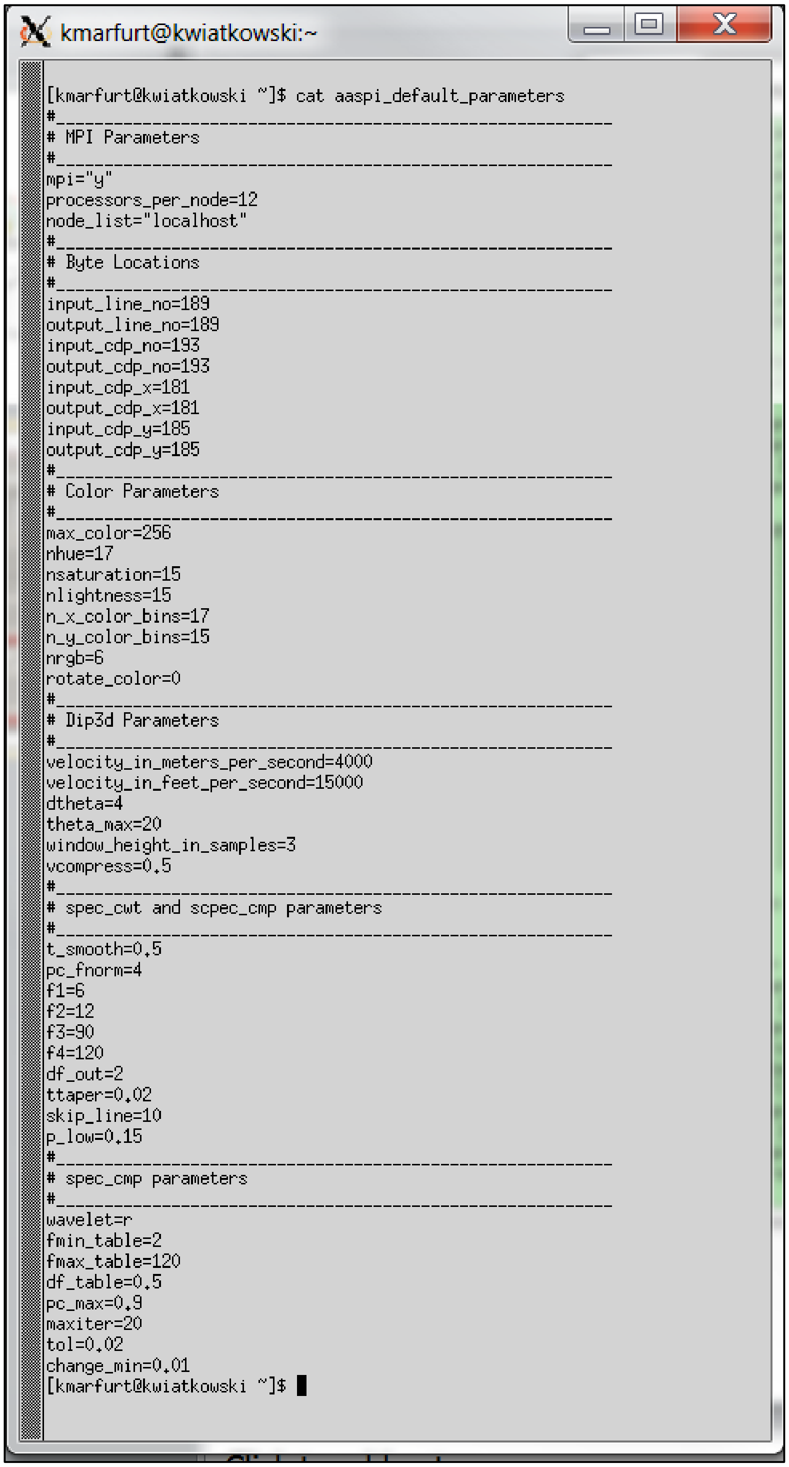
**Parameters used in spec_cmp spectral decomposition ( unit1 is either s, km, kft, m, ft, etc. )**

| | | | |
|---|---|---|---|
| Wavelet Type: | Ricker | First tabled wavelet freq. (cycles/unit1) | 2 |
| Last tabled wavelet freq. (cycles/unit1) | 120 | Table increment (cycles/unit1) | 0.5 |
| Fraction of Max envelope peak | 0.9 | Maximum number of iteration | 20 |

(c) 2008-2014 AASPI - The University of Oklahoma

Save AASPI default parameters

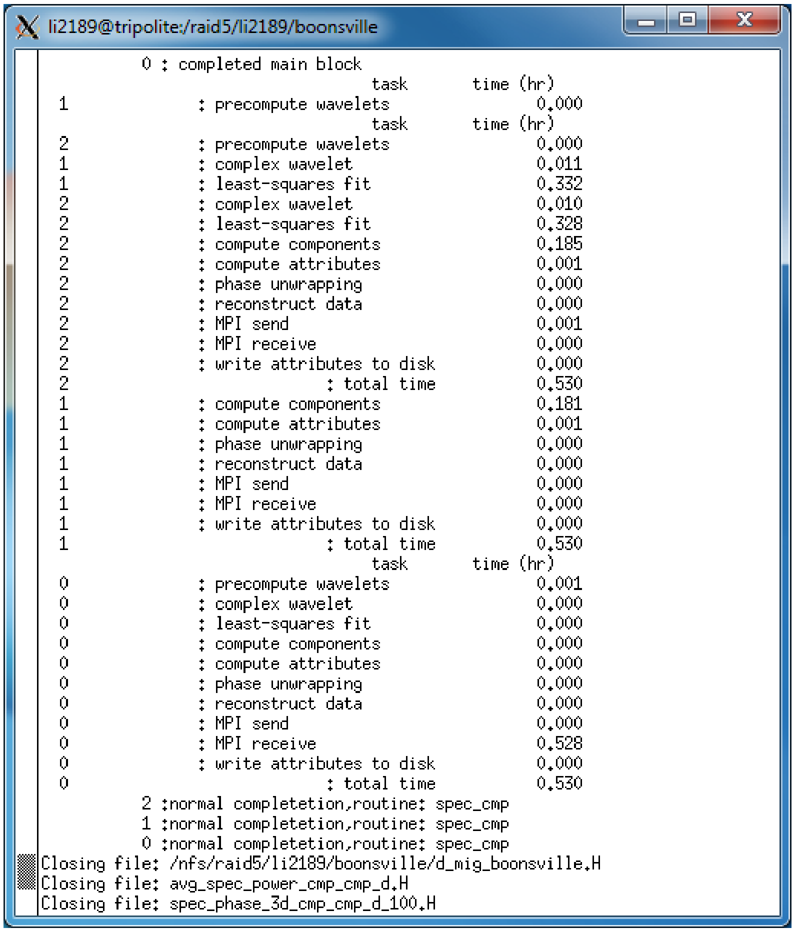
Your system administrator has installed a copy of the default file called ${AASPIHOME}/par/aaspi_default_parameters and perhaps set defaults for your work environment, such as machine names and byte locations for SEGY input and output. If you invoke this GUI from your home directory, modify the parameters, and save it, those parameters will override the defaults for every program you run. If in turn, you invoke this program in a local directory and save it, those parameters will override the defaults in your home directory. In Marfurt's home directory they look like this:



```
[kmarfurt@kwiatkowski ~]$ cat aaspi_default_parameters
#__________________________________________________________
# MPI Parameters
#__________________________________________________________
mpi="y"
processors_per_node=12
node_list="localhost"
#__________________________________________________________
# Byte Locations
#__________________________________________________________
input_line_no=189
output_line_no=189
input_cdp_no=193
output_cdp_no=193
input_cdp_x=181
output_cdp_x=181
input_cdp_y=185
output_cdp_y=185
#__________________________________________________________
# Color Parameters
#__________________________________________________________
max_color=256
nhue=17
nsaturation=15
nlightness=15
n_x_color_bins=17
n_y_color_bins=15
nrgb=6
rotate_color=0
#__________________________________________________________
# Dip3d Parameters
#__________________________________________________________
velocity_in_meters_per_second=4000
velocity_in_feet_per_second=15000
dtheta=4
theta_max=20
window_height_in_samples=3
vcompress=0.5
#__________________________________________________________
# spec_cwt and scpec_cmp parameters
#__________________________________________________________
t_smooth=0.5
pc_fnorm=4
f1=6
f2=12
f3=90
f4=120
df_out=2
ttaper=0.02
skip_line=10
p_low=0.15
#__________________________________________________________
# spec_cmp parameters
#__________________________________________________________
wavelet=r
fmin_table=2
fmax_table=120
df_table=0.5
pc_max=0.9
maxiter=20
tol=0.02
change_min=0.01
[kmarfurt@kwiatkowski ~]$ █
```

The file in your home directory will always take precedence over the one in the ${AASPIHOME}/scripts directory.

