CONSTRAINED LEAST-SQUARES SPECTRAL ANALYSIS – PROGRAM spec_clssa

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_clssa computation flow chart

While there is only one input file to program **spec_clssa**, 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, and phase residue 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.

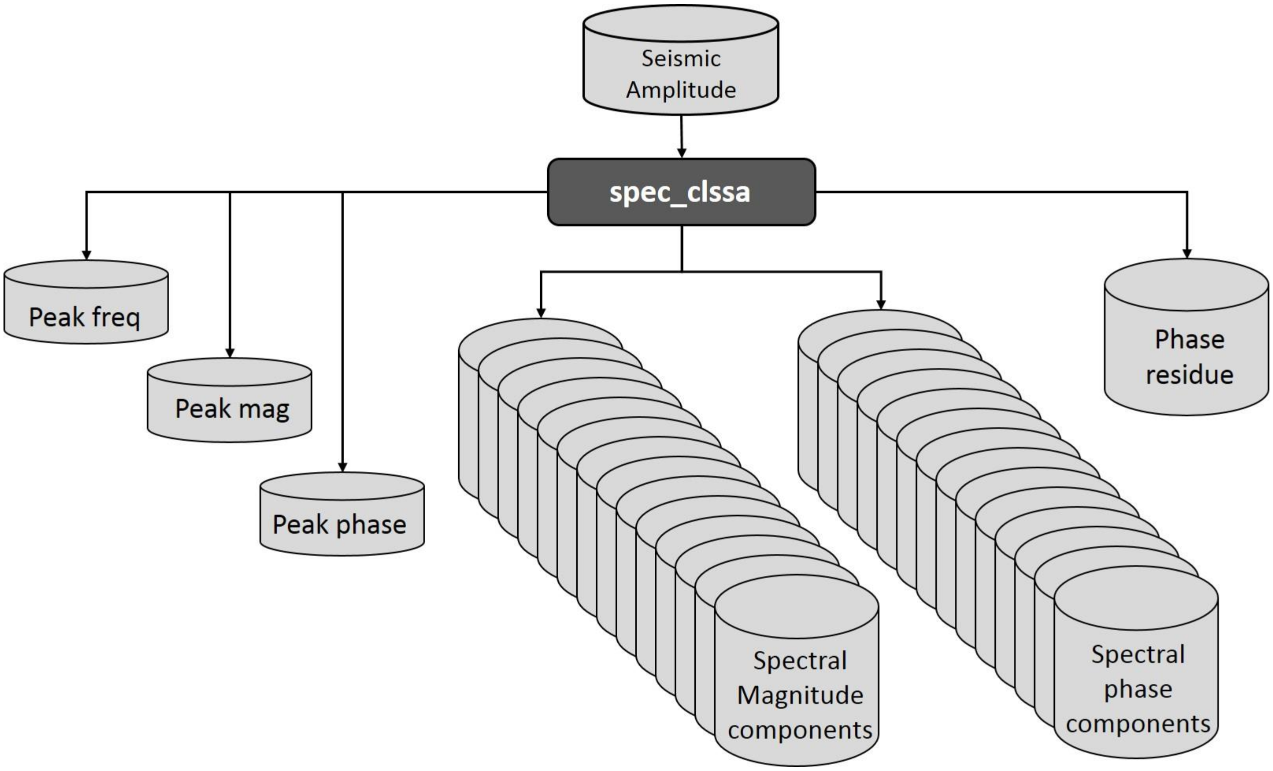
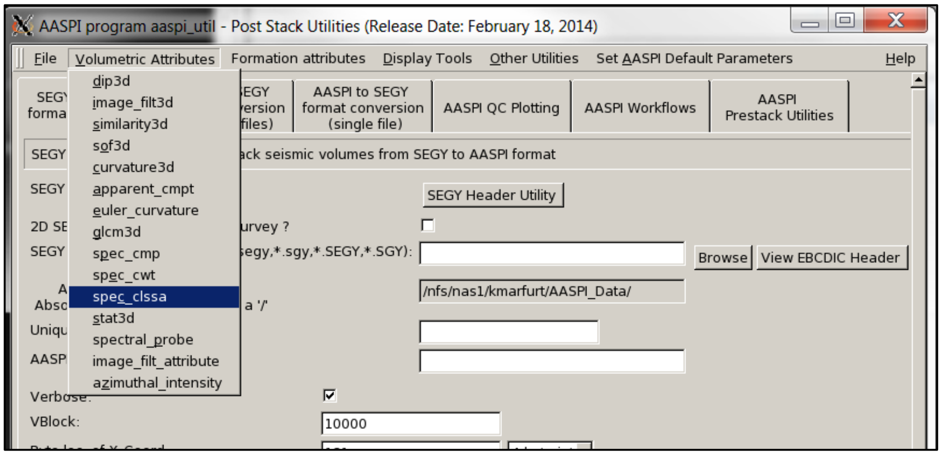


Figure 1. The flow chart for program **spec_clssa** – constrained least-squares spectral analysis.

Computing spectral components

To begin, click the *Volumetric Attributes* tab in the **aaspi_util** window and select the program **spec_clssa** :

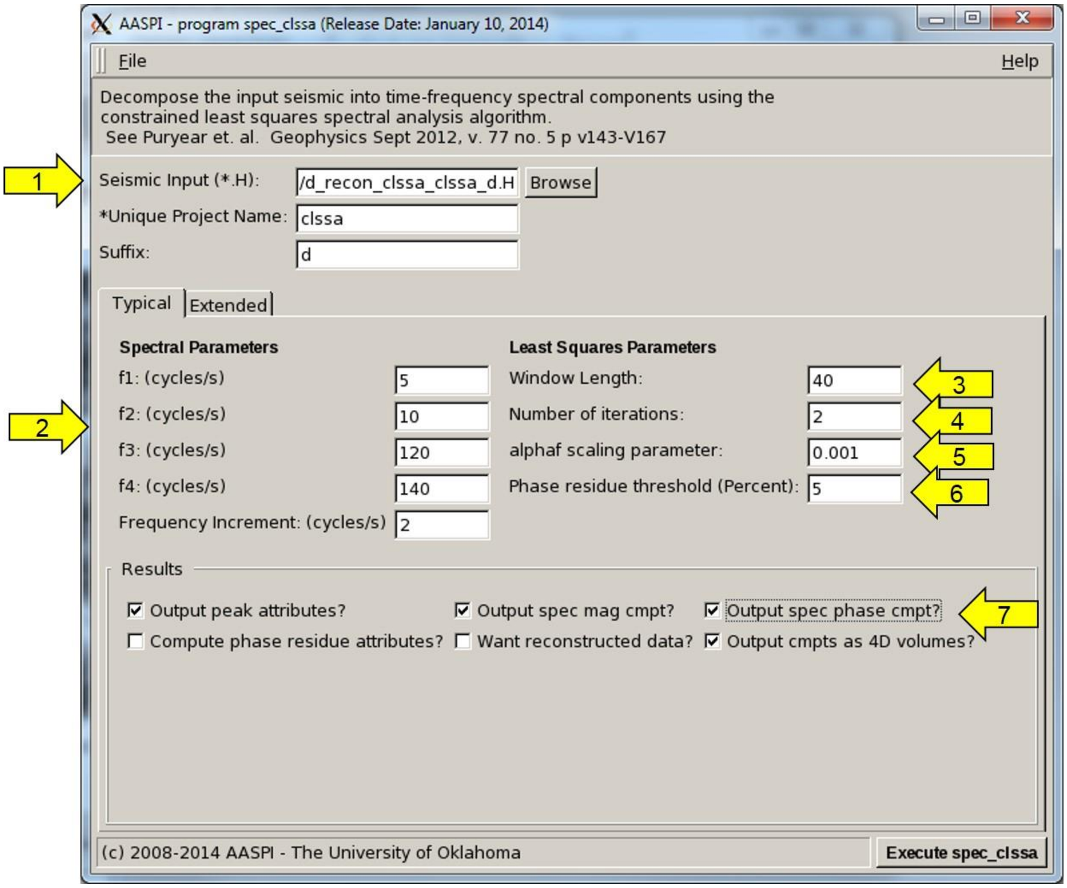


/nfs/nas1/kmarfurt/AASPI_Data/

Verbose: ☑

VBlock: 10000

Program **spec_clssa** performs spectral decomposition using a constrained least-squares spectral analysis method. The following window appears:



Decompose the input seismic into time-frequency spectral components using the constrained least squares spectral analysis algorithm.
See Puryear et. al. Geophysics Sept 2012, v. 77 no. 5 p v143-V167

Seismic Input (*.H): /d_recon_clssa_clssa_d.H
*Unique Project Name: clssa
Suffix: d

Spectral Parameters

f1: (cycles/s) 5
f2: (cycles/s) 10
f3: (cycles/s) 120
f4: (cycles/s) 140
Frequency Increment: (cycles/s) 2

Least Squares Parameters

Window Length: 40
Number of iterations: 2
alphaf scaling parameter: 0.001
Phase residue threshold (Percent): 5

Results

☑ Output peak attributes?
☑ Output spec mag cmpt?
☑ Output spec phase cmpt?
☐ Compute phase residue attributes?
☐ Want reconstructed data?
☑ Output cmpts as 4D volumes?

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

You will note that many of the attributes appear to be similar to those of program **spec_cmp**, spectral decomposition using a complex matching pursuit algorithm. First, enter the (1) the 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.

Constrained least-squares spectral analysis (clssa)

Puryear et al. (2012) proposed an inversion-based algorithm for computing the time frequency analysis of reflection seismograms using constrained least-squares spectral analysis. The method resulted in spectra that have reduced window smearing for a given window length relative to the discrete Fourier transform irrespective of window shape, and a time-frequency analysis with a combination of time and frequency resolution that is superior to the short time Fourier transform and the continuous wavelet transform. As compared with the continuous wavelet transform, the method has improved temporal resolution, particularly at low frequencies.

Seismic spectral decomposition is a trace-by-trace operation. Because each 1D seismic trace is converted to a 2D time-frequency panel, the process expands the dimensions of the original data via a non-unique transformation, suggesting an inversion-based approach to the problem. Several investigators have used different empirical criteria to define inversion-based spectral analysis methods. Following the Portniaguine and Castagna (2004) approach to seismic wavelet decomposition and reflectivity inversion, we invert the normal equations by applying an iteratively reweighted least-squares regularization algorithm to the complex spectral decomposition inverse problem as described in the box “Mathematical Implementation”.

