

SpectraShortener (OSX) Report

7.8.2003

R. Long

The purpose of the SpectraShortener program is to shorten Cross Spectra (CS and CSS) files for more efficient storage. This program can reduce each spectral value within a Cross Spectra file by .01 decibels (dB), .05 dB, .10 dB, .20 dB, .25 dB, .50 dB, or 1.0 dB. The amount of shortening can be selected in the GUI displayed when “Preferences” is selected from the Edit pull-down menu of SpectraShortener. Once preferences are chosen, CSS files can be dropped on SpectraShortener; these files are then reduced to “CSRs”. CSRs are the reduced versions of CSS files; they can then be stored for access at a later date. When the CSS files are needed, the CSRs can be accessed and dropped onto the SpectraShortener window to produce a new CSS file.

SpectraShortener was tested on a week’s worth of data from 32-range cell CSS files (~628KB each), 63-range cell Bistatic CSS files (~1.2MB), and 255-range cell Bistatic CSS files (~10MB). It was found that by reducing spectral values in a CSS file, an associated slight loss of data is to be expected in the new shortened CSS file. This loss is minimal for smaller amounts of shortening. For example, when a file is shortened by .01dB, a maximum of .23% data loss will occur (generally, data loss is lower at ~.10-.15%), although the overall file size will be reduced by nearly half (see Table I for percentage errors and file sizes associated with shortening).

Table I: Examples of CSS file size reduction and expected percent differences

Amount of Shortening (dB)	Accepted Percent Diff. of spectral values between original and shortened CSS (%)*	32-RC CSS files (~628KB) New file sizes after SShortener Reduction (KB)	SShortener Reduction + Stuffit (KB)	62-RC CSS files (~1.2MB) New file sizes after SShortener Reduction (KB)	SShortener Reduction + Stuffit (KB)	255-RC CSS files (~10MB) New file sizes after SShortener Reduction (MB)	SShortener Reduction + Stuffit (MB)
0.0	0.0	628	548	1200	1000	9.9	8.6
.01	.23	344	248	692	504	5.3	3.9
.05	1.2	236	192	524	400	3.7	3.0
.10	2.3	180	160	372	324	2.8	2.5
.20	4.7	164	136	336	280	2.5	2.1
.25	5.9	164	132	332	268	2.5	2.1
.50	12.2	164	112	332	228	2.5	1.7
1.0	25.9	164	96	332	196	2.5	1.5

The shortened files produced from SpectraShortener (CSRs), can be further reduced by using “Stuffit” (shown in Table I).

Additional tests of SpectraShortener included comparisons between radials derived from original CSS files (for 32-range cell files) and radials derived from shortened files created from SpectraShortener. These comparisons revealed standard deviations* between the two data sets

* See Appendix for mathematical descriptions of the study.

indicating expected differences in radial velocities for shortened versus unshortened files (see Table II).

Table II: Expected Radial Velocity differences between unshortened and shortened radials

Amount of Shortening (dB)	Standard Deviation (Radz) between shortened and unshortened radial velocities over one week of data
.01	2.383 cm/s
.10	5.135 cm/s
1.0	11.570 cm/s

Tests reveal that .01dB shortening reduces file sizes by nearly half, while maintaining 99.77% accuracy with raw spectral values obtained from the original file, and less than 2.4 cm/s standard deviation in the compared radial velocities. Larger amounts of shortening affect file size minimally, but increase standard deviations and decrease accuracy in raw spectral values

* See Appendix for mathematical descriptions of the study.

Appendix

SpectraShortener Study:

An extensive case study was performed on SpectraShortener to evaluate the shortening processes of the program. The case study analyses were focused on two objectives: 1. Determine if the program reduced all (self, real and imaginary) spectral values correctly, and 2. Determine the loss of data accompanying each increment of shortening for one week of standard CSS files consisting of 32 range cells.

Data Sets

To reach these two initial objectives, one week of standard CSS files consisting of 32 range cells, was used for testing. This data set of hourly-averaged CSS files, spanning from November 11, 2000 – November 18, 2000 came from Misquamicut, RI (site MISQ). This data set underwent spectral analyses as well as radial comparisons.

Two other data sets were used to conduct spectral analyses as confirmation of SpectraShortener's abilities with larger CSS files. A week's worth of CSS files was taken from Brant Beach, NJ (site MRS1) to test the larger 1.2MB Bistatic files containing 63 range cells, as well as a week of 10MB bistatic CSS files from Wildwood (site WILD) containing 255 range cells.

Spectral Analyses

A point was picked within the positive Bragg peak (25.625cm/s), the negative Bragg peak (-20.966cm/s), and within noise (-114.149cm/s) for reference points to determine percent differences in spectral values between shortened and unshortened CSS files. Acceptable percent differences for each increment of shortening can be found by this equation:

$$\text{Acceptable Percent Difference} \leq [10^{\wedge (\text{Amt. of shortening (in dB)/10)} - 1] * 100$$

(Between shortened and unshortened CSS files)

Twelve CSS files were picked randomly throughout the week of data to check percent differences at the three chosen points. Self and complex spectra were evaluated by taking the raw spectral value at each point, and computing this equation:

$$\text{Percent Difference} \leq \frac{|S_s - S_o|}{\left[\frac{\sqrt{(|S_s|^2 + |S_o|^2)}}{2} \right]}$$

where: S_s = shortened spectral value
 S_o = original spectral value

Results: Spectral Analyses

SpectraShortener proved to deal with self and complex spectral values in an acceptable manner. Typical percent differences calculated from the twelve test files in each data set (i.e. 32 RC, 63 RC, and 255 RC data sets) ranged far below the Accepted Percent Difference values for each increment of shortening.

Radial Comparisons

For further analyses, the original data set from Misquamicut (32 RC CSS files) was processed through to radials as well as the .01dB shortened data set, the .10dB, and the 1.0dB data set. By processing these data sets through to radials, a “Radial Comparison” could be carried out in SeaDisplay.

A Radial Comparison in SeaDisplay compares radial velocities produced from the same site. For this additional study of SpectraShortener, a comparison was done between radial velocities (Radz) in the original data set and each shortened data set (.01dB, .10dB, and 1.0dB) to determine the changes in radial velocities brought about by shortening CSS files. A radial comparison was also done on five separate hours within the week in order to compare the results of acquired from the week comparison and the results from the five specific hours within the week. SeaDisplay produces the “square root of the sum of the squares” as a standard deviation of radial velocities between the two data sets.

Results: Radial Comparisons

Radial comparisons showed that by shortening a week’s worth of CSS files by .01dB, one could expect the radial velocity standard deviation to be 2.383 cm/s. When comparing five individual hours of data from this time period, results showed similar results with an average difference of 2.414 cm/s (see Table A for Radial Comparison results).

Table A: Results from Radial Comparison of 32 RC data set

Amount of Shortening (dB)	Average of the Avg. Difference over five individual CSS files	Standard Deviation (Radz) from Radial Comparison in SeaDisplay over one week
.01	2.414 cm/s	2.383 cm/s
.10	4.790 cm/s	5.135 cm/s
1.0	12.003 cm/s	11.570 cm/s

Recommendations:

From the results obtained in this study of three different types of CSS files, SpectraShortener can be useful for reducing the file sizes of CSS files without significantly corrupting radial velocities from their original values. It has been shown that .01 dB shortening results in radial velocities differing up to 2.4 cm/s from the original. The file size of each CSS file, however, will be reduced by nearly half.

If CSS files are reduced more than .01dB, however, one can expect significant changes in radial velocities (5 cm/s or greater) produced from the shortened file.