As in all the AASPI GUIs, click *Execute* to run the program. The end of your run should looks something like the following:



```
          0 : completed main block
                                  task       time (hr)
  1             : precompute wavelets           0.000
                                  task       time (hr)
  2             : precompute wavelets           0.000
  1             : complex wavelet               0.011
  1             : least-squares fit             0.332
  2             : complex wavelet               0.010
  2             : least-squares fit             0.328
  2             : compute components            0.185
  2             : compute attributes            0.001
  2             : phase unwrapping              0.000
  2             : reconstruct data              0.000
  2             : MPI send                      0.001
  2             : MPI receive                   0.000
  2             : write attributes to disk      0.000
  2                           : total time      0.530
  1             : compute components            0.181
  1             : compute attributes            0.001
  1             : phase unwrapping              0.000
  1             : reconstruct data              0.000
  1             : MPI send                      0.000
  1             : MPI receive                   0.000
  1             : write attributes to disk      0.000
  1                           : total time      0.530
                                  task       time (hr)
  0             : precompute wavelets           0.001
  0             : complex wavelet               0.000
  0             : least-squares fit             0.000
  0             : compute components            0.000
  0             : compute attributes            0.000
  0             : phase unwrapping              0.000
  0             : reconstruct data              0.000
  0             : MPI send                      0.000
  0             : MPI receive                   0.528
  0             : write attributes to disk      0.000
  0                           : total time      0.530
          2 :normal completetion,routine: spec_cmp
          1 :normal completetion,routine: spec_cmp
          0 :normal completetion,routine: spec_cmp
Closing file: /nfs/raid5/li2189/boonsville/d_mig_boonsville.H
Closing file: avg_spec_power_cmp_cmp_d.H
Closing file: spec_phase_3d_cmp_cmp_d_100.H
```

Now, plot some of the results. Since we did not choose to store the spectral magnitude and phase components as a 4D cubes, we have several 3D volumes we can plot separately. Plotting the same time slice as in all the other examples, and setting Allpos=y in our AASPI Viewer GUI for the strictly positive magnitude, the *spec_mag_cmp_boonsville_0_20.H* (the 20 Hz magnitude component) file looks like this:

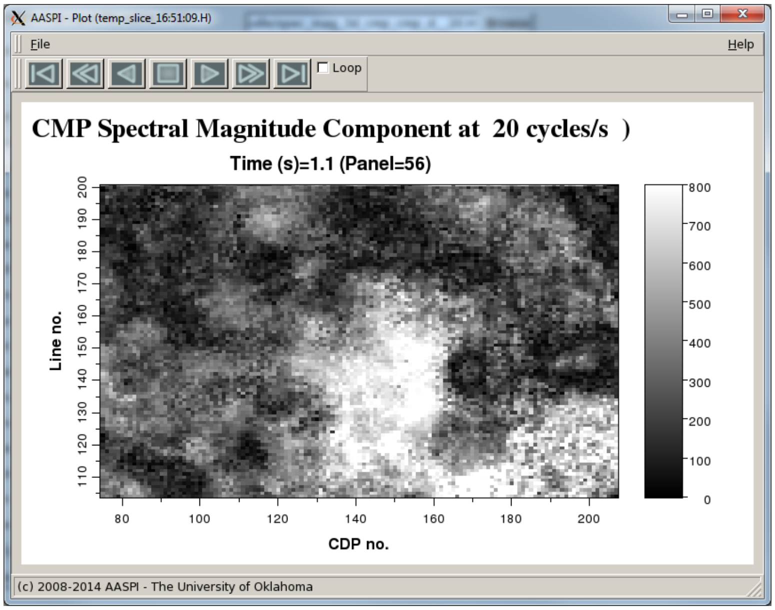

AASPI - Plot (temp_slice_16:51:09.H)
File
Help
Loop
CMP Spectral Magnitude Component at 20 cycles/s )
Time (s)=1.1 (Panel=56)
Line no.
CDP no.
110
120
130
140
150
160
170
180
190
200
80
100
120
140
160
180
200
800
700
600
500
400
300
200
100
0
(c) 2008-2014 AASPI - The University of Oklahoma

Figure 7. Time slice at t=1.1 s through the 20 Hz spectral magnitude component.

While the *spec_mag_cmp_boonsville_0_50.H* (the 50 Hz magnitude component) file looks like this:

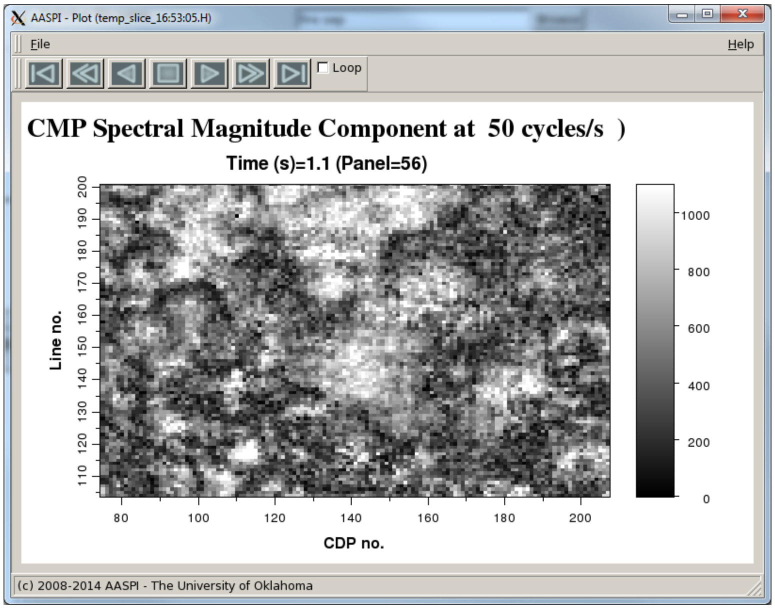

AASPI - Plot (temp_slice_16:53:05.H)
File
Help
Loop
CMP Spectral Magnitude Component at 50 cycles/s )
Time (s)=1.1 (Panel=56)
Line no.
CDP no.
110
120
130
140
150
160
170
180
190
200
80
100
120
140
160
180
200
1000
800
600
400
200
0
(c) 2008-2014 AASPI - The University of Oklahoma

Figure 8. Time slice at t=1.1 s through the 50 Hz spectral magnitude component.

The phase components will range from -180$^0$ to +180$^0$, so set *Fixed-scale* vs. *Auto-scale* and choose a cyclical color bar to plot *spec_phase_cmp_boonsville_0_20.H* (the 20 Hz phase component):

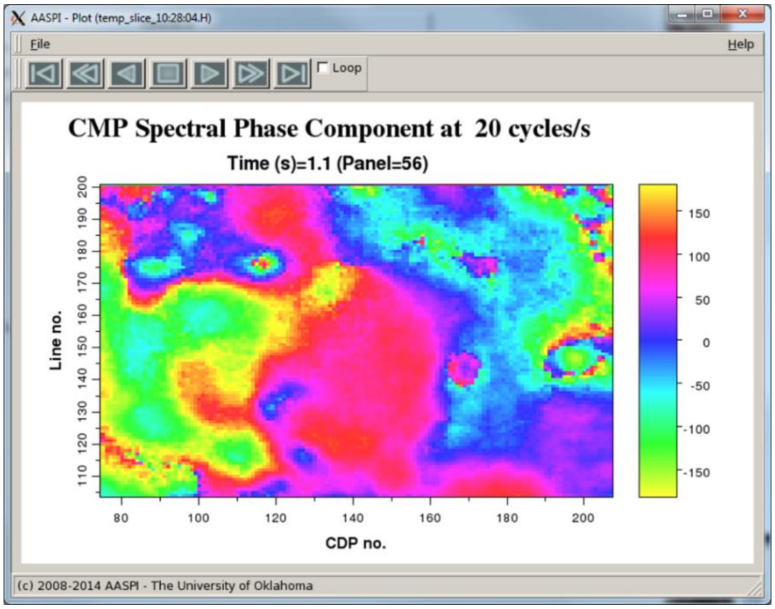

AASPI - Plot (temp_slice_10:28:04.H)
File
Help
Loop
CMP Spectral Phase Component at 20 cycles/s
Time (s)=1.1 (Panel=56)
Line no.
110
120
130
140
150
160
170
180
190
200
80
100
120
140
160
180
200
CDP no.
150
100
50
0
-50
-100
-150
(c) 2008-2014 AASPI - The University of Oklahoma

Figure 9: Time slice at t=1.1 s through the 50 Hz spectral phase component.

and *spec_phase_cmp_boonsville_0_50.H* (the 50 Hz phase component):

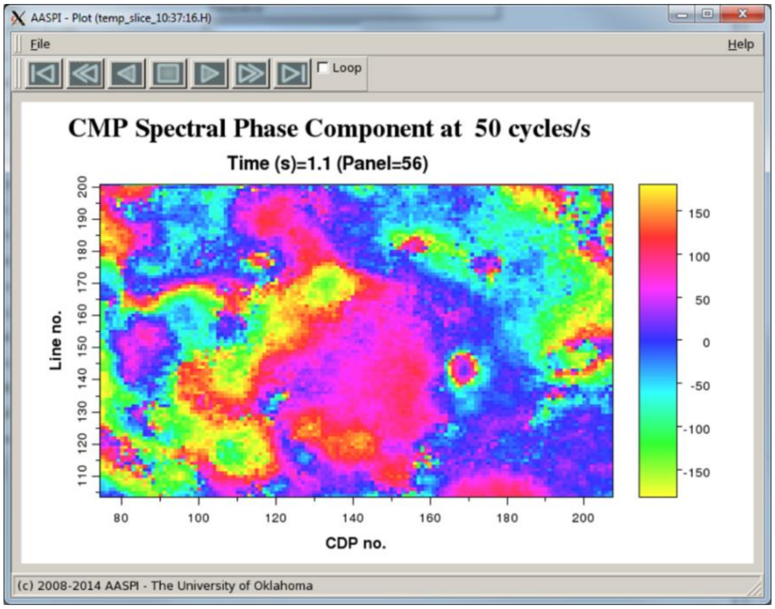

AASPI - Plot (temp_slice_10:37:16.H)
File
Help
Loop
CMP Spectral Phase Component at 50 cycles/s
Time (s)=1.1 (Panel=56)
Line no.
110
120
130
140
150
160
170
180
190
200
80
100
120
140
160
180
200
CDP no.
150
100
50
0
-50
-100
-150
(c) 2008-2014 AASPI - The University of Oklahoma

Figure 10: Time slice at t=1.1 s through the 50 Hz spectral phase component.