Mathematical Implementation

Start with the definition of the forward problem.

$$\mathbf{Fm} = \mathbf{d}\ , \tag{A1}$$

where **F** is the kernel matrix composed of complex sinusoidal basis functions,

$$F_{nk} = \exp[i(2\pi k\Delta f n\Delta t)] = \cos(2\pi k\Delta f n\Delta t) + i\sin(2\pi k\Delta f n\Delta t)\,, \tag{A2}$$

$m_k=m(2\pi k\Delta f)$ is the model parameter vector (i.e. the desired spectral coefficients), and $d_n=d(n\Delta t)$ is the windowed seismic amplitude data. Forming the normal equations by multipying both sides of equation A1 by $\mathbf{F^H}$, the complex conjugate or Hilbert transpose of **F**, gives

$$\mathbf{F^H F m} = \mathbf{F^H d}\,. \tag{A3}$$

The objective of the inverse problem is to compute **m** given **F** and **d**. The classical least-squares solution to equation A1 is then

$$\mathbf{m} = \left(\mathbf{F^H F}\right)^{-1}\mathbf{F^H d}\,. \tag{A4}$$

Conventional Fourier transforms used in filtering are constructed to be orthogonal, such that $\mathbf{F^H F} = \mathbf{I}$, where **I** is the identity matrix. In this case, equation A4 reduces to

$$\mathbf{m} = \mathbf{F^H d}\,, \tag{A5}$$

which is equivalent to the discrete Fourier transform of the windowed trace segment. If the analysis window is T=0.1 s, orthogonality requires $\Delta f=1/T=$10 Hz. If we wish to estimate intermediate spectral components at a 2 Hz interval, we need to use the more general equation A4. A finer frequency increment results an underdetermined system with more unknowns than equations. We constrain the inversion by introducing two diagonal matrices. The data weight matrix, $\mathbf{W}_\mathrm{d}$, acts as a taper, where smaller weights at the window edges provides for greater tolerance of the misfit. In contrast, the model weight matrix, $\mathbf{W}_\mathrm{m}$, change iteratively, in general favoring the strongest spectral components, and attenuating the corresponding weaker aliases. Applying $\mathbf{W}_\mathrm{m}$ and $\mathbf{W}_\mathrm{d}$ to equation A1,

$$\mathbf{W_d F W_m (W_m)^{-1} m = W_d d}\quad , \tag{A6}$$

we introduce the weighted quantities

$$\mathbf{F_w} = \mathbf{W_d F W_m} \text{ and } \mathbf{m_w} = \mathbf{W_m}^{-1}\mathbf{m}\ , \tag{A6}$$

and then recast equation A5 as a model and data-weighted ill-posed inverse problem:

$$\mathbf{F_w m_w} = \mathbf{W_d d}\,. \tag{A7}$$

To solve equation A7, we add regularization:

$$\min \varepsilon^2 = \left\|\mathbf{F_w m_w} - \mathbf{W_d d}\right\|^2 + \alpha\left\|\mathbf{m_w}\right\|^2 \tag{A8}$$

where α is a regularization parameter that can be varied to control the sparsity and stability of the solution. Note that if $\mathbf{W_m}$ is proportional to **m**, small values of **m** become large values of $\mathbf{m_w}$ which are then minimized, giving us a sparse solution. Using an analytical Lagrange solution to equation A8, we solve first for the weighted model parameters

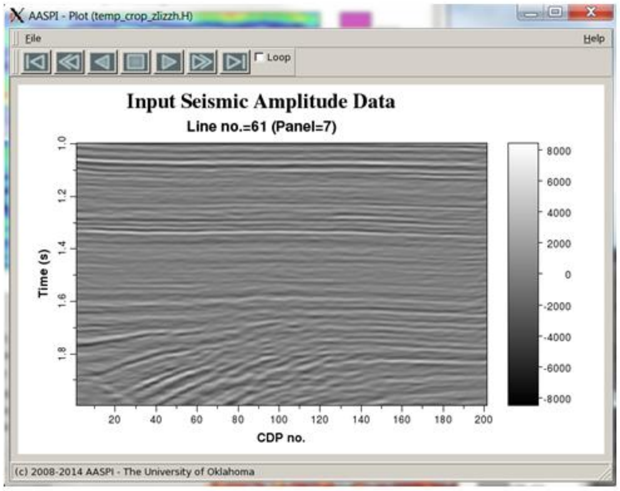
$$\mathbf{m_w} = \mathbf{F_w^H}\left(\mathbf{F_w F_w^H} + \alpha\mathbf{I}\right)^{-1}\mathbf{W_d d}\ , \tag{A9}$$

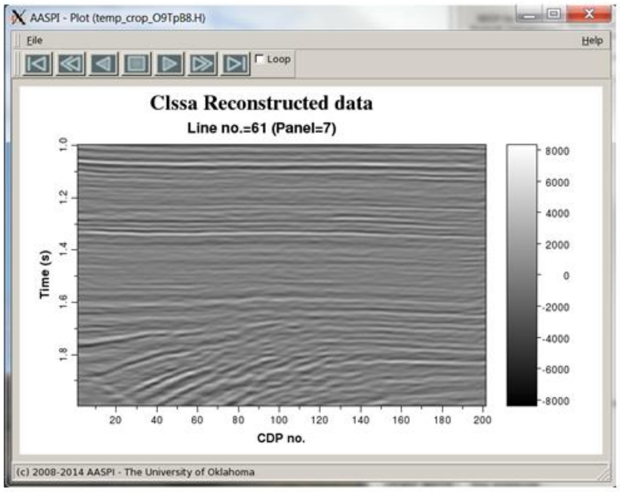
followed by reconstruction

$$\mathbf{m} = \mathbf{W_m m_w}\ . \tag{A10}$$

Values of α are chosen to be a percentage of $\left\|\mathbf{F_w F_w^H}\right\|$. Portniaguine and Castagna (2004) find the CLSSA utilizing a real basis function to be not quite as good as using a complex basis function, but better than the conventional DFT. This is significant because CLSSA using only the real waveform has precisely the same temporal resolution as the Fourier transform while having greatly improved frequency resolution. This indicates that, for a given window, CLSSA has a better Heisenberg uncertainty product (standard deviation of the waveform in time multiplied by the standard deviation of the spectrum) than does the Fourier transform. Thus far, they have not found a window for which this is not the case.

The *Reconstruction parameters* (2) 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 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 either spectral magnitude or spectral phase components as either a sequence of constant frequency volumes or as an equivalent 4D volume. In this example we used a 40 ms window, which implies a frequency resolution of Δf=1/.040 s=25 Hz is necessary to reconstruct the data using a DFT. Using this argument, we can say that the spectra is oversampled, which is validated by the excellent reconstruction on the right.





The (3) *Window length* indicates the length of the time window function used in spectral decomposition. As discussed above, the CLSSA algorithm is focused on solving an inverse problem. The window length parameter determines how long the inverse operator would cover, which can affect the time-frequency resolution. As an iterative method, the (4) *Number of iterations* defines the number of iterations used in iteratively reweighted least-squares solution. A larger number of iterations provides a more-accurate least-squares approximation, but also consumes more time. The (5) *alphaf scaling parameter* is the regularization parameter *α* in equations A8 and A9 above, which stabilizes the least-squares regression process.

The (6) *Phase residue threshold (Percent)* limits the phase residue attribute computation. When it is larger, more phase residue will appear corresponding to weaker events.

The Phase Residue

Ghiglia and Pritt (1998) provide an excellent survey of 2D phase-unwrapping techniques and show how a complex residue theorem based on vector calculus can be applied to the phase-unwrapping problem. They use a rectangular integration path aligned with the *t* and *f* axes. We choose a smaller diamond-shaped integration path about each sample *(j*Δ*t, k*Δ*f)* given by

$$I_{jk} = \frac{W\left[\psi(t_{j-1}, f_k) - \psi(t_j, f_{k-1})\right]}{2\pi} + \frac{W\left[\psi(t_j, f_{k-1}) - \psi(t_{j+1}, f_k)\right]}{2\pi} + \frac{W\left[\psi(t_{j+1}, f_k) - \psi(t_j, f_{k+1})\right]}{2\pi} + \frac{W\left[\psi(t_j, f_{k+1}) - \psi(t_{j-1}, f_k)\right]}{2\pi} \qquad \text{(B1)}$$

where ψ is the phase and *W* is a wrapping operator that produces an output that falls between ±π.If the integral I_{jk} in equation B1 is nonzero, there are inconsistent phase points, which Ghiglia and Pritt (1998) call “phase residues”. The figure below shows how the residue is calculated for a small portion of a typical wrapped time-frequency phase matrix. Area A is continuous, with a phase residue = 0.0, while area B is discontinuous with a phase residue = 1.0 .

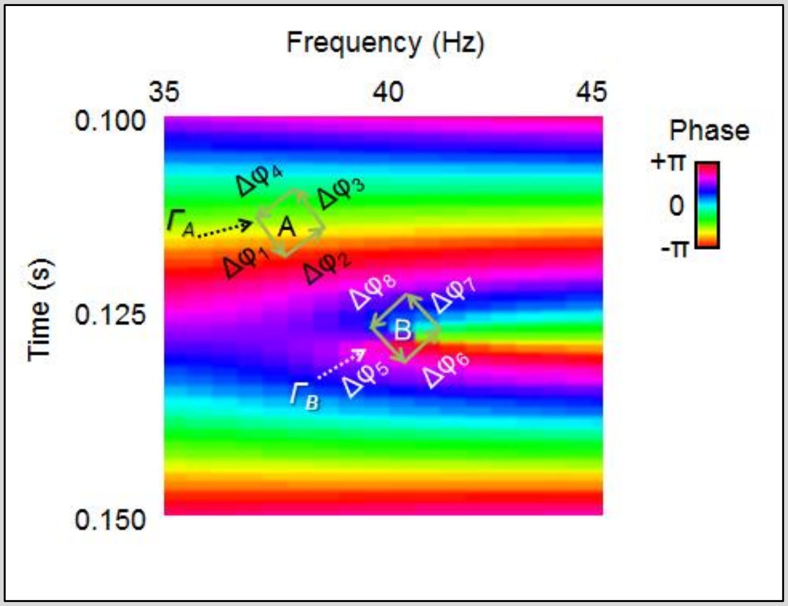


Figure B1. Line integrals about analysis points A and B used to compute the phase residue.

Matos et al. (2011) find phase attributes are sensitive to the same kinds of stratigraphic discontinuities seen by analyzing the magnitude component of time-frequency distribution using wavelet transforms and the continuous wavelet transform. Because phase is often a more seismic measure than magnitude, it holds significant promise in mapping stratigraphic unconformities.

Accurate computation of phase residues requires relatively fine sampling across frequency. We suggest setting the computational Δ*f*=1 Hz. Unfortunately, if you do this and choose to output these many components as well, you may fill up your disk drive. To analyze individual components we suggest a coarser output Δ*f*=5 Hz for subsequent loading into your interpretation workstation.

Statistical measures of the spectrum

Like program **spec_cmp**, program **spec_clssa** 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?*

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.

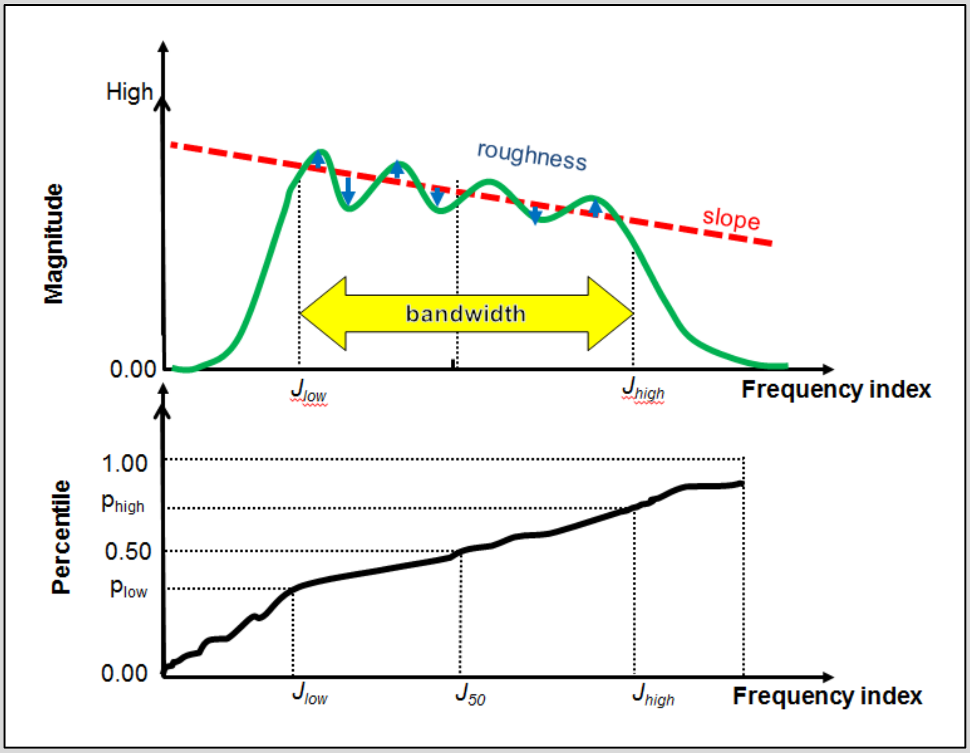
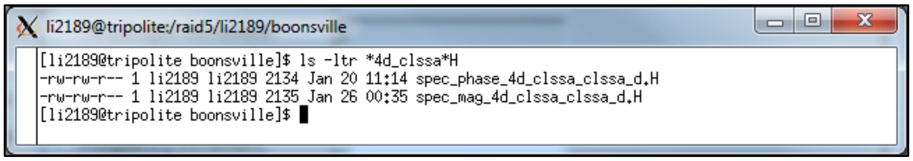


Figure B1. The shape of a spectrum (in green) containing 50 or more components approximated by average frequency, bandwidth, slope, and   roughness attributes.

