$\mathcal{AIPS}$ Memo 126 Revised

Spectral index fitting in $\mathcal{AIPS}$

Eric W. Greisen

April 24, 2025

Abstract

$\mathcal{AIPS}$ has had the task SPIXR to fit the spectral index of image cubes since 2005. Beginning in the 31DEC24 release, a new interactive task called TVSPX became available. It does the same fitting as SPIXR but then allows the user to examine the results and to try to improve upon them. This Memo will describe the usage of this new task.

1 Introduction

Spectral index has the definition in $\mathcal{AIPS}$

$$\log(T(x, y, f)) = \log(T(x, y, f_0)) + S(x, y)\log(f/f_0) + C(x, y)(\log(f/f_0))^2$$

where the logarithms are base 10, $T(x, y, f)$ is the brightness or flux density at a particular spatial coordinate and frequency, x and y are the spatial coordinates, f is frequency, f_0 is a reference frequency, $S(x, y)$ is the "spectral index" and $C(x, y)$ is the "spectral curvature".

$\mathcal{AIPS}$ has had the ability to fit images for spectral index for a very long time. In the early days of the VLA, this took the form of an operation in task COMB to compute the spectral index from two continuum images. In 2005, the task SPIXR was written. It takes a spectral-line cube that has been transposed to make the first axis be the spectral axis. That axis has units of frequency in Hz and can either be a regularly gridded function or an irregularly spaced axis whose values are described in an FQ table. As will be described below, the task has a number of adverbs which allow the user to reject anomalous solutions. These solutions are simply blanked in the output images of T, S, and C. In many cases, this is a quite reasonable result due to low signal levels and the like. However, some cubes contain many good spectral channels with some channels adversely affected by RFI or poor calibration and imaging. The new task TVSPX was written to allow the user to locate pixels for which good solutions are possible if some of the spectral channels are omitted.

2 Spectral index fitting: SPIXR

The usual INNAME *et al.* adverbs define the three-dimensional image cube to be fit. The cube is required to be transposed so that the frequency axis is first. That axis may be called FREQ, FQID, or SEQ.NUM.. The usual OUTNAME, *et al.* adverbs specify the name, sequence number, and disk of the output images. The OUTCLASS is controlled by the task as described below. Adverbs BLC and TRC may be used to limit the extent of the cube that is fit and the size of the output images. FLUX specifies the lowest brightness (usually in Jy/beam) used in the fit. REFREQ specifies f_0 in GHz where 0 says to use 1 GHz and less than 0 says to use the header frequency. OPTYPE = 'CURV' tells the task to solve for curvature (C) as well as spectral index. Otherwise only spectral index (S) is computed. PBPARM is used to adjust the FLUX cutoff for primary beam with PBPARM(1) = 0 turning the correction off. Output pixels are flagged if

$$
\begin{aligned}
N(x,y) &< \texttt{DPARM(1)} \\
S(x,y) &< \texttt{DPARM(2)} \\
S(x,y) &> \texttt{DPARM(3)} \\
T(x,y,f_0)/T_{min} &< \texttt{DPARM(4)} \\
T(x,y,f_0)/T_{max} &> \texttt{DPARM(5)} \\
T(x,y,f_0) &< \texttt{DPARM(6)} \\
T(x,y,f_0) &> \texttt{DPARM(7)} \\
C(x,y) &< \texttt{DPARM(8)} \\
C(x,y) &> \texttt{DPARM(9)} \\
\sigma(T(x,y,f_0)) &> \texttt{CPARM(1)} \\
\sigma(S(x,y)) &> \texttt{CPARM(2)} \\
\sigma(C(x,y)) &> \texttt{CPARM(3)}
\end{aligned}
$$

T_{min} is the minimum brightness included in the fit at the present (x, y) and T_{max} is the maximum brightness in that fit. $N(x, y)$ is the number of data samples included in the fit, *i.e.*, the number with brightness $>$ beam corrected `FLUX`. The $\sigma()$ are the uncertainties in the fit. Default values for the `DPARM` and `CPARM` are set so that no pixels are flagged if there are enough spectral channels above `FLUX` to do a fit.