Most interpreters are familiar with the instantaneous phase attribute introduced by Taner et al. (1979) and available on all interpretation workstation software. The instantaneous phase is a local measure of the reflectivity response about a given time sample and does not include the phase shift associated with the delay from time t=0.0 . Program spec_cmp provides a similar measure (for each frequency component) by subtracting the phase delay φ=2πft at each sample, giving vertical images that look like the following:

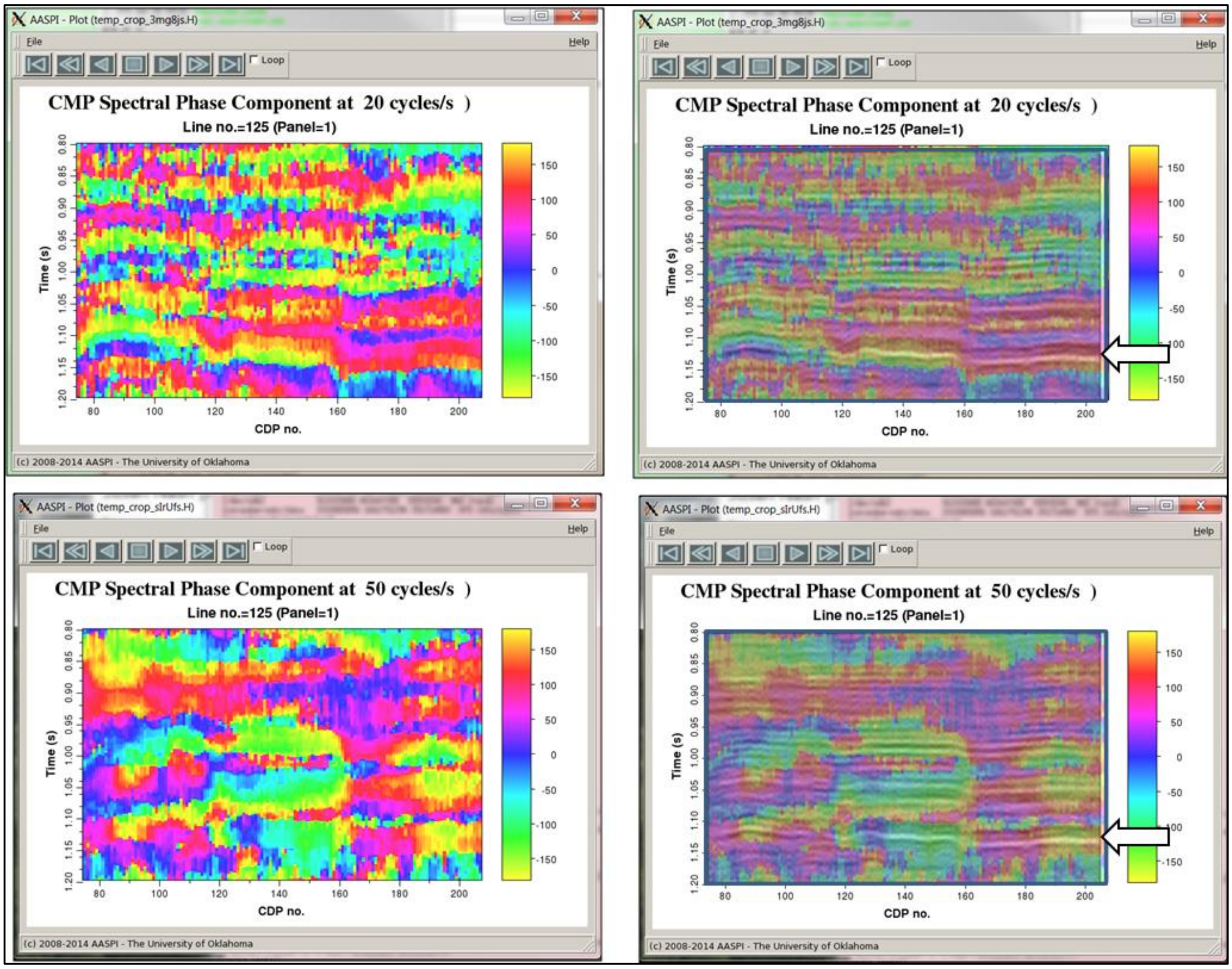

AASPI - Plot (temp_crop_3mg8js.H)
File
Help
Loop
CMP Spectral Phase Component at 20 cycles/s )
Line no.=125 (Panel=1)
Time (s)
0.80
0.85
0.90
0.95
1.00
1.05
1.10
1.15
1.20
CDP no.
80
100
120
140
160
180
200
150
100
50
0
-50
-100
-150
(c) 2008-2014 AASPI - The University of Oklahoma
AASPI - Plot (temp_crop_sIrUfs.H)
CMP Spectral Phase Component at 50 cycles/s )

Figure 11. Vertical slices through the (top) 20 Hz and (bottom) 50 Hz spectral phase component. The spectral phase is co-rendered with the seismic amplitude on the right.

Note that the vertical variation of the 20 Hz and 50 Hz images is comparable, and represents the geology rather than the fact the phase at 50 Hz rotates 2.5 times faster than the phase at 20 Hz. The two right hand images show the phase components co-rendered with the (spectrally balanced) seismic amplitude. White arrows show that the phase varies smoothly with geology, not with time dip. The phase shift of *φ=2πft* needs to be added back to these phase components prior to data reconstruction (with or without spectral balancing).

Several statistical measures of the complex spectrum at each time sample are also computed. The simplest one is the peak magnitude (the greatest value of the spectrum) here as the file *peak_magnitude_cmp_boonsville_0.H*:

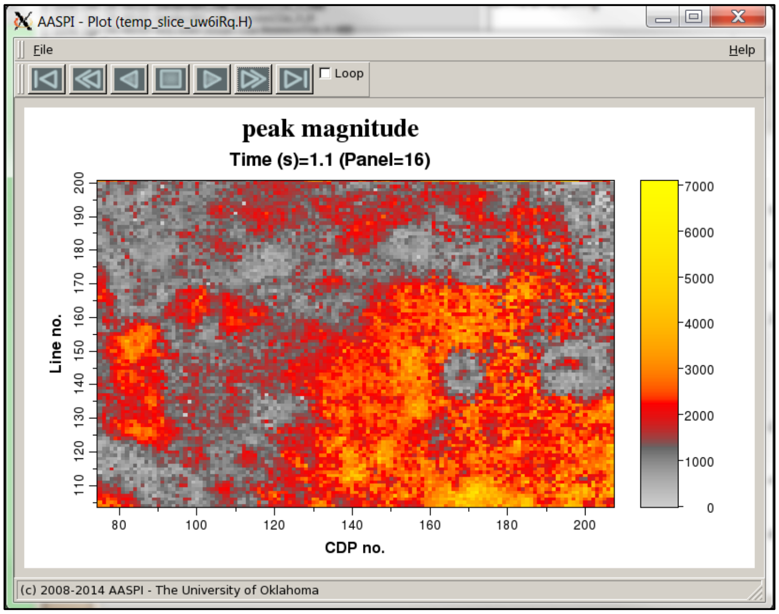


Figure 12: Time slice at t=1.1 s through the peak spectral magnitude attribute.

We can also plot the corresponding frequency at this value (the peak frequency) which in this example is called *peak_freq_cmp_boonsville_0.H.* Using the *frequency.sep* (magenta-red-yellow-green_cyan_blue) color bar gives:

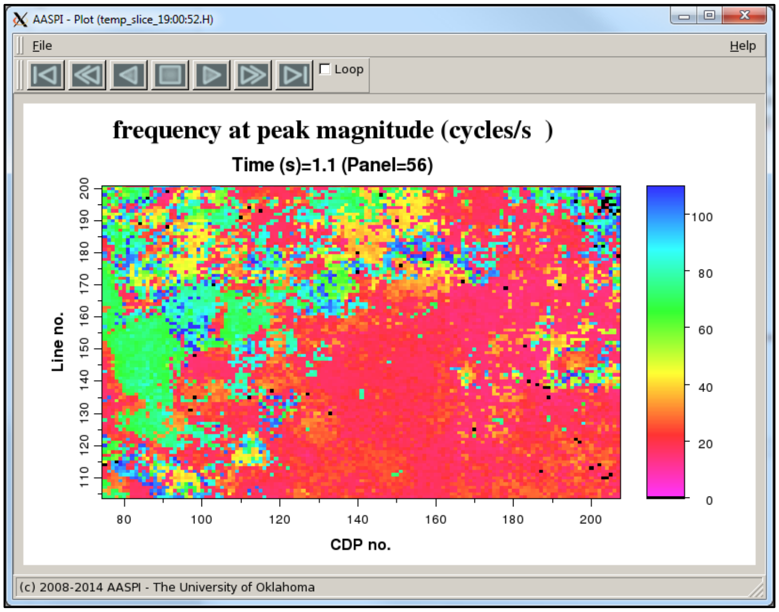
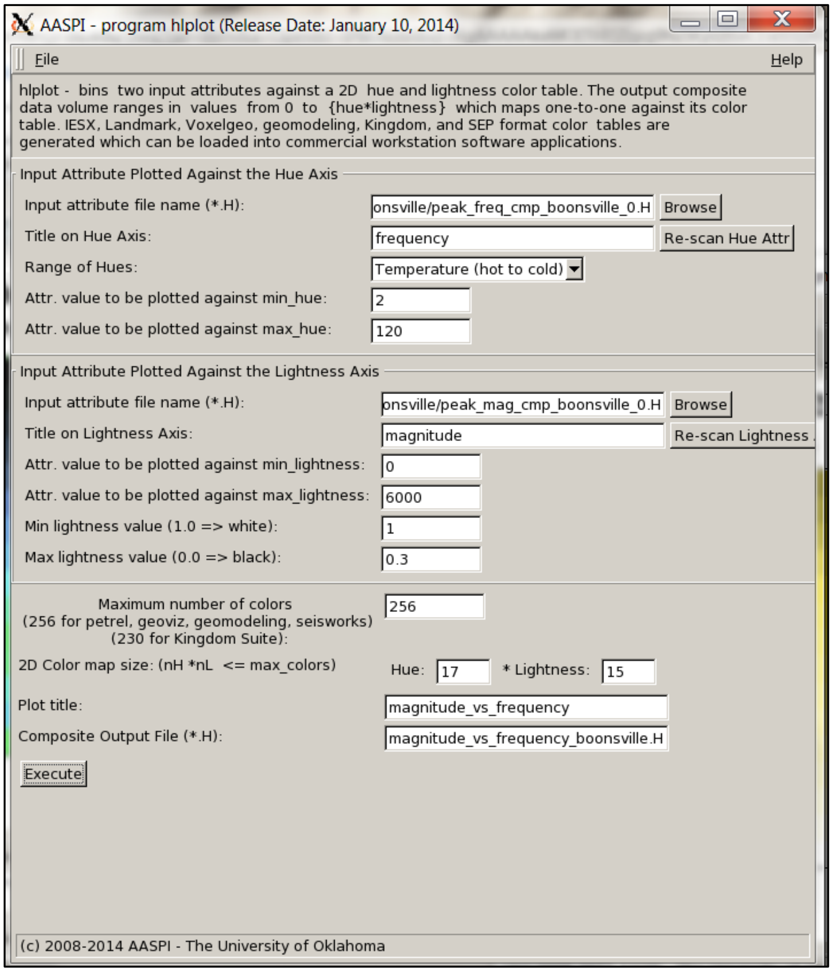

AASPI - Plot (temp_slice_19:00:52.H)
File
Help
Loop
frequency at peak magnitude (cycles/s )
Time (s)=1.1 (Panel=56)
Line no.
110
120
130
140
150
160
170
180
190
200
80
100
120
140
160
180
200
CDP no.
100
80
60
40
20
0
(c) 2008-2014 AASPI - The University of Oklahoma

Figure 13. Time slice at t=1.1 s through the peak spectral frequency attribute.

The red colors correspond to a low peak frequency of about 30 Hz in the SE half of this time slice. Since the value of the peak frequency is almost meaningless if the peak magnitude is close to zero, we can modulate the peak frequency by the peak magnitude using program **hlplot** described earlier. The GUI for **hlplot** looks like this:



AASPI - program hlplot (Release Date: January 10, 2014)

hlplot - bins two input attributes against a 2D hue and lightness color table. The output composite data volume ranges in values from 0 to {hue*lightness} which maps one-to-one against its color table. IESX, Landmark, Voxelgeo, geomodeling, Kingdom, and SEP format color tables are generated which can be loaded into commercial workstation software applications.

Input Attribute Plotted Against the Hue Axis

Input attribute file name (*.H): onsville/peak_freq_cmp_boonsville_0.H

Title on Hue Axis: frequency

Range of Hues: Temperature (hot to cold)

Attr. value to be plotted against min_hue: 2

Attr. value to be plotted against max_hue: 120

Input Attribute Plotted Against the Lightness Axis

Input attribute file name (*.H): onsville/peak_mag_cmp_boonsville_0.H

Title on Lightness Axis: magnitude

Attr. value to be plotted against min_lightness: 0

Attr. value to be plotted against max_lightness: 6000

Min lightness value (1.0 => white): 1

Max lightness value (0.0 => black): 0.3

Maximum number of colors
(256 for petrel, geoviz, geomodeling, seisworks)
(230 for Kingdom Suite): 256

2D Color map size: (nH *nL <= max_colors) Hue: 17 * Lightness: 15

Plot title: magnitude_vs_frequency

Composite Output File (*.H): magnitude_vs_frequency_boonsville.H

(c) 2008-2014 AASPI - The University of Oklahoma

*where the Attr. Against Hue (*.H )* is the peak frequency *peak_freq_cmp_boonsville_0.H* and Hue color bar is *Temperature (hot to cold)*. I've set the *Attr. value to be plotted against min_*hue to 0 Hz and the *Attr. value to be plotted against max_hue* to be 120 Hz. I choose that *Attr. Against Lightness (*.H*) to be the peak magnitude *peak_mag_cmp_boonsville_0.H.* I've set its range to vary between 0 and 6000. The resulting 2D color bar and histogram should look like this:

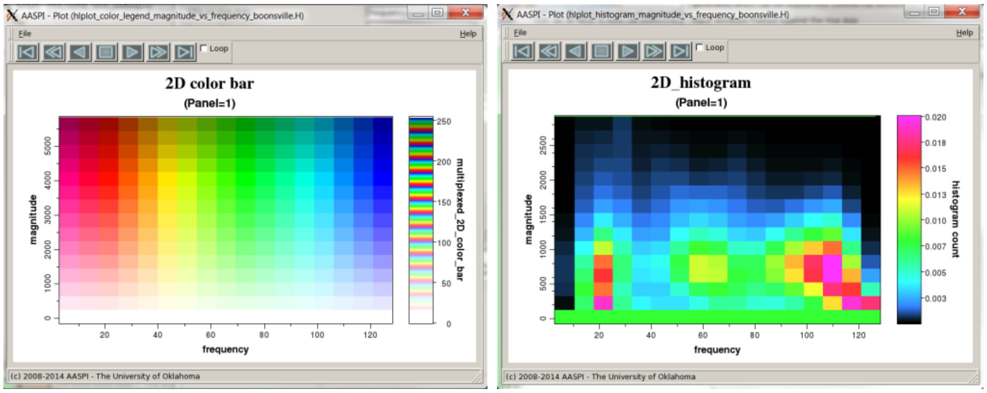


Figure 14. (Left) 2D color bar and (right) 2D histogram of peak frequency modulated by peak magnitude generated using program **hlplot.**

While the multi-attribute display looks like this:

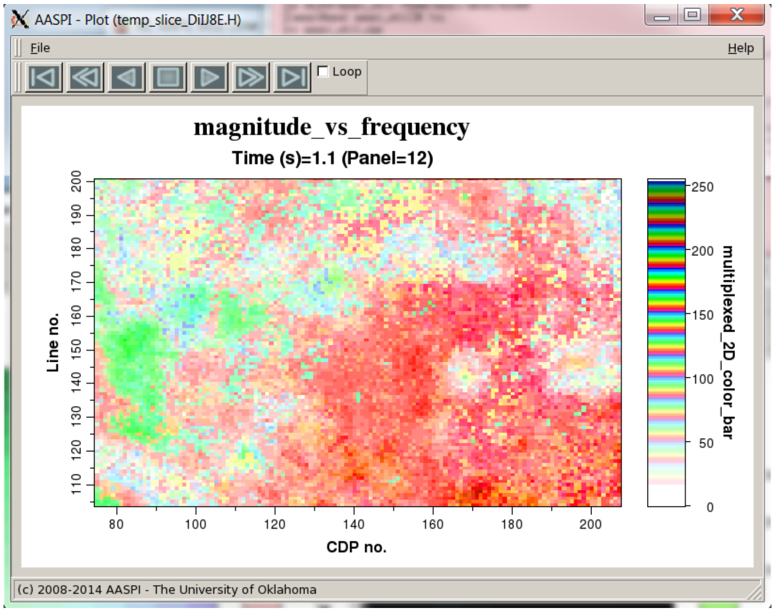
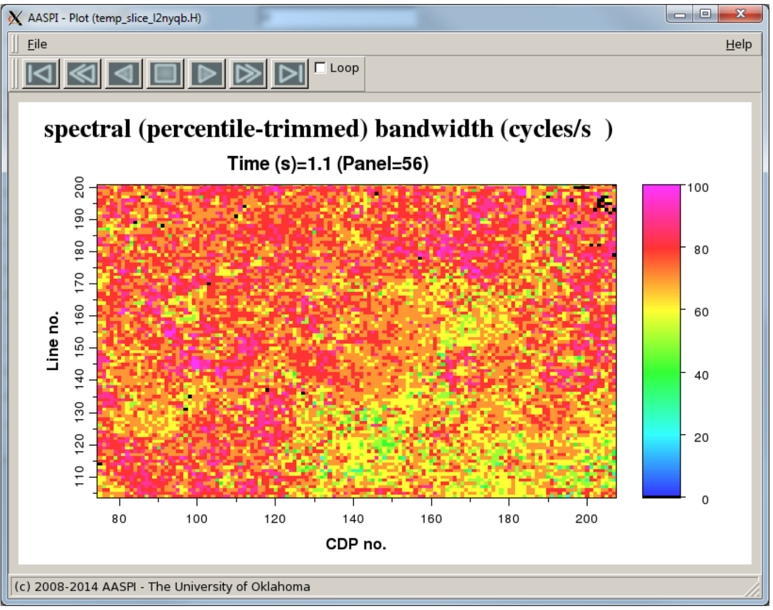


Figure 15: Time slice at t=1.1 s through a composite volume obtained by plotting peak frequency vs. hue and peak magnitude vs. lightness using program **hlplot.** The use of pastel color facilitates subsequent co-rendering with an edge attribute such as coherence.