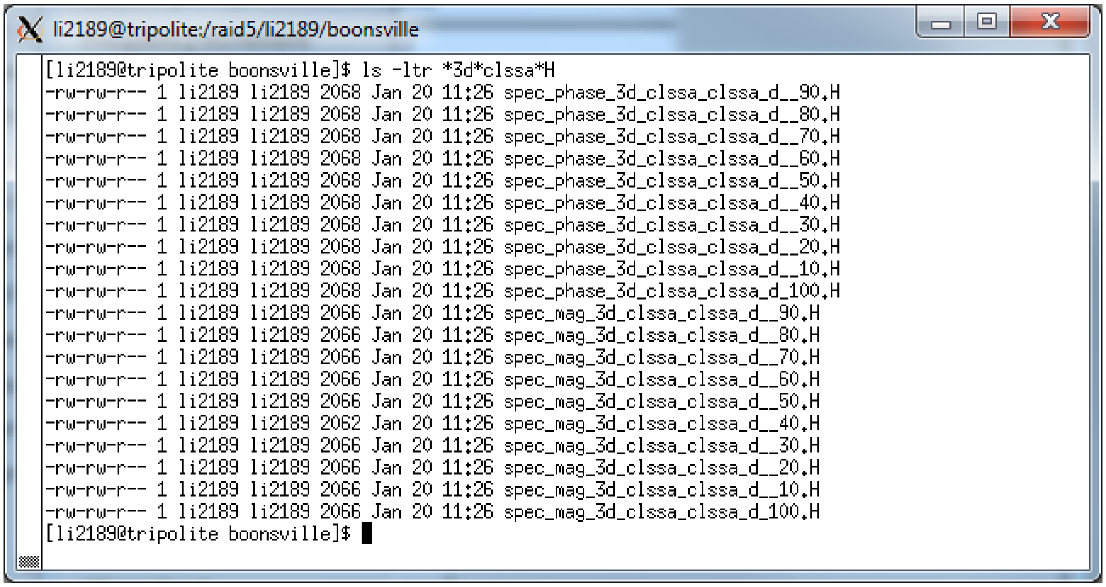
Individual spectral components

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_clssa*H
-rw-rw-r-- 1 li2189 li2189 2134 Jan 20 11:14 spec_phase_4d_clssa_clssa_d.H
-rw-rw-r-- 1 li2189 li2189 2135 Jan 26 00:35 spec_mag_4d_clssa_clssa_d.H
[li2189@tripolite boonsville]$ █
```

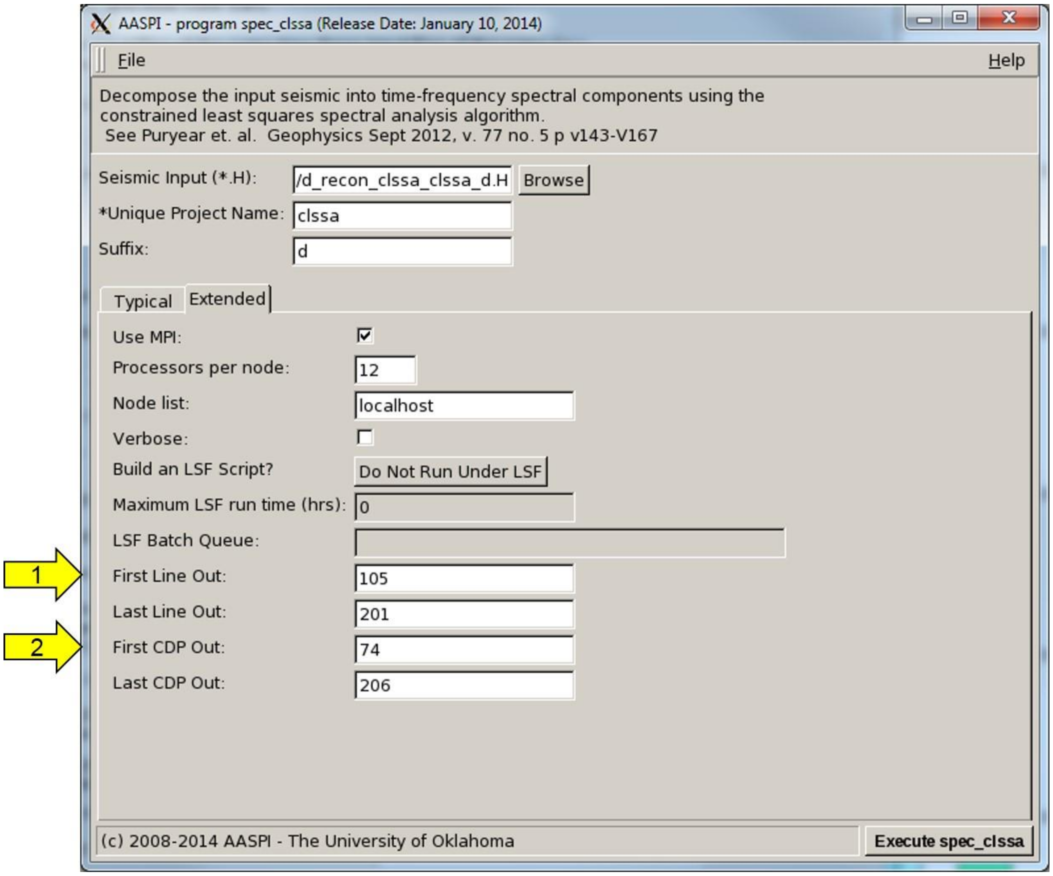
If you ask for spectral components not to be stored as a 4D cube the constant-frequency 3D spectral magnitude and spectral phase volumes will have the frequency value encoded in the file name:



```
[li2189@tripolite boonsville]$ ls -ltr *3d*clssa*H
-rw-rw-r-- 1 li2189 li2189 2068 Jan 20 11:26 spec_phase_3d_clssa_clssa_d__90.H
-rw-rw-r-- 1 li2189 li2189 2068 Jan 20 11:26 spec_phase_3d_clssa_clssa_d__80.H
-rw-rw-r-- 1 li2189 li2189 2068 Jan 20 11:26 spec_phase_3d_clssa_clssa_d__70.H
-rw-rw-r-- 1 li2189 li2189 2068 Jan 20 11:26 spec_phase_3d_clssa_clssa_d__60.H
-rw-rw-r-- 1 li2189 li2189 2068 Jan 20 11:26 spec_phase_3d_clssa_clssa_d__50.H
-rw-rw-r-- 1 li2189 li2189 2068 Jan 20 11:26 spec_phase_3d_clssa_clssa_d__40.H
-rw-rw-r-- 1 li2189 li2189 2068 Jan 20 11:26 spec_phase_3d_clssa_clssa_d__30.H
-rw-rw-r-- 1 li2189 li2189 2068 Jan 20 11:26 spec_phase_3d_clssa_clssa_d__20.H
-rw-rw-r-- 1 li2189 li2189 2068 Jan 20 11:26 spec_phase_3d_clssa_clssa_d__10.H
-rw-rw-r-- 1 li2189 li2189 2068 Jan 20 11:26 spec_phase_3d_clssa_clssa_d_100.H
-rw-rw-r-- 1 li2189 li2189 2066 Jan 20 11:26 spec_mag_3d_clssa_clssa_d__90.H
-rw-rw-r-- 1 li2189 li2189 2066 Jan 20 11:26 spec_mag_3d_clssa_clssa_d__80.H
-rw-rw-r-- 1 li2189 li2189 2066 Jan 20 11:26 spec_mag_3d_clssa_clssa_d__70.H
-rw-rw-r-- 1 li2189 li2189 2066 Jan 20 11:26 spec_mag_3d_clssa_clssa_d__60.H
-rw-rw-r-- 1 li2189 li2189 2066 Jan 20 11:26 spec_mag_3d_clssa_clssa_d__50.H
-rw-rw-r-- 1 li2189 li2189 2062 Jan 20 11:26 spec_mag_3d_clssa_clssa_d__40.H
-rw-rw-r-- 1 li2189 li2189 2066 Jan 20 11:26 spec_mag_3d_clssa_clssa_d__30.H
-rw-rw-r-- 1 li2189 li2189 2066 Jan 20 11:26 spec_mag_3d_clssa_clssa_d__20.H
-rw-rw-r-- 1 li2189 li2189 2066 Jan 20 11:26 spec_mag_3d_clssa_clssa_d__10.H
-rw-rw-r-- 1 li2189 li2189 2066 Jan 20 11:26 spec_mag_3d_clssa_clssa_d_100.H
[li2189@tripolite boonsville]$ █
```

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

Spectral Attributes: Program **spec_clssa**



Decompose the input seismic into time-frequency spectral components using the constrained least squares spectral analysis algorithm.
See Puryear et. al. Geophysics Sept 2012, v. 77 no. 5 p v143-V167

Seismic Input (*.H): /d_recon_clssa_clssa_d.H

*Unique Project Name: clssa

Suffix: d

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:

1 First Line Out: 105

Last Line Out: 201

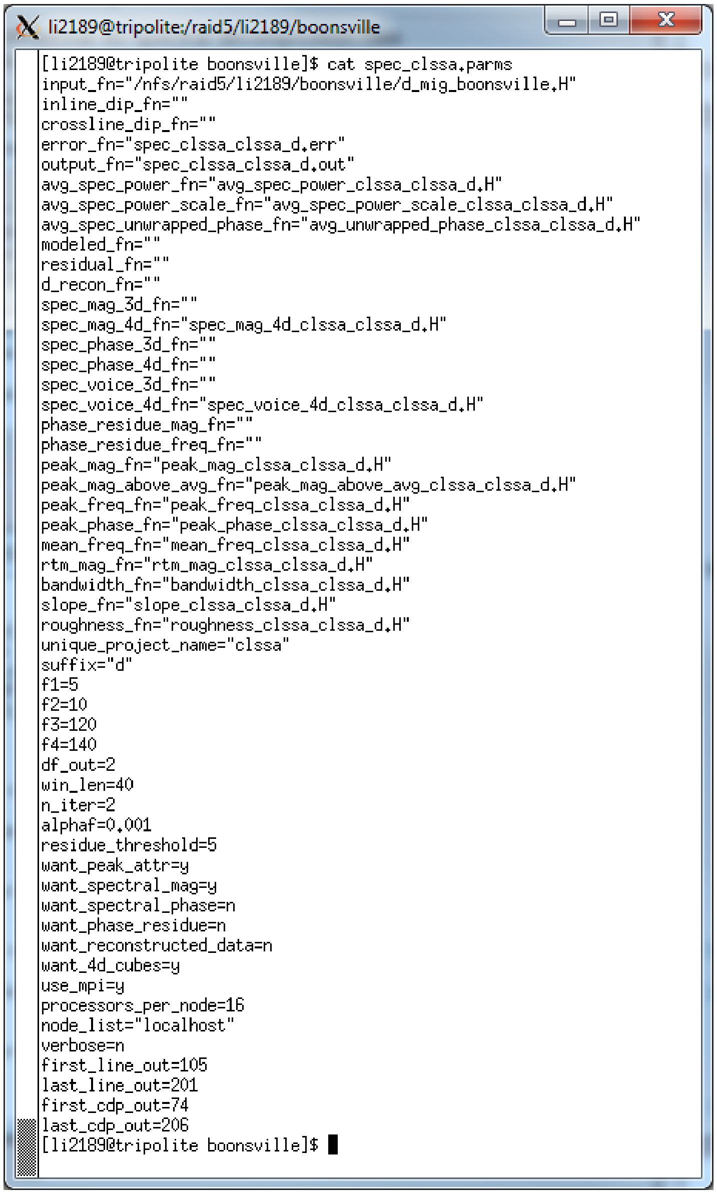
2 First CDP Out: 74

Last CDP Out: 206

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

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

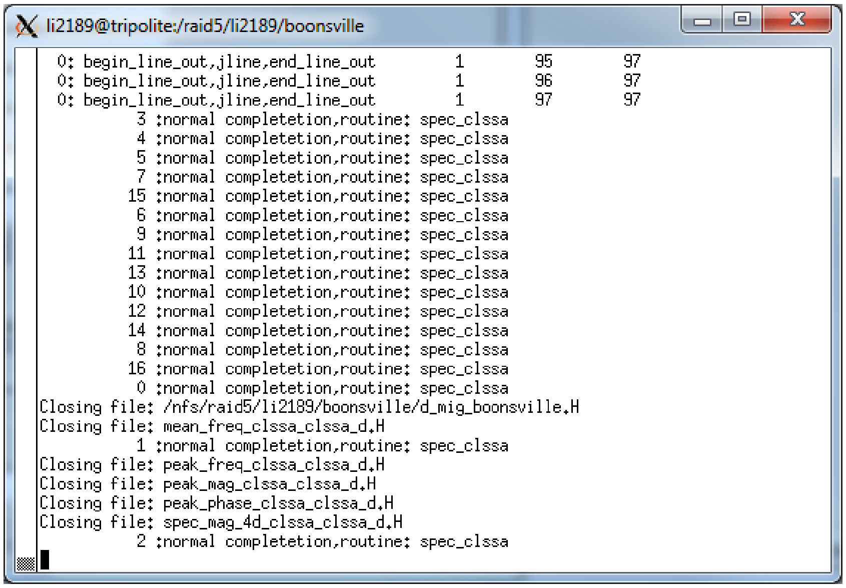
You will probably want to experiment with these parameters a bit to calibrate them for the kind of data your encounter. It is reasonable to expect that most surveys of a similar vintage from the Gulf of Mexico will have similar spectral ranges and signal-to-noise ratios. Likewise, similar surveys acquired in the Permian Basin of west Texas will be similar to each other. In order to simplify parameter choices, In order to simplify parameter choices, you can then *cat* AASPI ".parms" file to examine its content or set it to your desired default parameters:



```
[li2189@tripolite boonsville]$ cat spec_clssa.parms
input_fn="/nfs/raid5/li2189/boonsville/d_mig_boonsville.H"
inline_dip_fn=""
crossline_dip_fn=""
error_fn="spec_clssa_clssa_d.err"
output_fn="spec_clssa_clssa_d.out"
avg_spec_power_fn="avg_spec_power_clssa_clssa_d.H"
avg_spec_power_scale_fn="avg_spec_power_scale_clssa_clssa_d.H"
avg_spec_unwrapped_phase_fn="avg_unwrapped_phase_clssa_clssa_d.H"
modeled_fn=""
residual_fn=""
d_recon_fn=""
spec_mag_3d_fn=""
spec_mag_4d_fn="spec_mag_4d_clssa_clssa_d.H"
spec_phase_3d_fn=""
spec_phase_4d_fn=""
spec_voice_3d_fn=""
spec_voice_4d_fn="spec_voice_4d_clssa_clssa_d.H"
phase_residue_mag_fn=""
phase_residue_freq_fn=""
peak_mag_fn="peak_mag_clssa_clssa_d.H"
peak_mag_above_avg_fn="peak_mag_above_avg_clssa_clssa_d.H"
peak_freq_fn="peak_freq_clssa_clssa_d.H"
peak_phase_fn="peak_phase_clssa_clssa_d.H"
mean_freq_fn="mean_freq_clssa_clssa_d.H"
rtm_mag_fn="rtm_mag_clssa_clssa_d.H"
bandwidth_fn="bandwidth_clssa_clssa_d.H"
slope_fn="slope_clssa_clssa_d.H"
roughness_fn="roughness_clssa_clssa_d.H"
unique_project_name="clssa"
suffix="d"
f1=5
f2=10
f3=120
f4=140
df_out=2
win_len=40
n_iter=2
alphaf=0.001
residue_threshold=5
want_peak_attr=y
want_spectral_mag=y
want_spectral_phase=n
want_phase_residue=n
want_reconstructed_data=n
want_4d_cubes=y
use_mpi=y
processors_per_node=16
node_list="localhost"
verbose=n
first_line_out=105
last_line_out=201
first_cdp_out=74
last_cdp_out=206
[li2189@tripolite boonsville]$ █
```

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

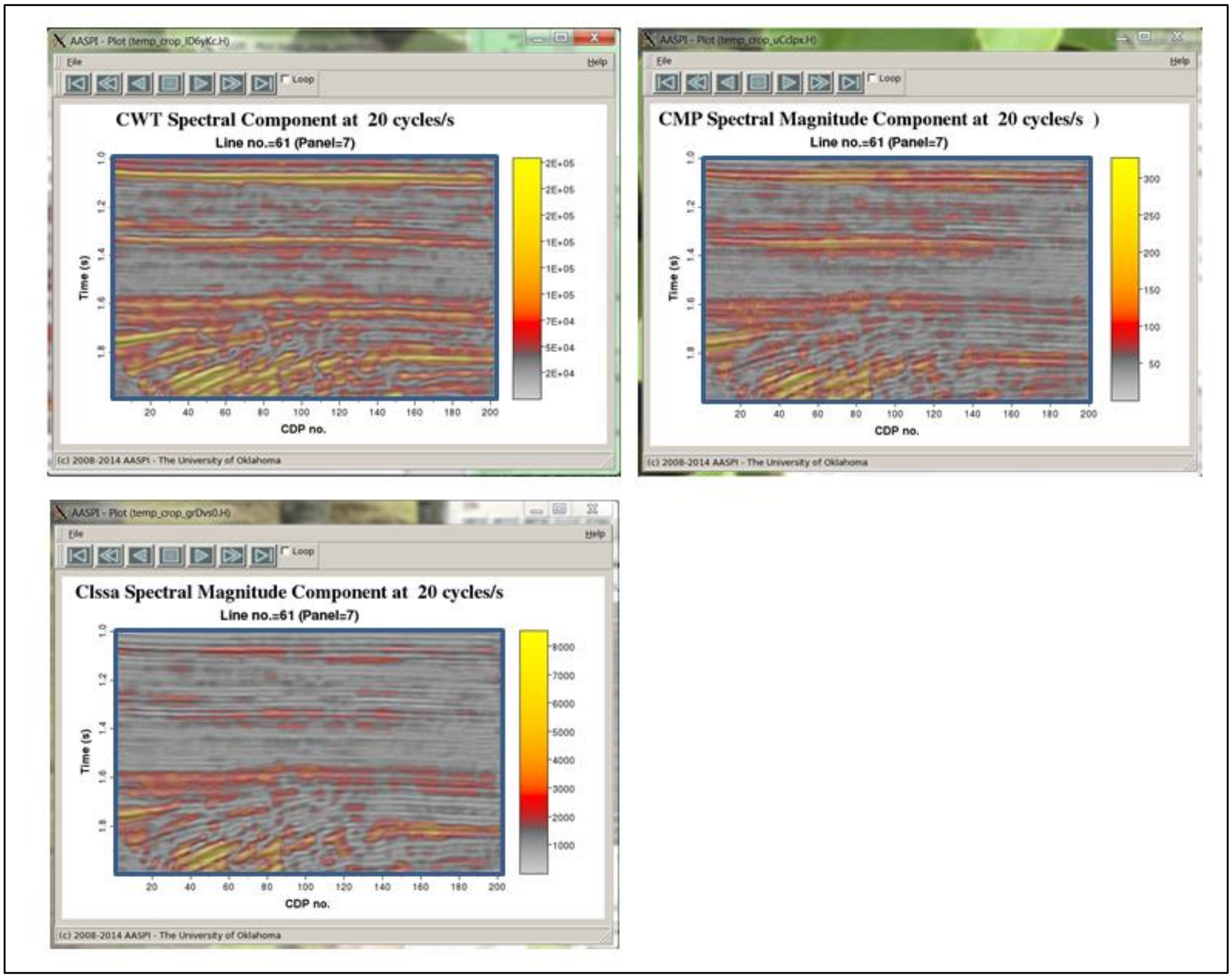
As in all the AASPI GUIs, click *Execute* to run the program. The end of your run should looks something like the following:

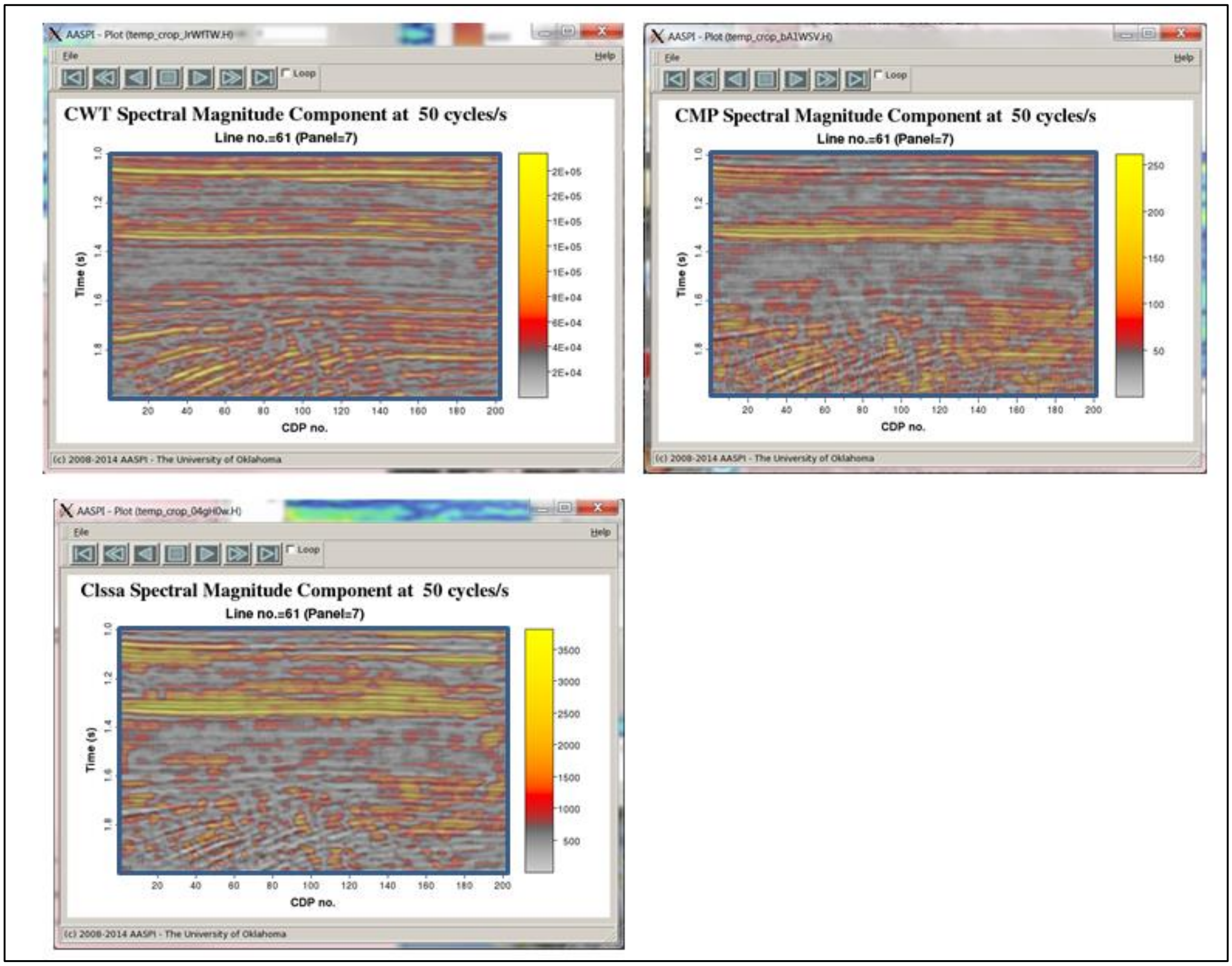