The 7 output images have class name `'B TEMP'` for $T(x, y, f_0)$, `'D BT'` for $\sigma(T(x, y, f_0))$, `'SP CNT'` for $N(x, y)$, `'SPIX'` for $S(x, y)$, `'D SPIX'` for $\sigma(S(x, y))$, and, optionally, `'SPCU'` for $C(x, y)$ and `'D SPCU'` for $\sigma(C(x, y))$.

3 Interactive task `TVSPX`

Beginning in $\mathcal{AIPS}$ version `31DEC24`, the task `TVSPX` allows the user to attempt to improve the results of `SPIXR` before the final images are written to disk. It has exactly the same adverbs as `SPIXR` used in the same way. The flagging caused by `DPARM` and `CPARM` is applied to any fit done in `TVSPX` including in the interactive phase. `TVSPX` begins by computing the output images exactly like `SPIXR`.

At that point, the $\mathcal{AIPS}$ TV screen shows a two-column menu as illustrated in Figure 1. The right-hand side is

`SHOW IMAGE NP`	Enter image interaction with image of number of channels used in fit
`SHOW IMAGE BR`	Enter image interaction with brightness image
`SHOW IMAGE EBR`	Enter image interaction with uncertainty in brightness image
`SHOW IMAGE SP`	Enter image interaction with spectral index image
`SHOW IMAGE ESP`	Enter image interaction with uncertainty in spectral index image
`SHOW IMAGE CU`	Enter image interaction with spectral curvature image
`SHOW IMAGE ECU`	Enter image interaction with uncertainty in spectral curvature image

To select one of these options in all menus, move the TV cursor to the option which will cause it to change color. Pressing button D causes helpful information to appear in the window in which you are running `AIPS`. Press keyboard letters A, B, or C to select an image to display. This causes an image display window to appear with its own menu and a display of the selected image in whole or in a previously selected sub-window. That screen will be illustrated and described below. But first, let us describe the left-hand menu in the main screen.