In this example, the red colors correlate to a peak magnitude at a low frequency.

The spectral bandwidth (the separation in Hz between the 15th and 85th percentile of the magnitude spectrum) is plotted below using the *hue.sep* color bar. Most areas have a broad bandwidth of about 80 Hz (plotted as red).


AASPI - Plot (temp_slice_l2nyqb.H)
File
Help
Loop
spectral (percentile-trimmed) bandwidth (cycles/s )
Time (s)=1.1 (Panel=56)
Line no.
110
120
130
140
150
160
170
180
190
200
80
100
120
140
160
180
200
CDP no.
100
80
60
40
20
0
(c) 2008-2014 AASPI - The University of Oklahoma

4

Figure 16. Time slice at t=1.1 s through the spectral bandwidth volume. In this Barnett Shale slice we don't see any anomalous tuning or attenuation.

The average magnitude that falls between the 15th and 85th percentile (within the bandwidth) of the spectral magnitude, or range-trimmed mean, is *rtm_mag_cmp_0.H* and looks like the following:

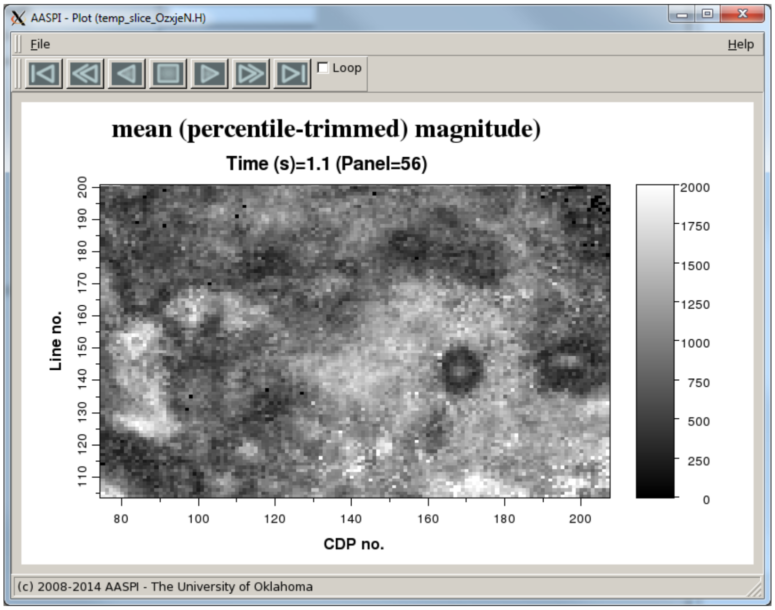


Figure 17: Time slice at t=1.1 through the mean magnitude volume.

When looking at channel and fans, it is often useful to look at the amount of tuning above the average spectrum. **spec_cmp** generates this by subtracting the range-trimmed-mean spectrum from the peak magnitude to obtain the peak magnitude above average. The *peak_mag_above_avg_cmp_0.H* file looks like this:

4

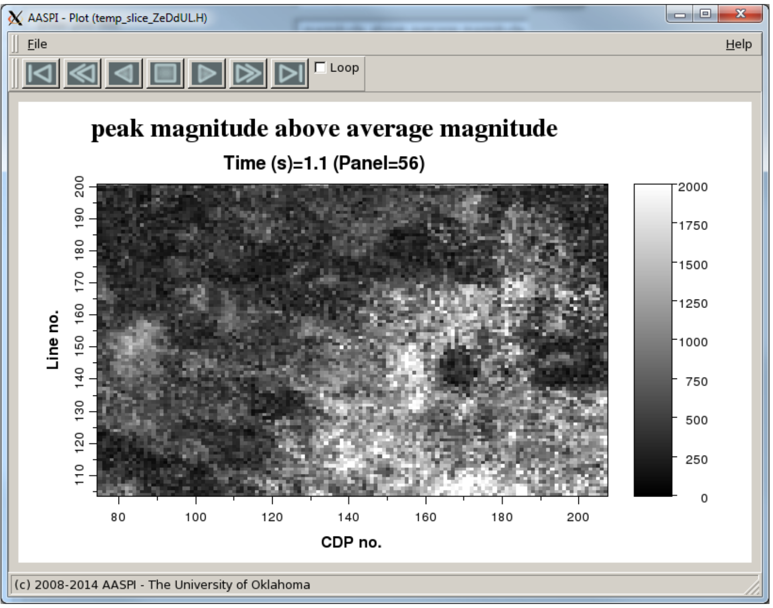


Figure 18: Time slice at t=1.1 through the peak magnitude above average volume. \ Since there are no channels at this level, the Boonsville data does not have a significant amount of lateral tuning, so the modulating the peak frequency by the peak magnitude above average gives almost the same image (though now with a different magnitude range of 0-3000 rather than 0-6000):

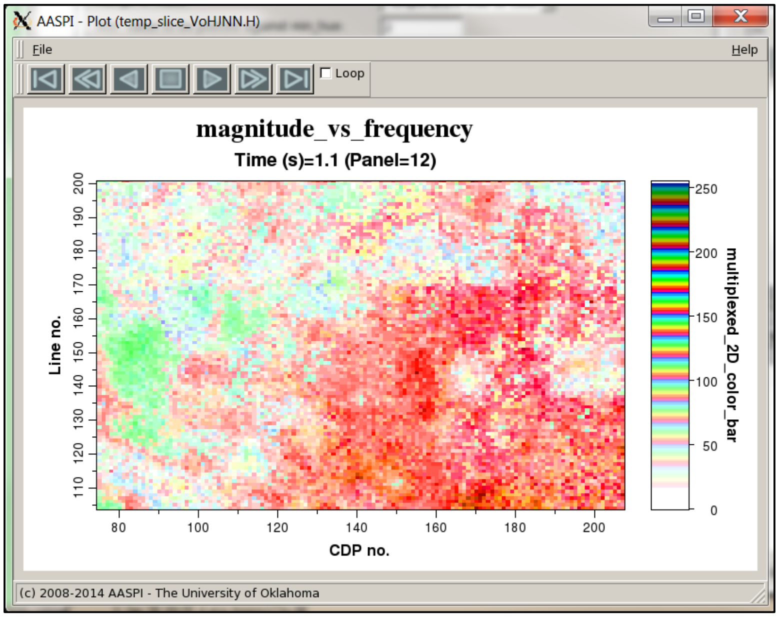
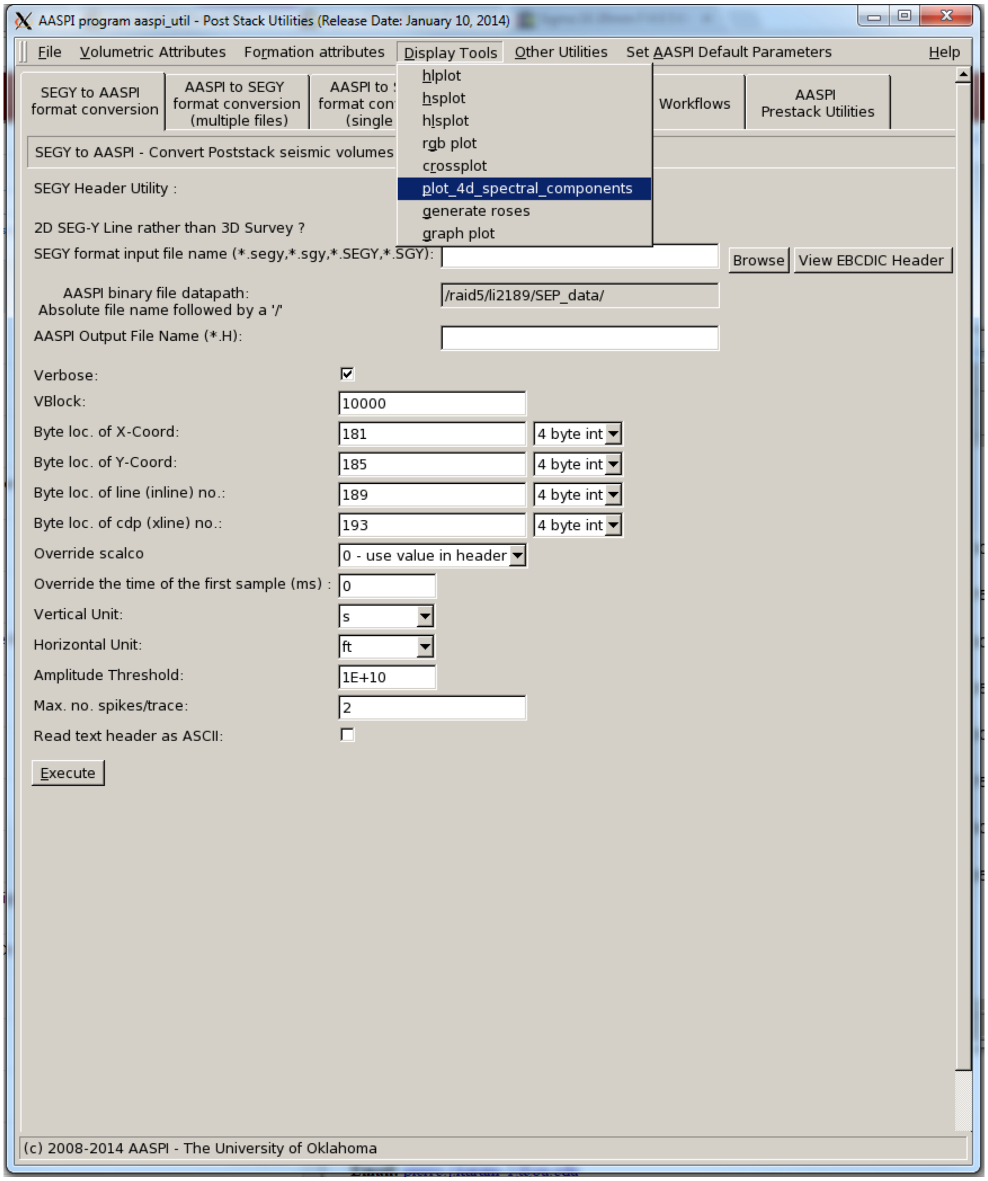