```
  0: begin_line_out,jline,end_line_out        1       95        97
  0: begin_line_out,jline,end_line_out        1       96        97
  0: begin_line_out,jline,end_line_out        1       97        97
           3 :normal completetion,routine: spec_clssa
           4 :normal completetion,routine: spec_clssa
           5 :normal completetion,routine: spec_clssa
           7 :normal completetion,routine: spec_clssa
          15 :normal completetion,routine: spec_clssa
           6 :normal completetion,routine: spec_clssa
           9 :normal completetion,routine: spec_clssa
          11 :normal completetion,routine: spec_clssa
          13 :normal completetion,routine: spec_clssa
          10 :normal completetion,routine: spec_clssa
          12 :normal completetion,routine: spec_clssa
          14 :normal completetion,routine: spec_clssa
           8 :normal completetion,routine: spec_clssa
          16 :normal completetion,routine: spec_clssa
           0 :normal completetion,routine: spec_clssa
Closing file: /nfs/raid5/li2189/boonsville/d_mig_boonsville.H
Closing file: mean_freq_clssa_clssa_d.H
           1 :normal completetion,routine: spec_clssa
Closing file: peak_freq_clssa_clssa_d.H
Closing file: peak_mag_clssa_clssa_d.H
Closing file: peak_phase_clssa_clssa_d.H
Closing file: spec_mag_4d_clssa_clssa_d.H
           2 :normal completetion,routine: spec_clssa
```

Example 1: Comparison of the three decomposition algorithms

Our first example compares vertical slices of spectral components at 20 Hz and 50 Hz through the UT BEG Stratton data volume. Note that 20 Hz has a period of 0.050 s which is larger than the 0.040 s analysis window used in **spec_clssa**. The 20 Hz component from **spec_cwt** is not really a component, but rather the relatively broad band response of a Morlet wavelet centered at 20 Hz, with a Gaussian spectrum with half power = f. The 20 Hz component from spec_cmp is the complex sum of all 20 Hz components of Morlet wavelets about a given time sample. For this reason, both **spec_clssa** and **spec_cmp** have higher frequency resolution than **spec_cwt**.

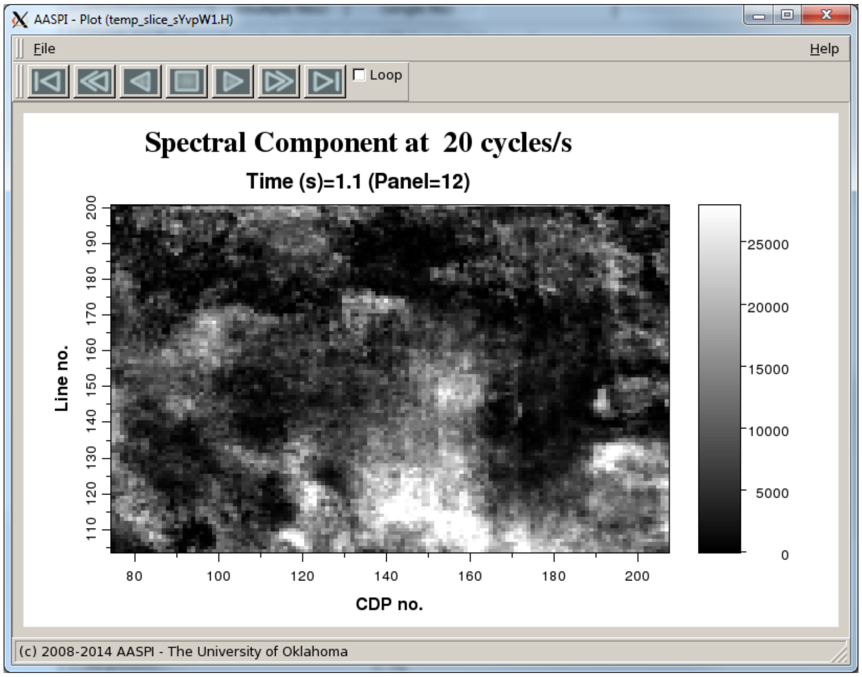

AASPI - Plot (temp_crop_lD6yKc.H)
File
Help
Loop
CWT Spectral Component at 20 cycles/s
Line no.=61 (Panel=7)
Time (s)
1.0
1.2
1.4
1.6
1.8
20 40 60 80 100 120 140 160 180 200
CDP no.
2E+05
2E+05
2E+05
1E+05
1E+05