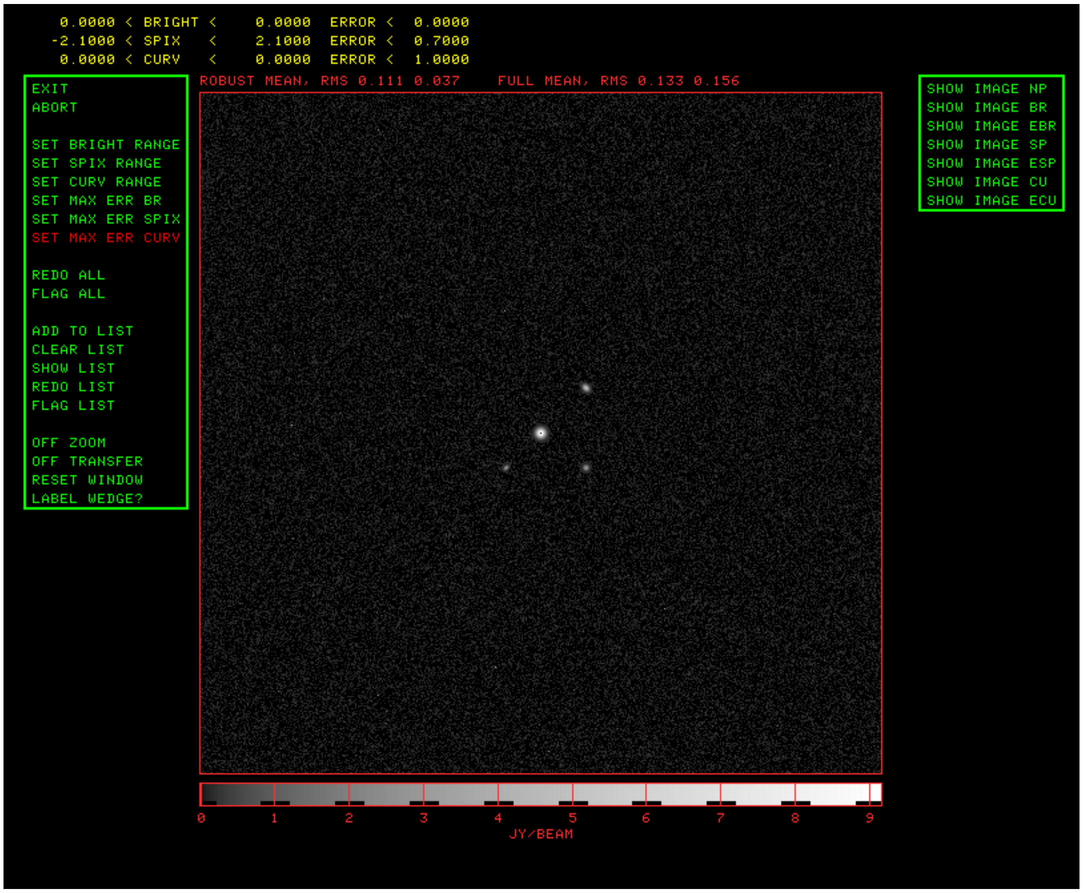

Figure 1: Main interactive window after a `SHOW IMAGE BR`, a `LABEL WEDGE`, a `SET SPIX RANGE`, and two `SET MAX ER` operations. The image is a trivial example produced by new task `SPXMD`.

`EXIT`	Exit `TVSPX`, writing output images
`ABORT`	Exit `TVSPX`, deleting output images
`SET BRIGHT RANGE`	Set range of valid brightness values
`SET SPIX RANGE`	Set range of valid spectral index values
`SET CURV RANGE`	Set range of valid spectral curvature values
`SET MAX ER BR`	Set maximum uncertainty in brightness
`SET MAX ER SPIX`	Set maximum uncertainty in spectral index
`SET MAX ER CURV`	Set maximum uncertainty in spectral curvature
`REDO ALL`	Re-do all solutions which are not okay following the above criteria
`FLAG ALL`	Mark bad all solutions which are not okay
`ADD TO LIST`	Type in output pixel coordinates to add to list
`CLEAR LIST`	To reset the list to empty
`SHOW LIST`	Display pixel coordinates in list
`REDO LIST`	Re-do solutions for all pixels in list
`FLAG LIST`	Flag solutions for all pixels in list
`OFF ZOOM`	Turn of TV zoom
`OFF TRANSFER`	Turn off black & white and color TV enhancements
`RESET WINDOW`	Display full view of current image
`LABEL WEDGE?`	Turn on/off labeling of step wedge
`CHAR MULT`	Set the character size multiplier

This menu enables two forms of editing. The first form edits all pixels which have values outside user-specified ranges. The `SET BRIGHT RANGE` option prompts the user for 2 numbers to be entered in the window in which `AIPS` is being run. If the first is less than the second, values of $T(x, y, f_0)$ less than the first or greater than the second are viewed as suspect. If the first is greater than the second, values of $T(x, y, f_0)$ between the two are viewed as suspect. The `SET SPIX RANGE` and `SET CURV RANGE` are similar and apply to $S(x, y)$ and $C(x, y)$, respectively. The `SET MAX ER` options prompt for a single number which is viewed as the maximum uncertainty that is not viewed as suspect in the specified parameter. These limits are displayed at the top of the screen after they are set. Then the `FLAG ALL` option will flag all "suspect" pixels, while the `REDO ALL` will enter an interactive window for each "suspect" pixel in turn. The range and error limits are reset after `FLAG ALL` and, if not terminated prematurely, after `REDO ALL`.

The second form of editing begins with a list of (x, y) pixels to be edited. `ADD TO LIST` allows the user to type in an x, y pair to be added to the list while `CLEAR LIST` empties the list, and `SHOW LIST` shows the current list in the message window. The image display window allows the list to be increased by selecting pixels with a button push while running `CURVALUE`. Then `FLAG LIST` will mark all pixels in the list as bad and `REDO LIST` will enter the interactive editing window for each entry in the list. The list is cleared after the former and reduced to any pixels not processed in the latter.

The other options turn off any zoom setting, reset any black and white or color enhancements, reset the sub-image selection to the full image, turn on or off the labeling of the step wedge, and, for large TV windows only, change the size of the displayed characters.