Figure 19: Time slice at t=1.1 s through a composite volume obtained by plotting peak frequency vs. hue and peak magnitude above average vs. lightness using program **hlplot.** There is little lateral variation in tuning in the volume such that this image looks very similar to that in Figure 15.

## Plotting Spectral Components

We provide a simple graphical interface to quality control the spectral components. Many commercial workstation software products now provide excellent interactive visualization of 4D volumes (*t, x, y,* and typically offset *h*, but in our case frequency, *f*). Our crude tool **plot_4D_spectral_components** can be found under the *Display tools* tab:

Spectral Attributes – Program **spec_cmp**


AASPI program aaspi_util - Post Stack Utilities (Release Date: January 10, 2014)
File
Volumetric Attributes
Formation attributes
Display Tools
Other Utilities
Set AASPI Default Parameters
Help
hlplot
hsplot
hlsplot
rgb plot
crossplot
plot_4d_spectral_components
generate roses
graph plot
SEGY to AASPI format conversion
AASPI to SEGY format conversion (multiple files)
AASPI to : format con (single
Workflows
AASPI Prestack Utilities
Browse
View EBCDIC Header
Execute

SEGY to AASPI - Convert Poststack seismic volumes

SEGY Header Utility :

2D SEG-Y Line rather than 3D Survey ?

SEGY format input file name (*.segy,*.sgy,*.SEGY,*.SGY):

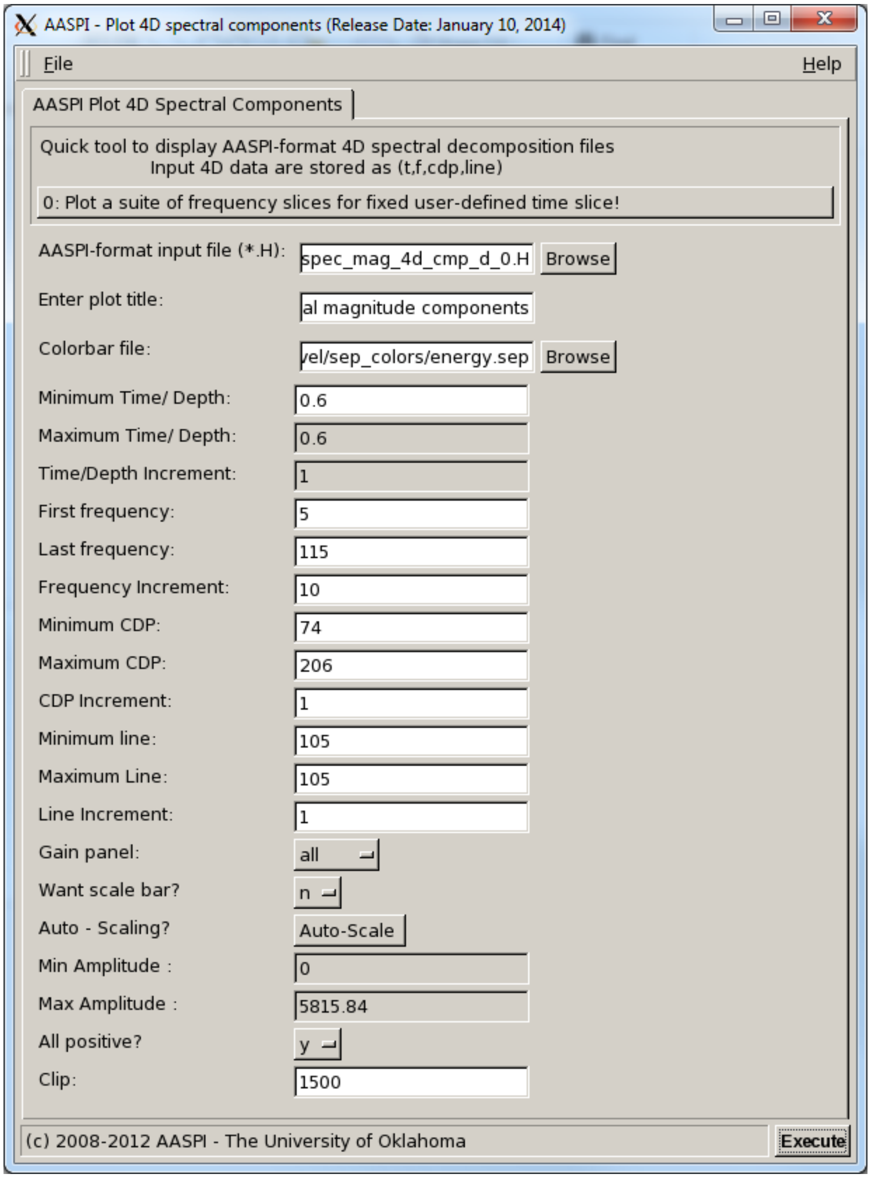
AASPI binary file datapath: Absolute file name followed by a '/' /raid5/li2189/SEP_data/

AASPI Output File Name (*.H):

| | | |
|---|---|---|
| Verbose: | ☑ | |
| VBlock: | 10000 | |
| Byte loc. of X-Coord: | 181 | 4 byte int |
| Byte loc. of Y-Coord: | 185 | 4 byte int |
| Byte loc. of line (inline) no.: | 189 | 4 byte int |
| Byte loc. of cdp (xline) no.: | 193 | 4 byte int |
| Override scalco | 0 - use value in header | |
| Override the time of the first sample (ms) : | 0 | |
| Vertical Unit: | s | |
| Horizontal Unit: | ft | |
| Amplitude Threshold: | 1E+10 | |
| Max. no. spikes/trace: | 2 | |
| Read text header as ASCII: | □ | |

(c) 2008-2014 AASPI - The University of Oklahoma

Previously, I had computed spectral components for the Boonsville survey and stored them as a 4D volume (*t,f,line_no,cdp_no)* in a file *spec_mag_4d_cmp_d.H*



AASPI - Plot 4D spectral components (Release Date: January 10, 2014)
File
Help
AASPI Plot 4D Spectral Components

Quick tool to display AASPI-format 4D spectral decomposition files
Input 4D data are stored as (t,f,cdp,line)

0: Plot a suite of frequency slices for fixed user-defined time slice!

AASPI-format input file (*.H): spec_mag_4d_cmp_d_0.H

Enter plot title: al magnitude components

Colorbar file: vel/sep_colors/energy.sep

Minimum Time/ Depth: 0.6

Maximum Time/ Depth: 0.6

Time/Depth Increment: 1

First frequency: 5

Last frequency: 115

Frequency Increment: 10

Minimum CDP: 74

Maximum CDP: 206

CDP Increment: 1

Minimum line: 105

Maximum Line: 105

Line Increment: 1

Gain panel: all

Want scale bar? n

Auto - Scaling? Auto-Scale

Min Amplitude : 0

Max Amplitude : 5815.84

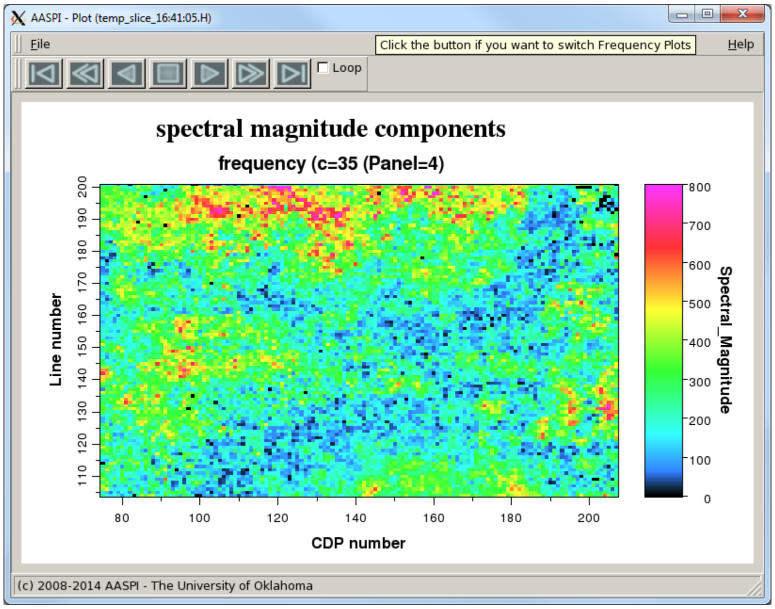
All positive? y

Clip: 1500

(c) 2008-2012 AASPI - The University of Oklahoma

The upper selection bar allows me to plot a constant frequency section or a constant time slice. I’ve chosen the 0.6s time slice. I choose the *energy.sep* color bar and obtain these slices with different frequencies at 0.6 s:

First, 35 Hz component:

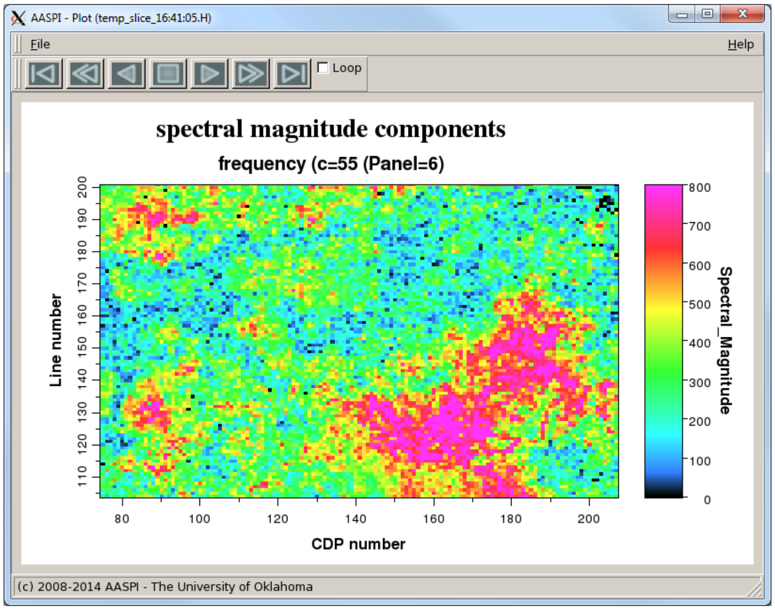


Click the button if you want to switch Frequency Plots

(c) 2008-2014 AASPI - The University of Oklahoma

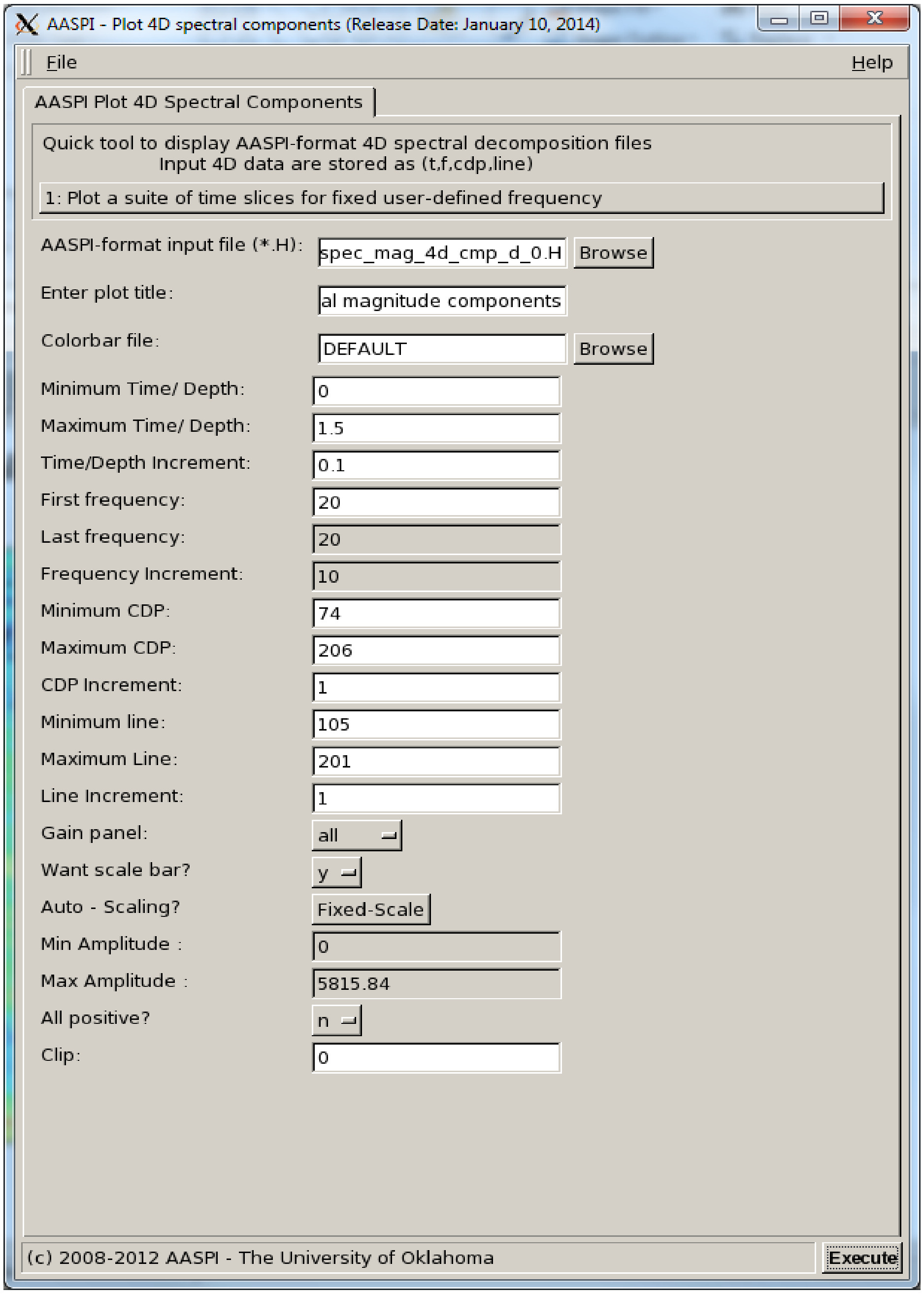
Figure 21: Time slice at t=0.6 s through the 35 Hz spectral magnitude volume.

Then, 55 Hz component:



(c) 2008-2014 AASPI - The University of Oklahoma

Figure 22: Time slice at t=0.6 s through the 55 Hz spectral magnitude volume.



AASPI - Plot 4D spectral components (Release Date: January 10, 2014)

Quick tool to display AASPI-format 4D spectral decomposition files
Input 4D data are stored as (t,f,cdp,line)

1: Plot a suite of time slices for fixed user-defined frequency

| | |
|---|---|
| AASPI-format input file (*.H): | spec_mag_4d_cmp_d_0.H |
| Enter plot title: | al magnitude components |
| Colorbar file: | DEFAULT |
| Minimum Time/ Depth: | 0 |
| Maximum Time/ Depth: | 1.5 |
| Time/Depth Increment: | 0.1 |
| First frequency: | 20 |
| Last frequency: | 20 |
| Frequency Increment: | 10 |
| Minimum CDP: | 74 |
| Maximum CDP: | 206 |
| CDP Increment: | 1 |
| Minimum line: | 105 |
| Maximum Line: | 201 |
| Line Increment: | 1 |
| Gain panel: | all |
| Want scale bar? | y |
| Auto - Scaling? | Fixed-Scale |
| Min Amplitude : | 0 |
| Max Amplitude : | 5815.84 |
| All positive? | n |
| Clip: | 0 |

(c) 2008-2012 AASPI - The University of Oklahoma

The upper selection bar allows me to plot a constant frequency section or a constant time slice. I’ve chosen the 20 Hz component. I choose the *energy.sep* color bar and obtain these slices at 1.0 s and 1.1 s through a 20 Hz volume:

First, 1s time slice of 20 Hz component:

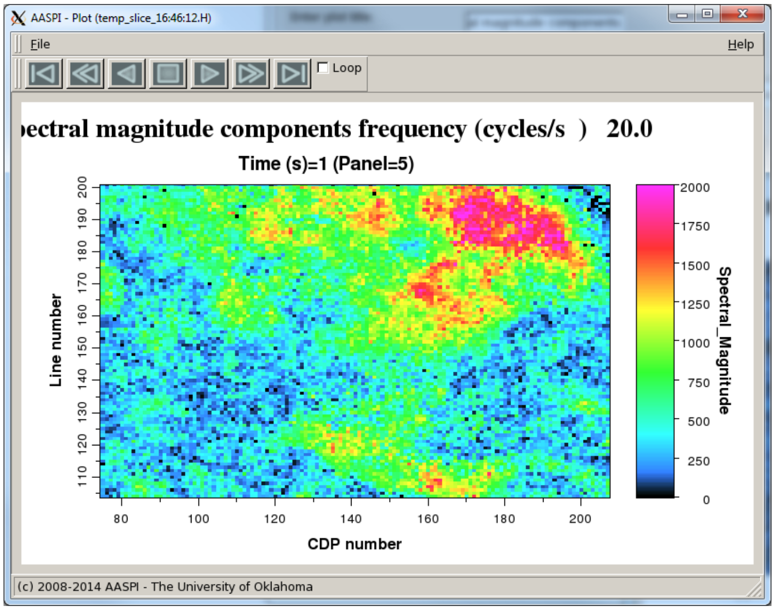


Figure 23: Time slice at t=1.0 s through the 20 Hz spectral magnitude volume.