1E+05
7E+04
5E+04
2E+04
(c) 2008-2014 AASPI - The University of Oklahoma
AASPI - Plot (temp_crop_uCdpx.H)
File
Help
Loop
CMP Spectral Magnitude Component at 20 cycles/s)
Line no.=61 (Panel=7)
Time (s)
1.0
1.2
1.4
1.6
1.8
20 40 60 80 100 120 140 160 180 200
CDP no.
300
250
200
150
100
50
(c) 2008-2014 AASPI - The University of Oklahoma
AASPI - Plot (temp_crop_grDvs0.H)
File
Help
Loop
Clssa Spectral Magnitude Component at 20 cycles/s
Line no.=61 (Panel=7)
Time (s)
1.0
1.2
1.4
1.6
1.8
20 40 60 80 100 120 140 160 180 200
CDP no.
8000
7000
6000
5000
4000
3000
2000
1000
(c) 2008-2014 AASPI - The University of Oklahoma


AASPI - Plot (temp_crop_JrWfTW.H)
File
Help
Loop
CWT Spectral Magnitude Component at 50 cycles/s
Line no.=61 (Panel=7)
Time (s)
1.0
1.2
1.4
1.6
1.8
20
40
60
80
100
120
140
160
180
200
CDP no.
2E+05
2E+05
1E+05
1E+05
1E+05
8E+04
6E+04
4E+04
2E+04
(c) 2008-2014 AASPI - The University of Oklahoma
AASPI - Plot (temp_crop_bA1WSV.H)
CMP Spectral Magnitude Component at 50 cycles/s
Line no.=61 (Panel=7)
250
200
150
100
50
AASPI - Plot (temp_crop_04gH0w.H)
Clssa Spectral Magnitude Component at 50 cycles/s
Line no.=61 (Panel=7)
3500
3000
2500
2000
1500
1000
500

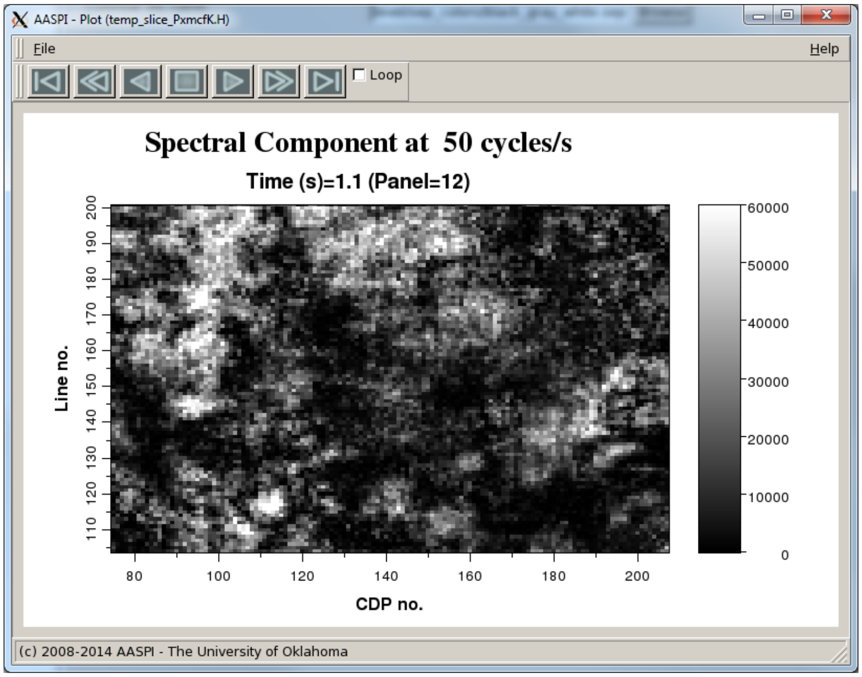
Example 2: clssa applied to the Boonsville survey

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.

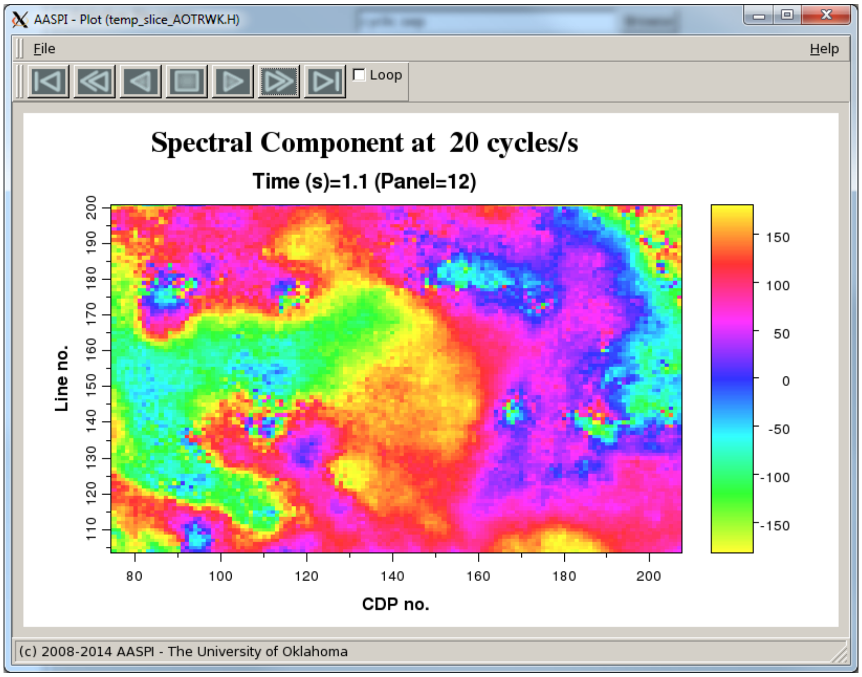
Plotting the same time slice as in all the other examples, and setting Allpos=y in our SEP Viewer GUI for the strictly positive magnitude, the *spec_mag_3d_clssa_d_20.H* (the 20 Hz magnitude component) file looks like this:



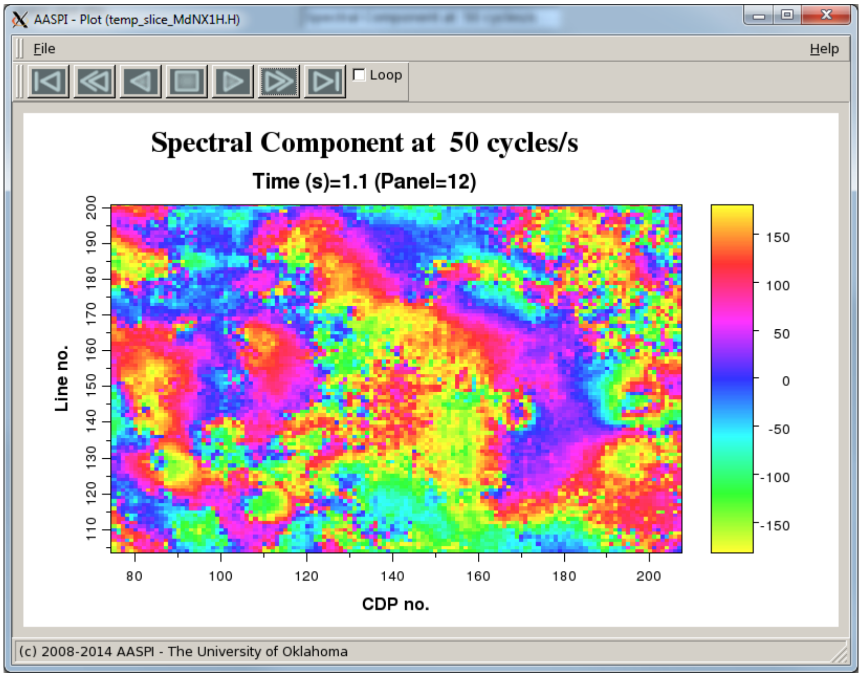
While the *spec_mag_3d_clssa_d_50.H* (the 50 Hz magnitude component) file looks like this:



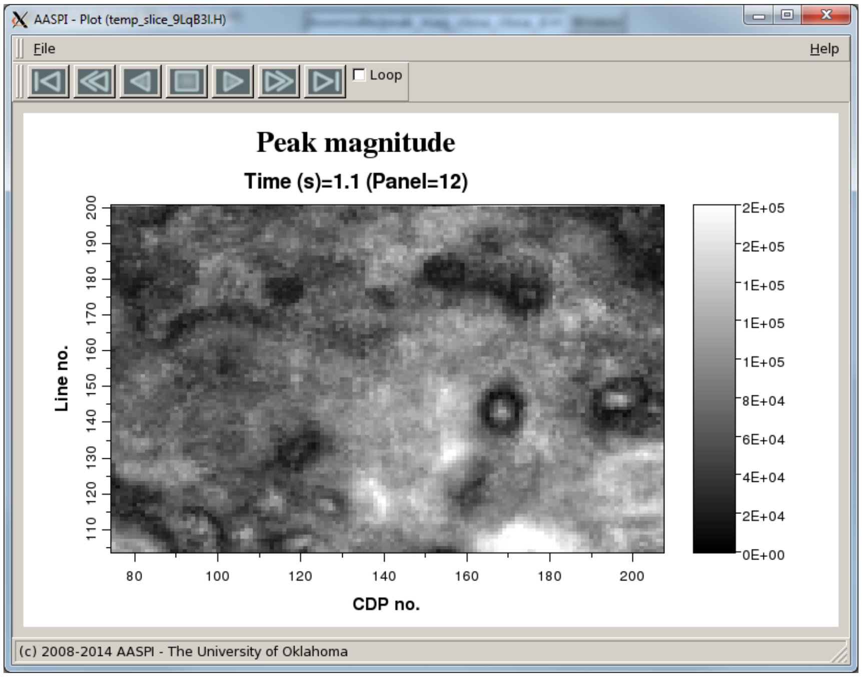
The phase components will range from -180^0 to +180^0, so set Allpos=n and choose a cyclical color bar to plot *spec_phase_3d_clssa_d_20.H* (the 20 Hz phase component):



and *spec_phase_3d_clssa_d_50.H* (the 50 Hz phase component):

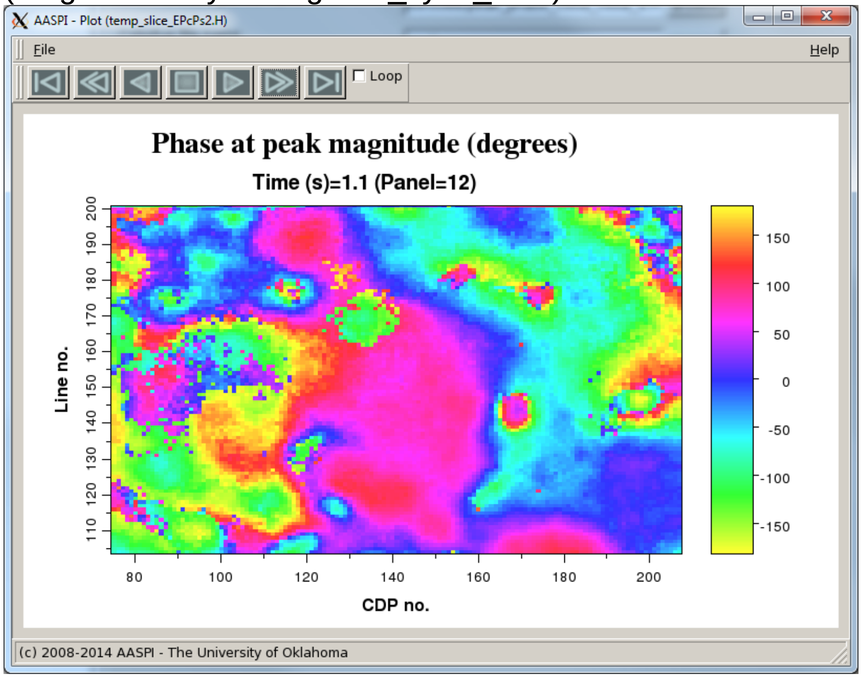


Several measures of the spectrum at each time sample are also computed. The simplest one is the peak magnitude (the greatest value of the spectrum) here as the file *peak_magnitude_clssa_d.H*:



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

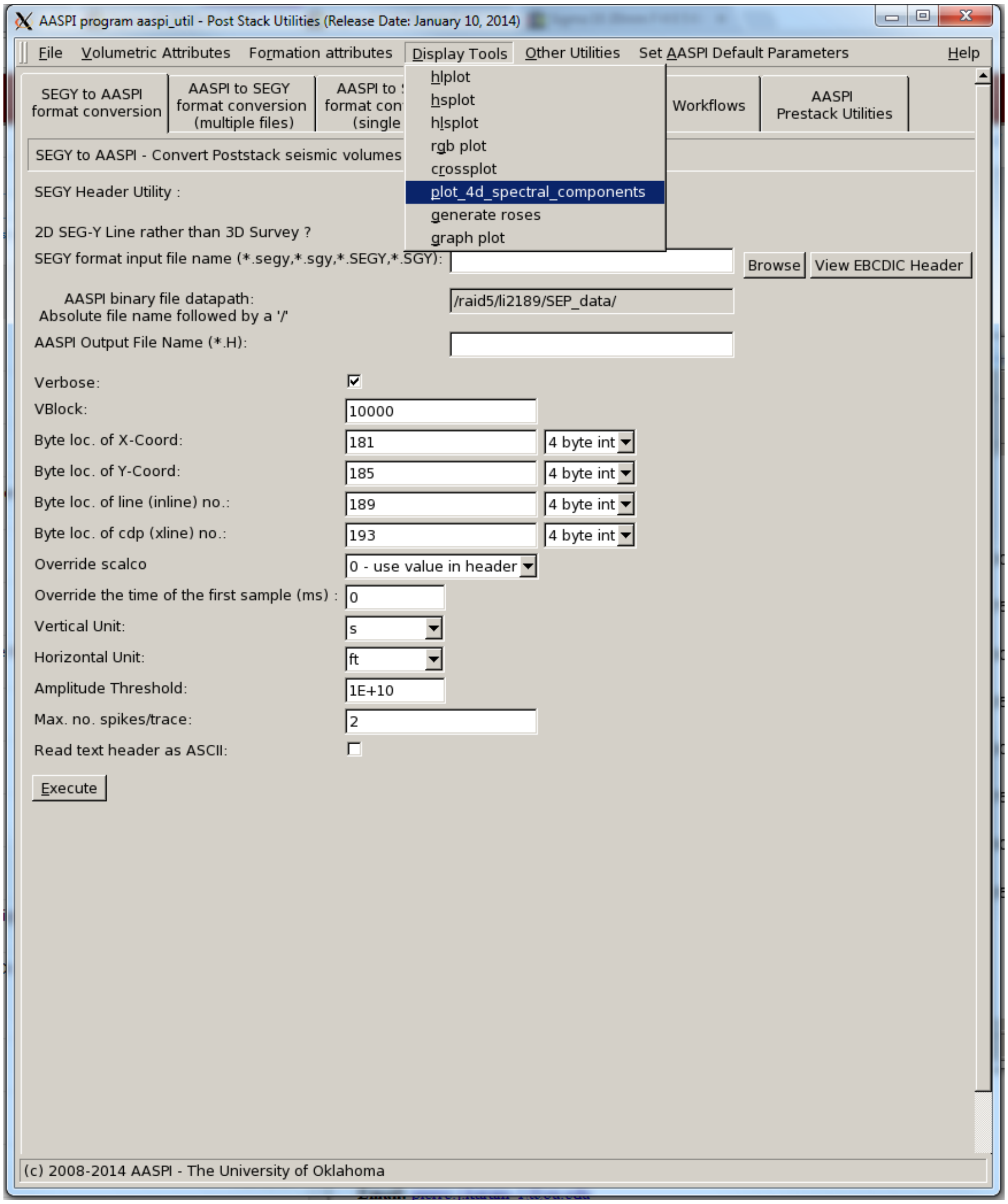