When a `SHOW IMAGE` option is selected, a new window appears. At the top a text line in yellow identifies the type of image and the range of pixel values displayed. The image itself is displayed and labeled in red with the robust and full mean and rms of the full image. The displayed image is either the full image or a selected sub-image. The image is accompanied with a step wedge which may or may not be labeled. This screen is illustrated in Figure 2. This screen is accompanied by a menu containing

`RETURN`	Return to the main menus, image stays displayed
`LOAD AS SQ`	Re-load image with square root transfer function
`LOAD AS LG`	Re-load image with log transfer function
`LOAD AS L2`	Re-load image with extreme log transfer function
`LOAD AS LN`	Re-load image with linear transfer function
`SET WINDOW`	Set a sub-image to view
`RESET WINDOW`	Return too viewing the full image
`OFF TRANSF`	Turn off enhancement done with `TVTRANSF`
`OFF COLOR`	Turn off color enhancements
`TVTRANSF`	Black & white image enhancement
`TVPSEUDO`	Color enhancement of numerous sorts
`TVPHLAME`	Color enhancement of flame type, multiple colors
`OFMCOLOR`	Many standard color tables (OFMs) available
`TVZOOM`	Interactive zooming and centering of image
`IMSTAT`	Compute statistics of current sub-window
`CURVALUE`	Display value under cursor, mark pixels for list
`NEXT WINDOW`	Move to next window in large images

The image is initially loaded with a linear transfer function. Only one `LOAD AS` option is displayed which is the next one in sequence. Selecting, for example, `LOAD AS SQ` will reload the image with a square-root transfer function and will change the menu to `LOAD AS LG`. The remaining options are mostly familiar as `AIPS` verbs. `SET WINDOW` allows the user to select a sub-window over which the display scaling and the `FLAG ALL` and `REDO ALL` operations function. `OFF TRANSF` and `OFF COLOR` turn off black and white and color enhancements, respectively. `TVTRANSF` enhances the image in black and white, while `TVPSEUDO`, `TVPHLAME`, `OFMCOLOR` add color with numerous options selected by the buttons and cursor position. `TVZOOM` allows the display to be zoomed during window setting and `CURVALUE` This last displays the pixel coordinates and the

