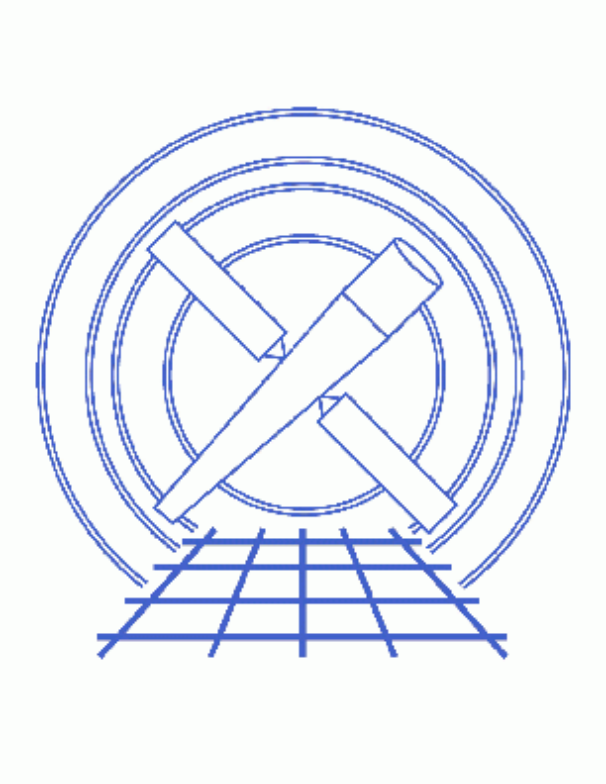
Measure Grating Dispersion Distance



CIAO 3.4 Science Threads

Table of Contents

Measure Grating Dispersion Distance

CIAO 3.4 Science Threads

Overview

Last Update: 1 Dec 2006 − reviewed for CIAO 3.4: no changes

Synopsis:

It can be useful to visualize the wavelengths or energies directly on the sky image of a grating observation, e.g. to see what the chip coverage is or where contaminating sources lie. The region file created by `tg_create_mask` is a spatial filter that marks the source and grating arm(s); it does not indicate the wavelengths or energies. It is necessary to create a separate file which annotates the dispersion distance along the grating arm in the desired units.

Purpose:

To use a S−Lang function to create an ASCII region file that labels distances along the grating arm in wavelength or energy units.

Read this thread if:

you are working with an ACIS or HRC grating observation and would like to mark the dispersion distance on an image of the data.

Related Links:

- Analysis Guide for Chandra High Resolution Spectroscopy: an in−depth discussion of grating analysis.

Proceed to the HTML or hardcopy (PDF: A4 | letter) version of the thread.

Getting Started

Sample ObsID used: 1010 (ACIS−S/HETG, Capella)

File types needed: evt2

The `tg_scale_reg` script requires a FITS file with the REGION block (created by `tg_create_mask` and appended to the file by `tg_resolve_events`), which may be either the evt1a or evt2 file. The Basic Script Syntax section also explains how to manually supply the information if the regions are not available.

Important note for multiobi users

If you are working with a multiobi dataset (not sure? Read the why topic), the ROLL_NOM value in the merged evt2 file may not produce truly accurate results from this script. For the most reliable grating scale, run tg_scale_reg.sl on each ***evt1a*** dataset. It is also possible to use the command line method, substituting the ROLL_PNT header keyword value where ROLL_NOM is used in the example.

Download the script

This thread uses the tg_scale_reg and tg_scale_reg.sl scripts (tg_scale_reg is an slsh wrapper that allows tg_scale_reg.sl to be run from the command line). The most recent version of these scripts is v1.3 (04 August 2003) and v1.4 (12 December 2005), respectively:

```
unix% grep Id `which tg_scale_reg`
% $Id: tg_scale_reg,v 1.3 2003/08/04 16:56:42 dburke Exp $

unix% grep Version $ASCDS_CONTRIB/share/slsh/local-packages/tg_scale_reg.sl
% Version: 1.4 (12 December 2005)
```

Note that $ASCDS_CONTRIB/share/slsh/local-packages/ is the default path in the standard CIAO scripts installation; see the Scripts page for more information. ***Please check that you are using the most recent version before continuing.*** If you do not have the scripts installed or need to update to a newer version, please refer to the Scripts page.

Optional: Using the script in Sherpa

The script is written with general S-Lang commands and may be run on the command line − ***as shown in this thread*** − or in either *Sherpa* or *ISIS*.

The function can be made available to *Sherpa* using the evalfile command:

```
sherpa> evalfile("tg_scale_reg.sl")
1
```

Executing the script without any options will print a usage message. For information on using scripts in *Sherpa* see the Sherpa and Scripts and Customizing Sherpa with a Resource File threads.