We can also plot the corresponding frequency at this value (the peak frequency) which for this case is the *file peak_freq_3d_clssa_d.H* plotted against the *frequency.sep* (magenta-red-yellow-green_cyan_blue) color bar:



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

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:


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 S format con (single
Workflows
AASPI Prestack Utilities
SEGY to AASPI - Convert Poststack seismic volumes
Browse
View EBCDIC Header
Execute
(c) 2008-2014 AASPI - The University of Oklahoma

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

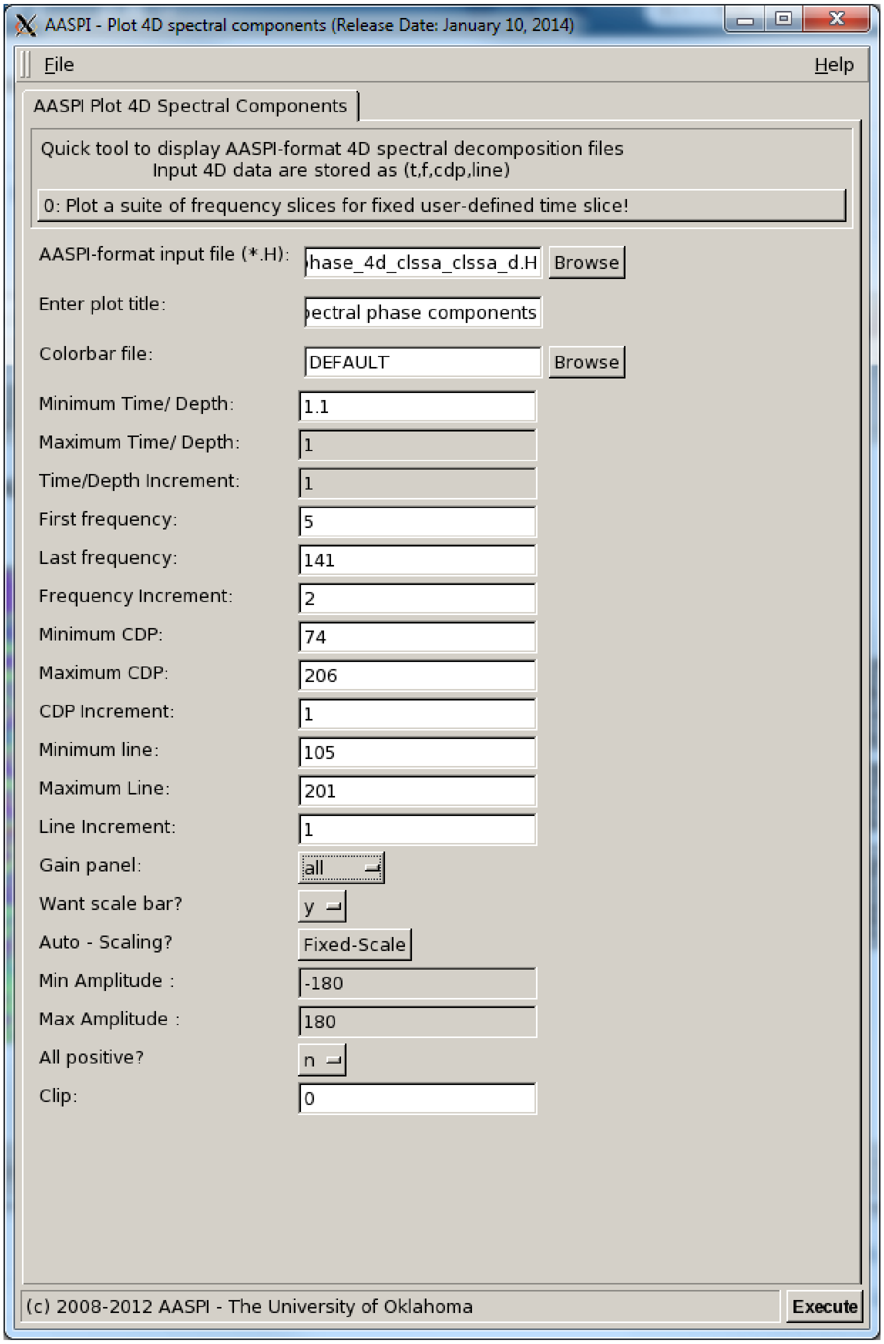
Horizontal Unit: ft

Amplitude Threshold: 1E+10

Max. no. spikes/trace: 2

Read text header as ASCII: □

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_phase_4d_clssa_d_0*.H



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)

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

AASPI-format input file (*.H): ›hase_4d_clssa_clssa_d.H

Enter plot title: ›ectral phase components

Colorbar file: DEFAULT

Minimum Time/ Depth: 1.1

Maximum Time/ Depth: 1

Time/Depth Increment: 1

First frequency: 5

Last frequency: 141

Frequency Increment: 2

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 : -180

Max Amplitude : 180

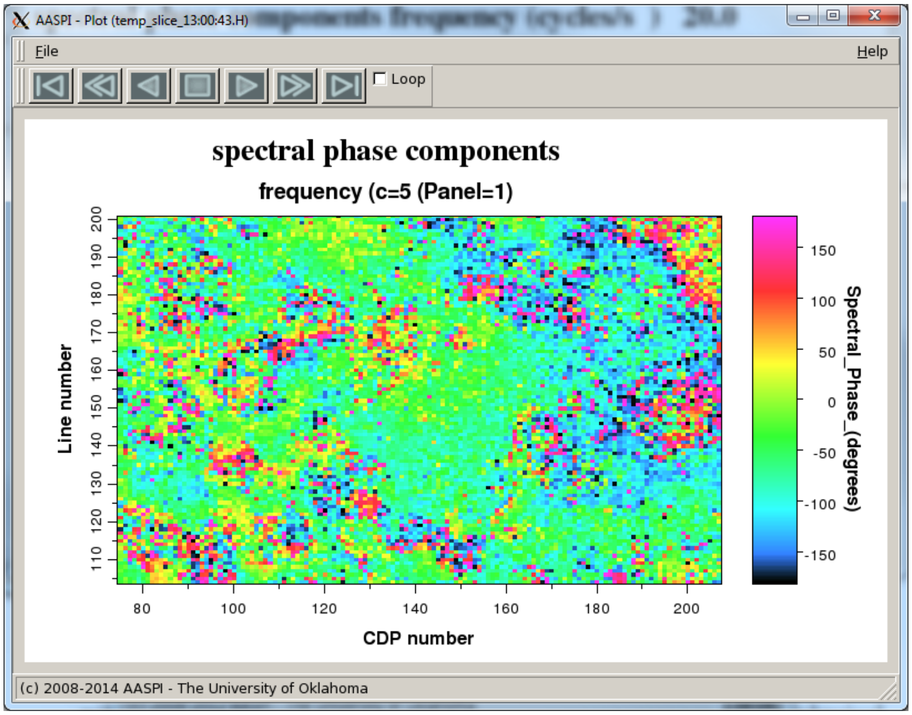
All positive? n

Clip: 0

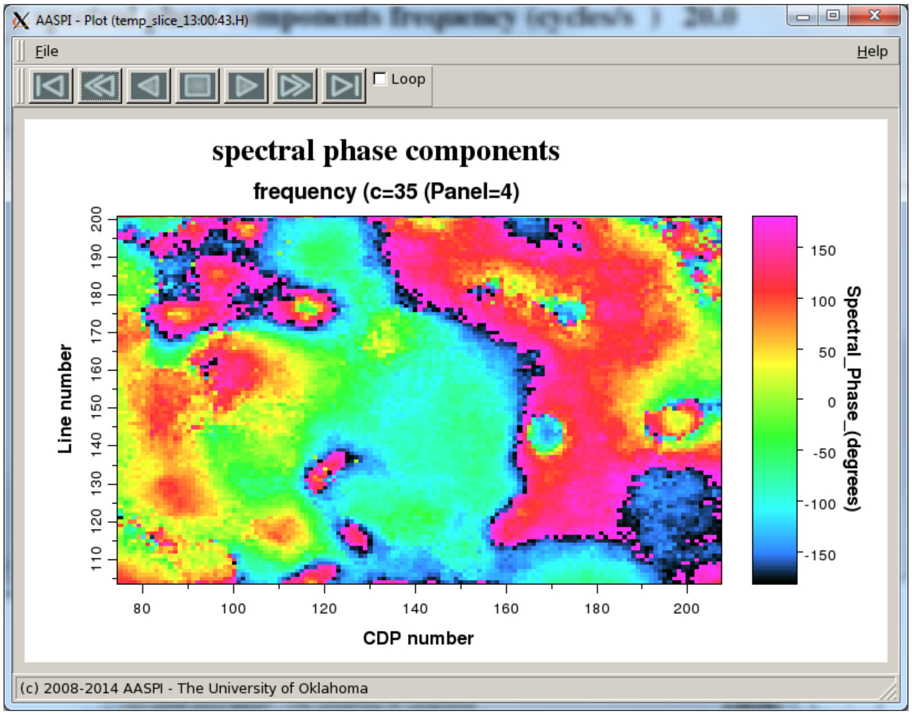
(c) 2008-2012 AASPI - The University of Oklahoma

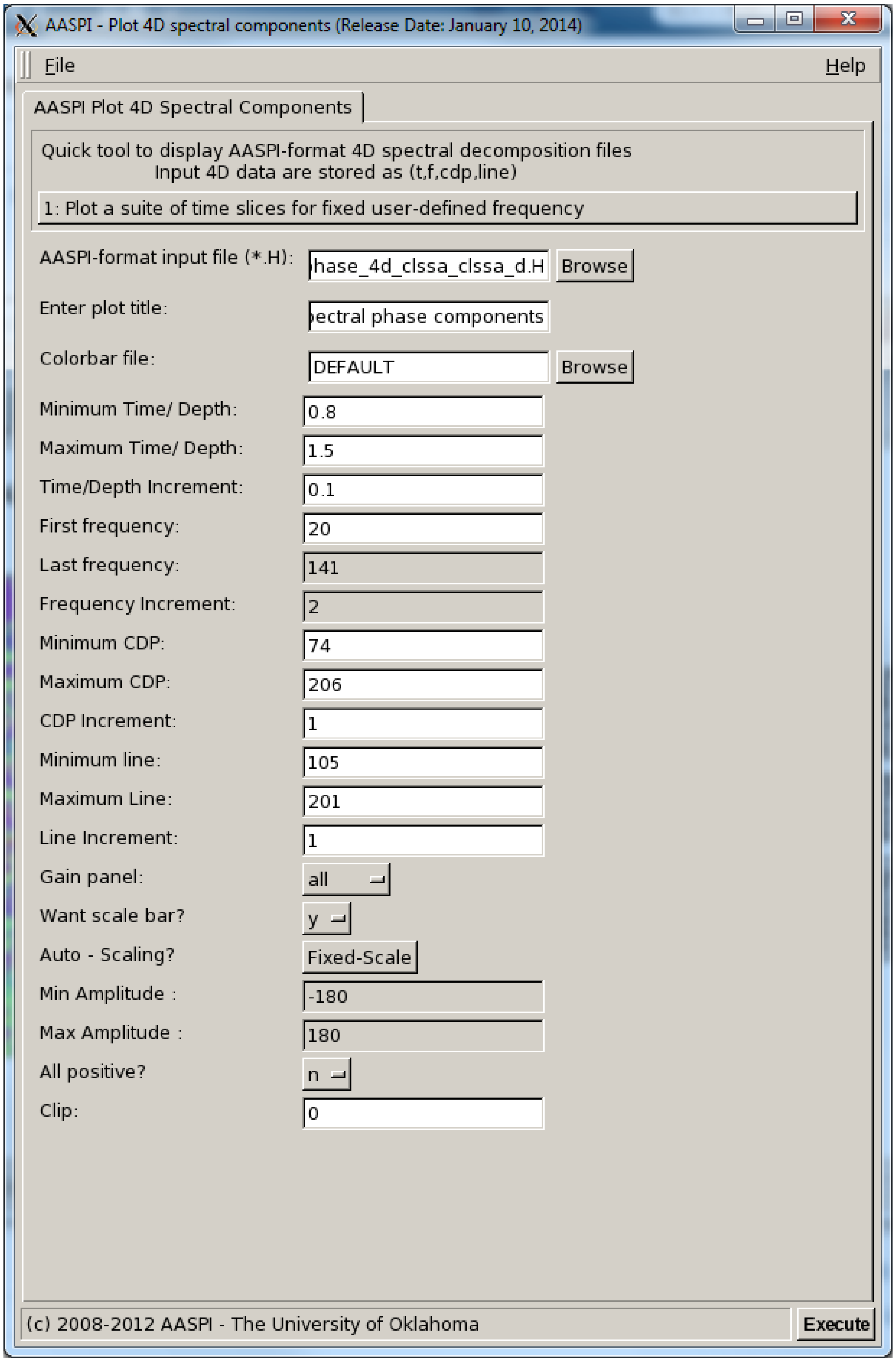
The upper selection bar allows me to plot a constant frequency section or a constant time section. I’ve chosen the 1.1s time slice. I choose the *energy.sep* color bar and obtain these slices of different frequencies at 1.1s time slice:

First, 5 Hz component at 1.1s:



Then, 35 Hz component at 1.1s:


AASPI - Plot (temp_slice_13:00:43.H)
File
Help
Loop
spectral phase components
frequency (c=35 (Panel=4)
Line number
110
120
130
140
150
160
170
180
190
200
80
100
120
140
160
180
200
CDP number
150
100
50
0
-50
-100
-150
Spectral_Phase_(degrees)
(c) 2008-2014 AASPI - The University of Oklahoma



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)

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

AASPI-format input file (*.H): hase_4d_clssa_clssa_d.H Browse

Enter plot title: ectral phase components

Colorbar file: DEFAULT Browse

Minimum Time/ Depth: 0.8