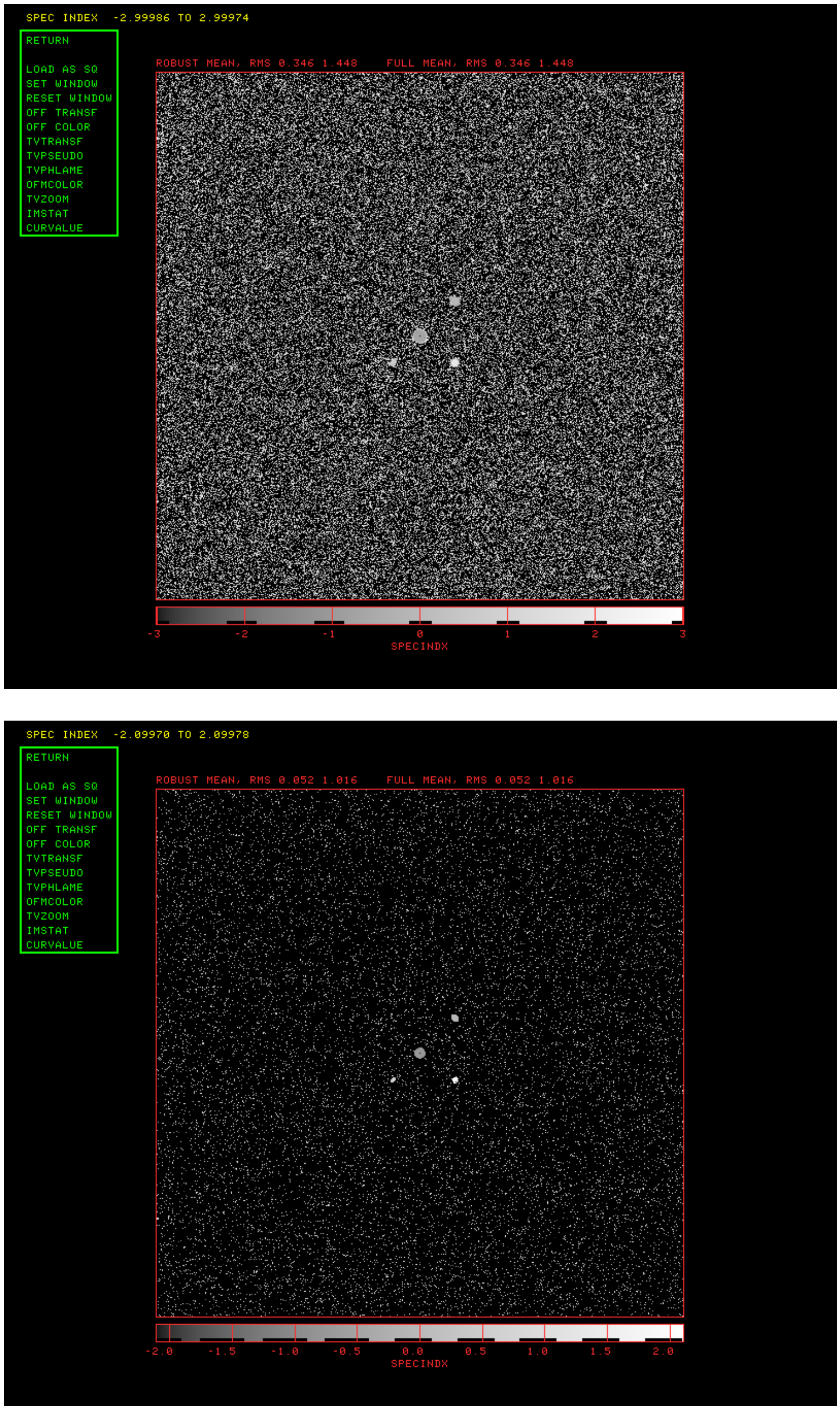

Figure 2: Top: Image display window showing the spectral index image before any interactive flagging. Pixels with spectral index < -3 or > 3 were flagged by `DPARM`, but many bad pixels remain. Bottom: image after a `FLAG ALL` was applied using the limits shown in Figure 1.

image value under the cursor. If one hits buttons A, B, or C while running `CURVALUE`, the selected pixel is added to the edit list. `IMSTAT` computes the full (using all pixels) and robust mean and rms over the selected sub-image. If that is the full image, the statistics displayed with the image are updated. In either case, the statistics are displayed in the message window. Note that the statistics of all 5 or 7 images are computed when the interactive portion of the code is begun and whenever a `REDO` or `FLAG` operation is done. But you must select `IMSTAT` to report the statistics in a sub-image.

Very large images may not be able to fit on your TV. When the image exceeds the size of the TV, the top line will show the sub-image number followed by the parameter type and the `NEXT WINDOW` option will appear. The first sub-image displayed is called number 0 and shows the full image every n'th pixel in X and Y (see message window for n). Sub-image 1 begins at the lower left. Later sub-images move right, then back to the left and up, and so forth until the top right is reached. Every pixel is displayed in these sub-images. Use the `NEXT WINDOW` option to step through the sub-images in a circular fashion.

There is one more screen which appears when doing the `REDO ALL` and `REDO LIST` functions. This screen is illustrated in Figure 3. The spectrum at the selected (x, y) is displayed with image value on the vertical axis and frequency on the horizontal axis. The celestial coordinates of the pixel and the current fit parameters are displayed below the spectrum. A solid line shows the current fit and plus signs show the data. The menu contains

`CHAN RANGE`	To display only a portion of the spectrum
`RESET CHANS`	To display all of the spectrum
`FLAG POINTS`	Select data points to omit from fit
`UNDO FLAGS`	Reset all current flagging
`NEW FIT`	Try a new fit
`GOOD`	Stop this screen return current fit values
`BAD`	Stop this screen, flag this pixel
`QUIT`	Stop the `REDO` option now