First, 1.1s time slice of 20 Hz component:

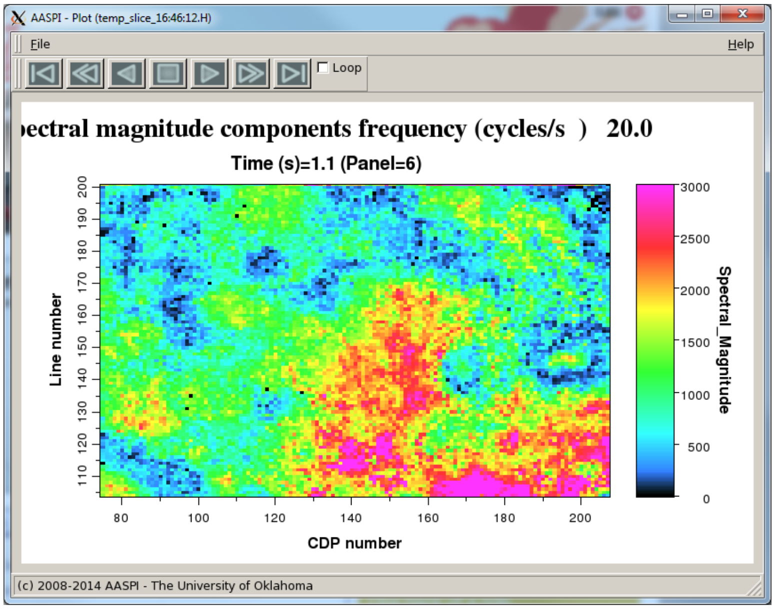
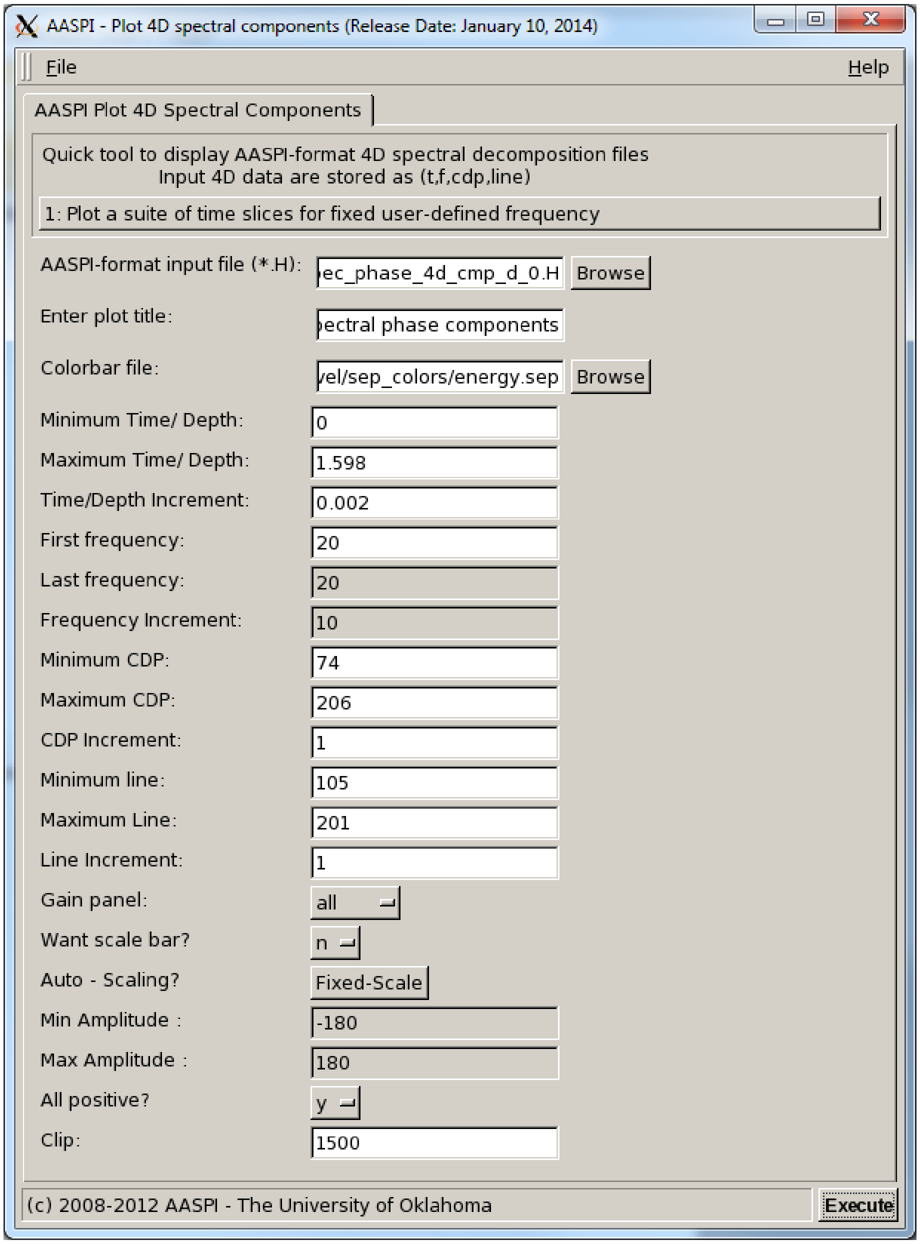


Figure 24: Time slice at t=1.1 s through the 20 Hz spectral magnitude volume.

I can generate similar plots of the *spec_phase_4d_cmp_0.H* 4D volume:



AASPI - Plot 4D spectral components (Release Date: January 10, 2014)

Quick tool to display AASPI-format 4D spectral decomposition files
Input 4D data are stored as (t,f,cdp,line)

1: Plot a suite of time slices for fixed user-defined frequency

| | |
|---|---|
| AASPI-format input file (*.H): | ec_phase_4d_cmp_d_0.H |
| Enter plot title: | ectral phase components |
| Colorbar file: | el/sep_colors/energy.sep |
| Minimum Time/ Depth: | 0 |
| Maximum Time/ Depth: | 1.598 |
| Time/Depth Increment: | 0.002 |
| First frequency: | 20 |
| Last frequency: | 20 |
| Frequency Increment: | 10 |
| Minimum CDP: | 74 |
| Maximum CDP: | 206 |
| CDP Increment: | 1 |
| Minimum line: | 105 |
| Maximum Line: | 201 |
| Line Increment: | 1 |
| Gain panel: | all |
| Want scale bar? | n |
| Auto - Scaling? | Fixed-Scale |
| Min Amplitude : | -180 |
| Max Amplitude : | 180 |
| All positive? | y |
| Clip: | 1500 |

(c) 2008-2012 AASPI - The University of Oklahoma

These parameters provide the phase component at 20 Hz:

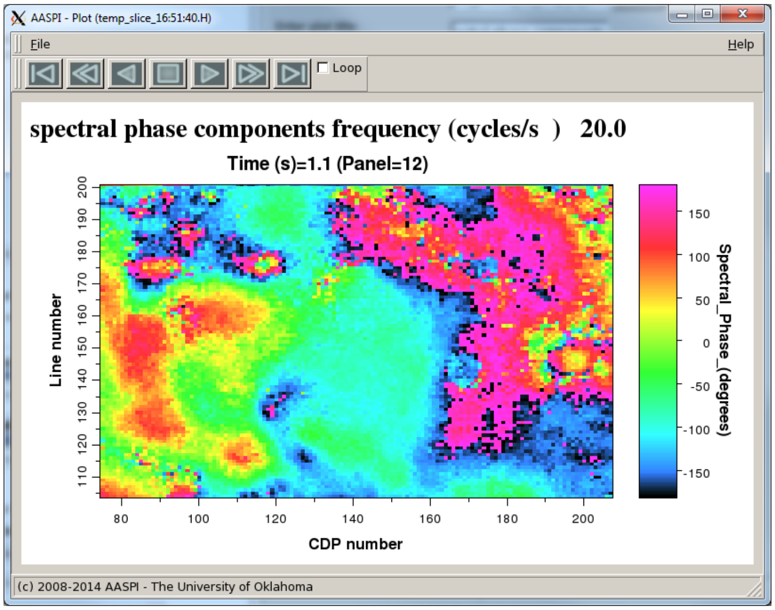


Figure 25. Time slice at t=1.1 s through the 20 Hz spectral phase volume.

## References

Liu, J. L., and K. J. Marfurt, 2007, Instantaneous spectral attributes to detect channels: Geophysics, **72**, P23-P31.

Liu, J. L., and K. J. Marfurt, 2007, Multi-color display of spectral attributes: The Leading Edge, **26**, 268-271.

Neep, J. P., 2007, Time variant colored inversion and spectral bluing: 69th EAGE Annual Meeting, Extended Abstracts, B009.

Zhang, K., K. J. Marfurt, and X. Guo, 2008, Volumetric application of skewed spectra: 77th Annual International Meeting of the SEG, Expanded Abstracts, 919-923.

Zhang, K., 2010, Seismic attribute analysis of unconventional reservoirs and stratigraphic patterns: Ph.D. dissertation, The University of Oklahoma.