Maximum Time/ Depth: 1.5

Time/Depth Increment: 0.1

First frequency: 20

Last frequency: 141

Frequency Increment: 2

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 : -180

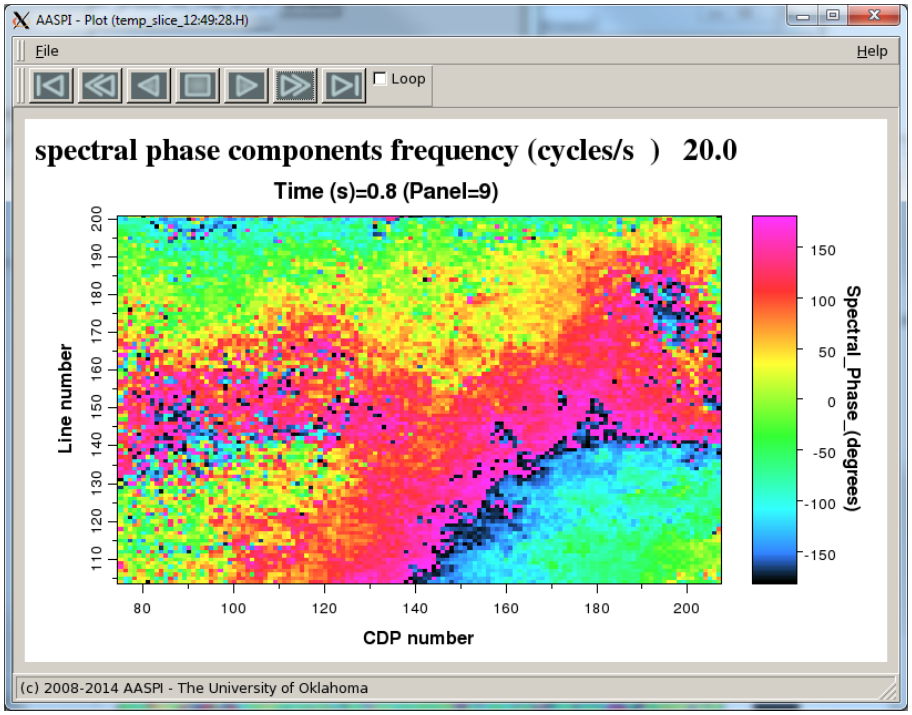
Max Amplitude : 180

All positive? n

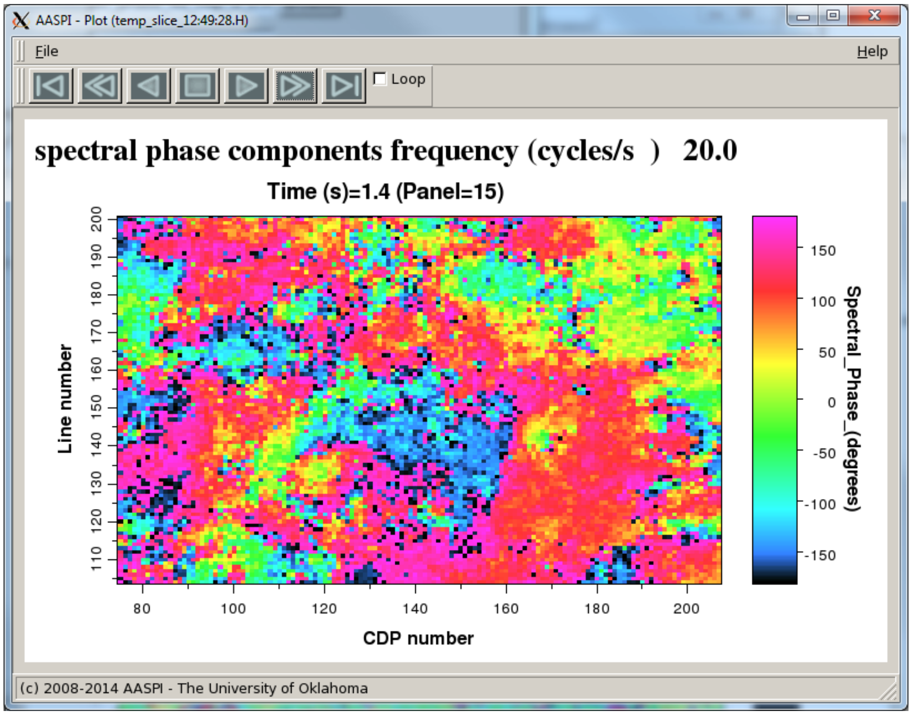
Clip: 0

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

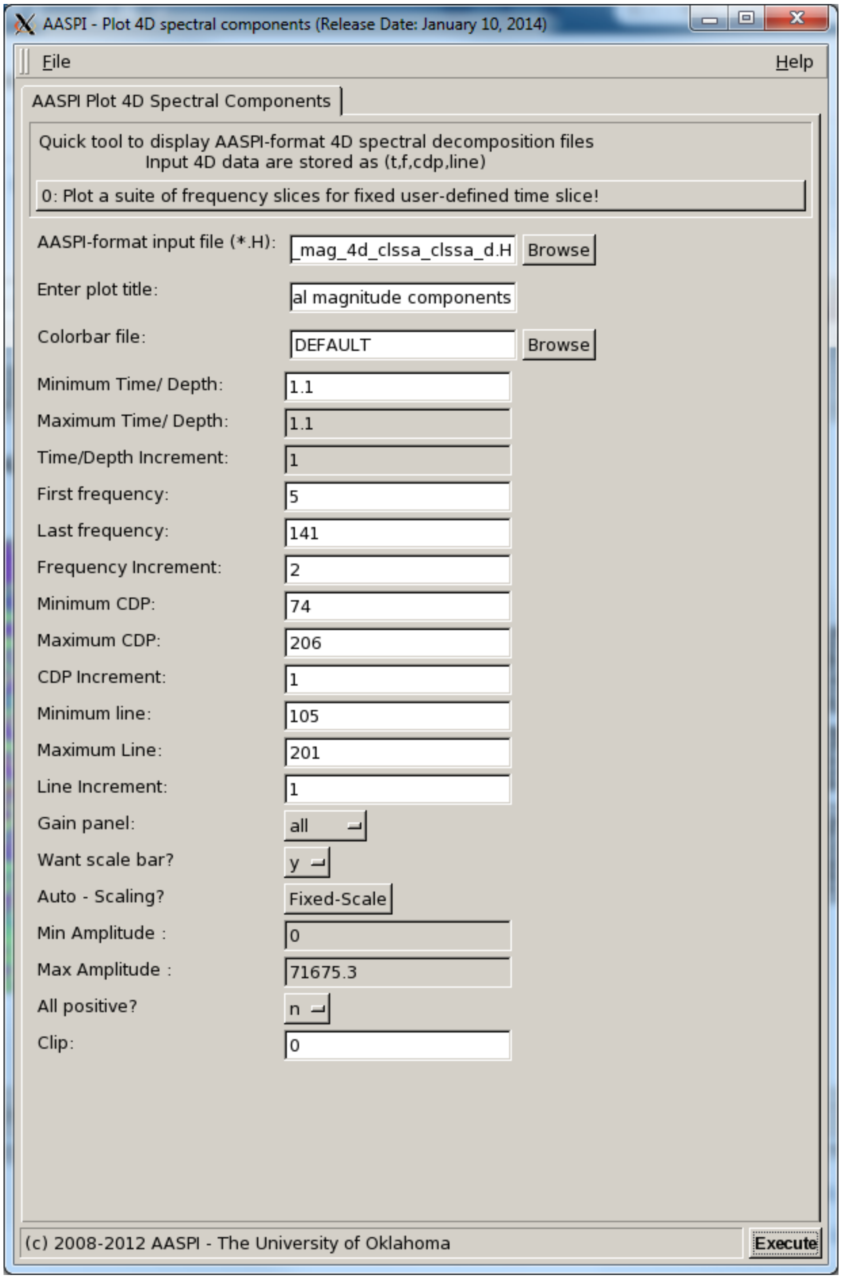
First, 0.8s time slice of 20 Hz component:



Then, 1.4s time slice of 20 Hz component:



I can also generate similar plots of the *spec_mag_4d_clssa_0.H* 4D volume.


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

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

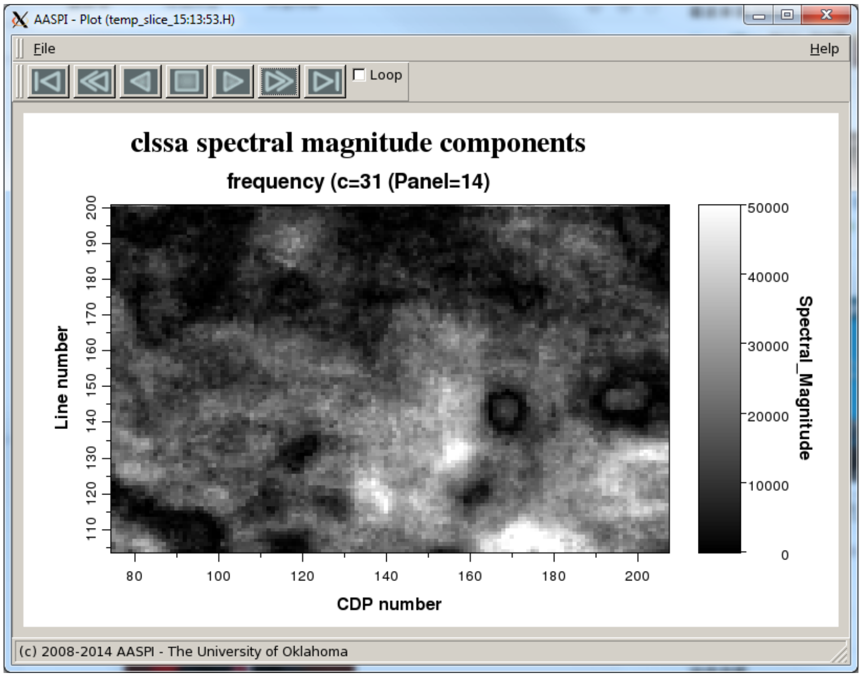
0: Plot a suite of frequency slices for fixed user-defined time slice!

AASPI-format input file (*.H): _mag_4d_clssa_clssa_d.H
Enter plot title: al magnitude components
Colorbar file: DEFAULT
Minimum Time/ Depth: 1.1
Maximum Time/ Depth: 1.1
Time/Depth Increment: 1
First frequency: 5
Last frequency: 141
Frequency Increment: 2
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 : 71675.3
All positive? n
Clip: 0

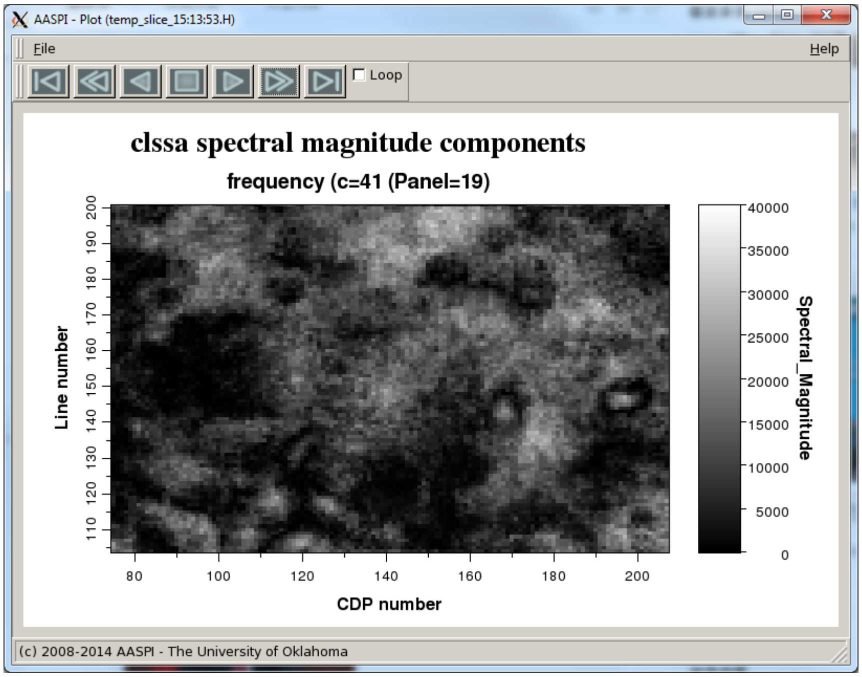
(c) 2008-2012 AASPI - The University of Oklahoma

I display the 31 Hz and 41 Hz spectral magnitude slices at 1.1 s:

First, 31 Hz spectral magnitude slices at 1.1 s:


AASPI - Plot (temp_slice_15:13:53.H)
File
Help
Loop
clssa spectral magnitude components
frequency (c=31 (Panel=14)
Line number
110
120
130
140
150
160
170
180
190
200
80
100
120
140
160
180
200
CDP number
Spectral_Magnitude
0
10000
20000
30000
40000
50000
(c) 2008-2014 AASPI - The University of Oklahoma

Then, 41 Hz spectral magnitude slices at 1.1 s:


AASPI - Plot (temp_slice_15:13:53.H)
File
Help
Loop
clssa spectral magnitude components
frequency (c=41 (Panel=19)
Line number
110
120
130
140
150
160
170
180
190
200
80
100
120
140
160
180
200
CDP number
Spectral_Magnitude
0
5000
10000
15000
20000
25000
30000
35000
40000
(c) 2008-2014 AASPI - The University of Oklahoma

References

Portniaguine, O., and J. P. Castagna, 2004, Inverse spectral decomposition: 74th Annual International Meeting, SEG, Expanded Abstracts, 1786–1789.

Puryear C. I., O. N. Portniaguine, C. M. Cobos, and J. P. Castagna, 2012, Constrained least-squares spectral analysis: Application to seismic data. Geophysics, **77**, V143–V167.