Basic Script Syntax

The parameter file shows the available options::

```
unix% plist tg_scale_reg

Parameters for /home/username/cxcds_param/tg_scale_reg.par

        infile =                  Name of FITS file with obs info (or blank)
       outfile =                  Name of output file
             x =                  x source pos in sky coords (for infile='')
             y =                  y source pos in sky coords (for infile='')
       rot_ang =                  Roll angle of obs (for infile='')
       grating = leg              Grating arm (for infile='')
```

```
   instrument = acis            Instrument used (for infile='')
      (script = ${ASCDS_CONTRIB}/share/slsh/local-packages/tg_scale_reg.sl) S-Lang script with tg_s
       (scale = angstrom)       Axis units
      (values = )               Comma separated list of marks (or blank)
     (verbose = 0)              Debug Level (0-5)
        (mode = ql)
```

The default value of the `script` parameter is set to work with the standard CIAO scripts installation; see the Scripts page for more information.

It is possible to run this script in two ways: with a FITS region file as input or by supplying all the necessary information on the command line. Both methods are illustrated in the Run `tg_scale_reg` section.

Using a FITS file: only the name of the file with region information (`infile`) and the output filename (`outfile`) are required.

Without an input file: in the case where a FITS file is not available, the information be provided by a series of parameters:

- `outfile`: output filename.
- `x` , `y`: x and y source position, respectively.
- `rot_ang`: rotation angle. This is the roll angle for the observation, which is stored in the `ROLL_NOM` header keyword (or `ROLL_PNT`, see the important note for multiobi users below).
- `grating`: the grating arm. Allowed values are leg, hetg, heg, or meg. If hetg is used, the output file will contain information for both the HEG and MEG arms.
- `instrument`: the intrument used for the observation. Allowed values are acis or hrc.

For both cases: the `scale` and `value` parameters may be used as well; both are optional. The `scale` parameter changes the distance units to energy [keV]; the default setting for the scale is wavelength [Å]. The `value` parameter takes a comma−separated list of the values to be annotated.

Run tg_scale_reg

FITS region file as input

The REGION block appended to `acisf01010N001_evt2.fits` is used to create the output ruler file. The default `scale` and `values` are used.

```
unix% punlearn tg_scale_reg
unix% pset tg_scale_reg infile="acisf01010N001_evt2.fits"
unix% pset tg_scale_reg outfile="wave_1010.reg"
unix% tg_scale_reg
Name of FITS file with obs info (or blank) (acisf01010N001_evt2.fits):
Name of output file (wave_1010.reg):

       ** Found 1 source.

       Output region file "wave_1010.reg" has been created
```

To view the region file, open the event list in ds9:

```
unix% ds9 acisf01010N001_evt2.fits &
```

The extraction regions should be loaded automatically; if they aren't, use "Region−> Load Regions−> acisf01010N001_evt2.fits[REGION]" to do so.

To add the scale, use the "Region −> Load Regions..." menu to select the file. Figure 1 shows the finished image, binned to "block 4."

Supplying the parameter values

There are several pieces of information that need to be obtained before using this method. As mentioned in the Basic Script Syntax section, we need the source position, rotation angle, grating arm, and instrument.

- ***source position:*** the simplest way to find the sky coordinates of the source is to examine the event file in ds9. Figure 2 shows a cross marking the approximate center of the source; double−clicking on the region brings up the region information box from which the position may be read.

  The source position is `(x,y) = (4101,4089)`.
- ***rotation angle:*** the roll angle for the observation is recorded in the `ROLL_NOM` header keyword. If you are working with a multiobi observation you should use the `ROLL_PNT` value instead, as explained in the important note for multiobi users.

  ```
  unix% dmkeypar acisf01010N001_evt2.fits ROLL_NOM echo+
  262.65257360233
  ```

  The rotation angle is 262.65257.
- ***grating arm:*** although most users will know by this point what transmission grating was used, it is also stored in the `GRATING` header keyword:

  ```
  unix% dmkeypar acisf01010N001_evt2.fits GRATING echo+
  HETG
  ```

  For this example, we choose to use only the MEG arm of HETG.
- ***instrument:*** if you are unsure which instrument was used for your observation, check the `INSTRUME` keyword:

  ```
  unix% dmkeypar acisf01010N001_evt2.fits INSTRUME echo+
  ACIS
  ```

  The instrument is ACIS, as indicated by the filename as well (***`acis`***`f01010N001_evt2.fits`).

Now that we have all the information, we can run `tg_scale_reg`; we let the script prompt for the necessary parameters:

```
unix% punlearn tg_scale_reg
unix% pset tg_scale_reg scale=kev
unix% tg_scale_reg
Name of FITS file with obs info (or blank) ():
Name of output file (): meg_en_1010.reg
x source pos in sky coords (for infile='') (): 4101
y source pos in sky coords (for infile='') (): 4089
Roll angle of obs (for infile='') (): 262.65257
Grating arm (for infile='') (leg|getg|heg|meg) (leg): meg
Instrument used (for infile='') (acis|hrc) (acis):

        Output region file "meg_en_1010.reg" has been created
```

Recall that setting `scale=kev` forces the ruler scale to energy instead of wavelength.