The `CHAN RANGE` operation allow you to display only a portion of the spectrum to make flagging of that portion easier. The cursor controls the window selection. Hit buttons A or B to switch to the other end of the window and buttons C or D to select the window and display only that portion. `RESET CHANS` reverts to displaying the full spectrum. The `FLAG POINTS` operation shows at the top left of the screen the data value and frequency under the cursor. Hit buttons A or B to mark the nearest data sample "flagged". After marking all desired samples, hit buttons C or D to return to the present menu. `UNDO FLAGS` simply restores the initial data points. `NEW FIT` takes the data samples minus the flagged samples and repeats the basic spectral index fit. Note that the results are compared to the limits set in adverbs `DPARM` and `CPARM` which may flag the solution. A good solution is plotted on the screen, a flagged solution leaves the plot alone but unflags all flagged data samples. When tired of this pixel, select `GOOD` to return the current fit parameters for this pixel or `BAD` to flag this pixel. `QUIT` tells both `REDO` operations to stop abruptly. On `QUIT`, any pixels not yet processed remain in the list and any limits remain set.

4 Interactive task `LISPX`

`TVSPX` is designed to fit spectral indices to full image cubes. At times, users find themselves with a few spectra taken at one or more celestial coordinates rather than full images. `LISPX` is a new task to assist in this situation. It reads a text file giving one or more spectra with the first column of data being the channel frequencies and the second column being the data values. An optional third column, frequently used for uncertainty, is read and written but not otherwise used. The units of these data columns are arbitrary but must be consistent with the adverb values.

The input adverbs to this task are similar to those of `TVSPX` and `SPIXR`. `INFILE` is required to specify the

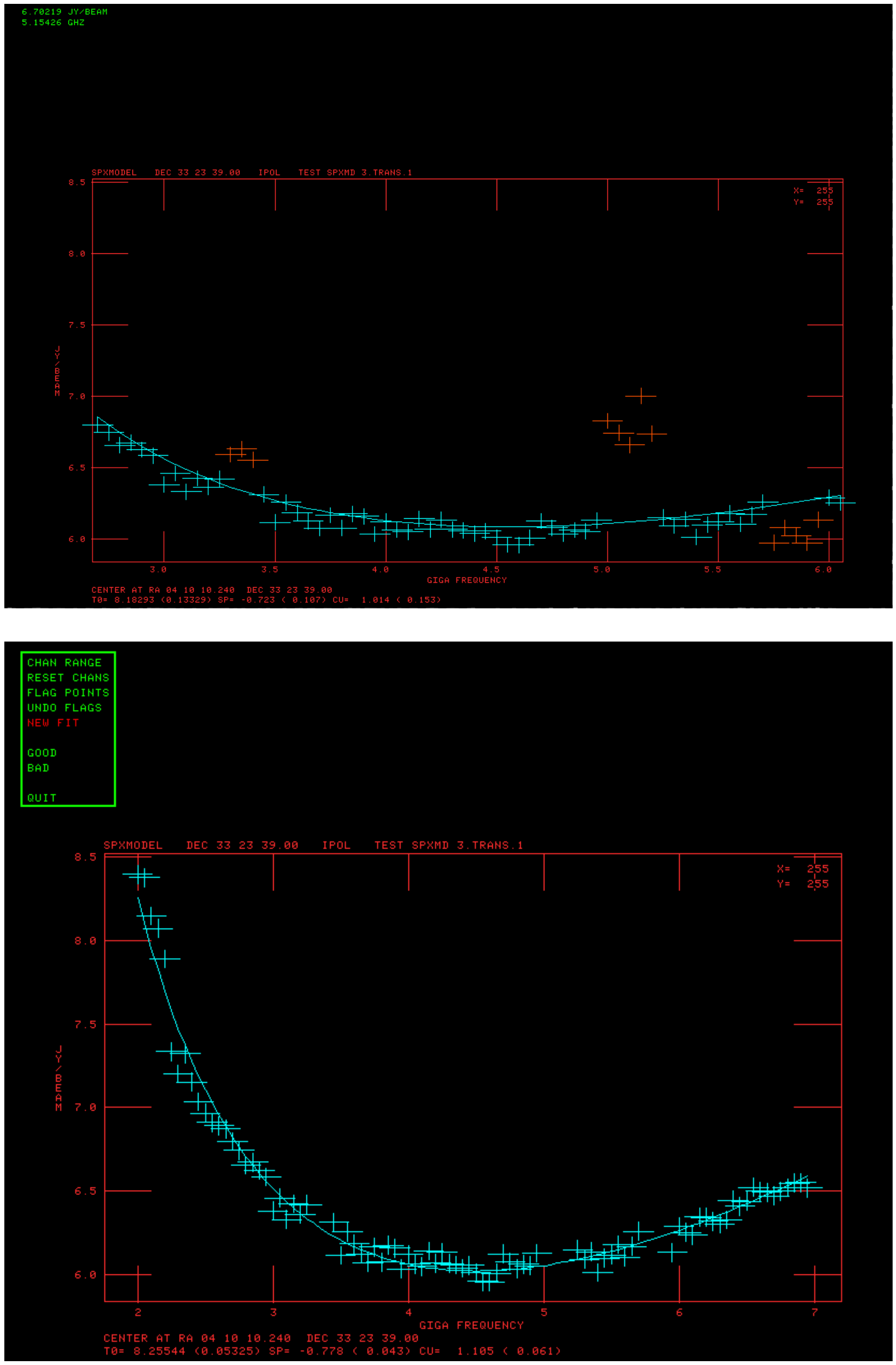

Figure 3: Top: Spectrum at image center after a CHAN RANGE to make the data samples more distinct and showing bad samples already flagged by (and still in) FLAG POINTS with the initial fit to all samples. Bottom: spectrum after RESET CHANS and NEW FIT. SPXMD can offset specified channel ranges, simulating bad gains in some spectral windows, an option that was used here.

data file to be read. Each spectrum in the file begins with a line

`SPECTRUM 'identifying string'`

where SPEC must appear in the first four characters and the identifying string is any character string beginning with a single quote (') to annotate the particular spectrum. The ending quote mark is optional. OUTFILE is an optional file to which the solutions are written along with the spectra including the optional third column and additional columns giving the computed value fit for that channel and the difference between the data and the fit. The remaining adverbs are used in the same ways as those described above for SPIXR and TVSPX. FLUX is the minimum spectral value to be used in the fit and REFREQ is the reference frequency. These two numbers must be given in the same units as the data in INFILE. OPTYPE may be used to request a fit for curvature and DPARM and CPARM are used to flag bad solutions.

After LISPX reads in the data and does a fit for each spectrum, the TV displays the minimal menu:

EXIT	Exit LISPX, writing output file
ABORT	Exit LISPX, with no output file
REDO ALL	Re-do all solutions

EXIT is the normal closing of the task, allowing the answers to appear in the message window and the output file, if any, to be written. ABORT exits the task without these outputs. REDO ALL loops through all spectra showing the options described above in the section on TVSPX and illustrated in Figure 3.

5 Spectral index image simulation with SPXMD

The collection of data simulation tasks was found to be incomplete when TVSPX was created. Therefore, a new task named SPXMD was created to make model image cubes with spectral index and spectral curvature. The task can start with an image cube specified by the INNAME, *et al.* adverbs along with BLC and TRC and scaled by FACTOR. Alternatively, the output image parameters may be set with adverbs COORDINA, IMSIZE, and CELLSIZE. In both cases the reference frequency is set by REFREQ (with sensible defaults). The image noise level is set by FLUX. The frequency of spectral channel one is set by APARM(1), the frequency increment by APARM(2), and the number of frequencies by APARM(3). Adverbs FPARM, RPARM, VPARM, and DOFIT may be used to set up to 30 channel ranges in which the amplitude is scaled by a specified factor and the noise level is scaled by a second specified factor. Up to 9999 "sources" may be described in a text file pointed to by INLIST. In that file, each source is given a peak brightness, (x, y) pixel position, 3 width parameters, a type, a spectral index, and a spectral curvature. Types allowed are point, Gaussian, uniform disk, uniform rectangle, optically thin sphere, and exponential.