We can use the ds9 command−line syntax to load the region file on top of the image:

```
unix% ds9 acisf01010N001_evt2.fits −region meg_en_1010.reg &
```

Figure 3 shows the scale overlaid on the image.

History

14 Dec 2004	updated for CIAO 3.2: path for script
12 Dec 2005	updated for CIAO 3.3: version 1.4 of `tg_scale_reg.sl`
01 Dec 2006	reviewed for CIAO 3.4: no changes

URL: http://cxc.harvard.edu/ciao/threads/ds9_scale/ Last modified: 1 Dec 2006

Image 1: Wavelength scale region file

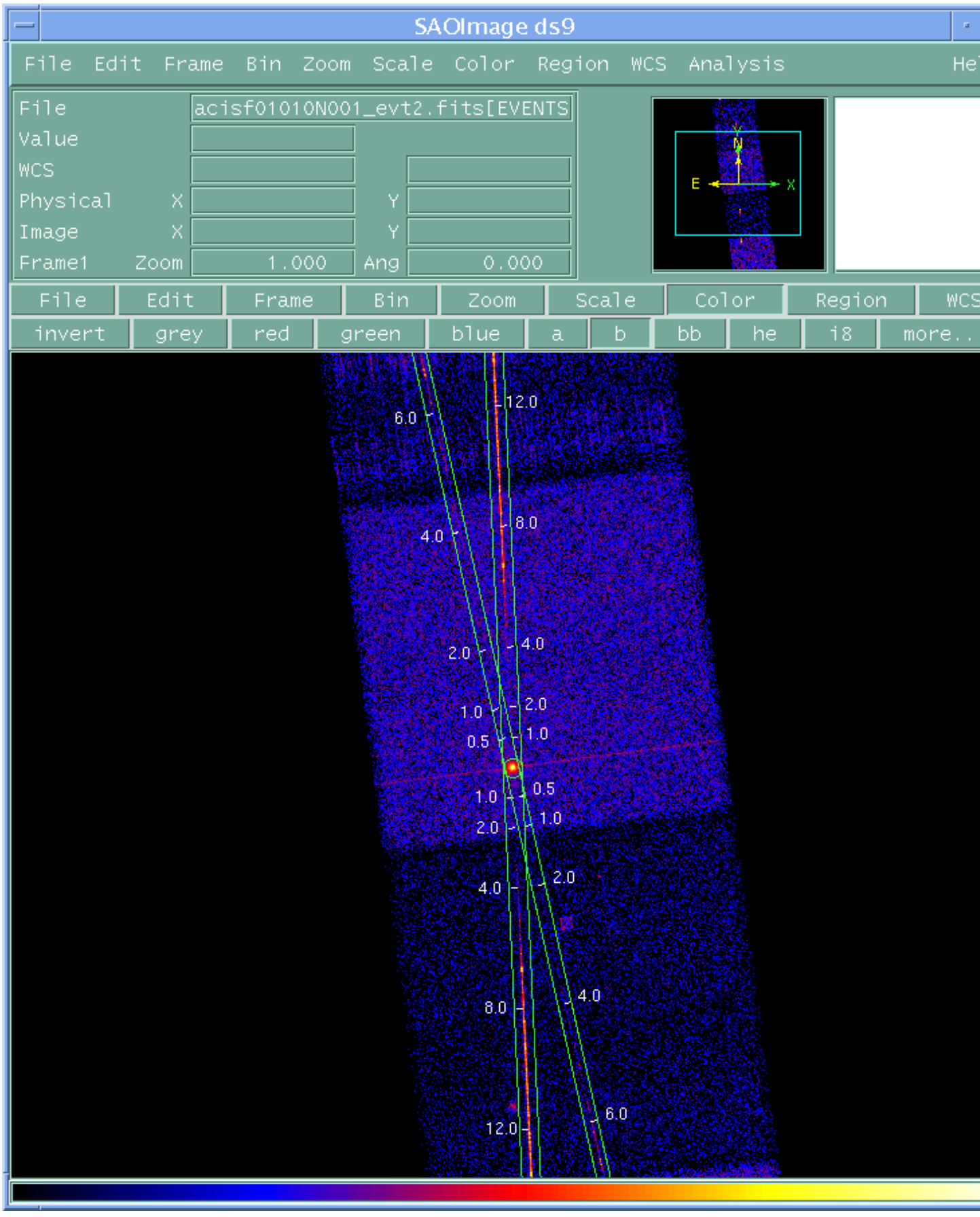


Image 1: Wavelength scale region file

Image 2: Source position marked in ds9

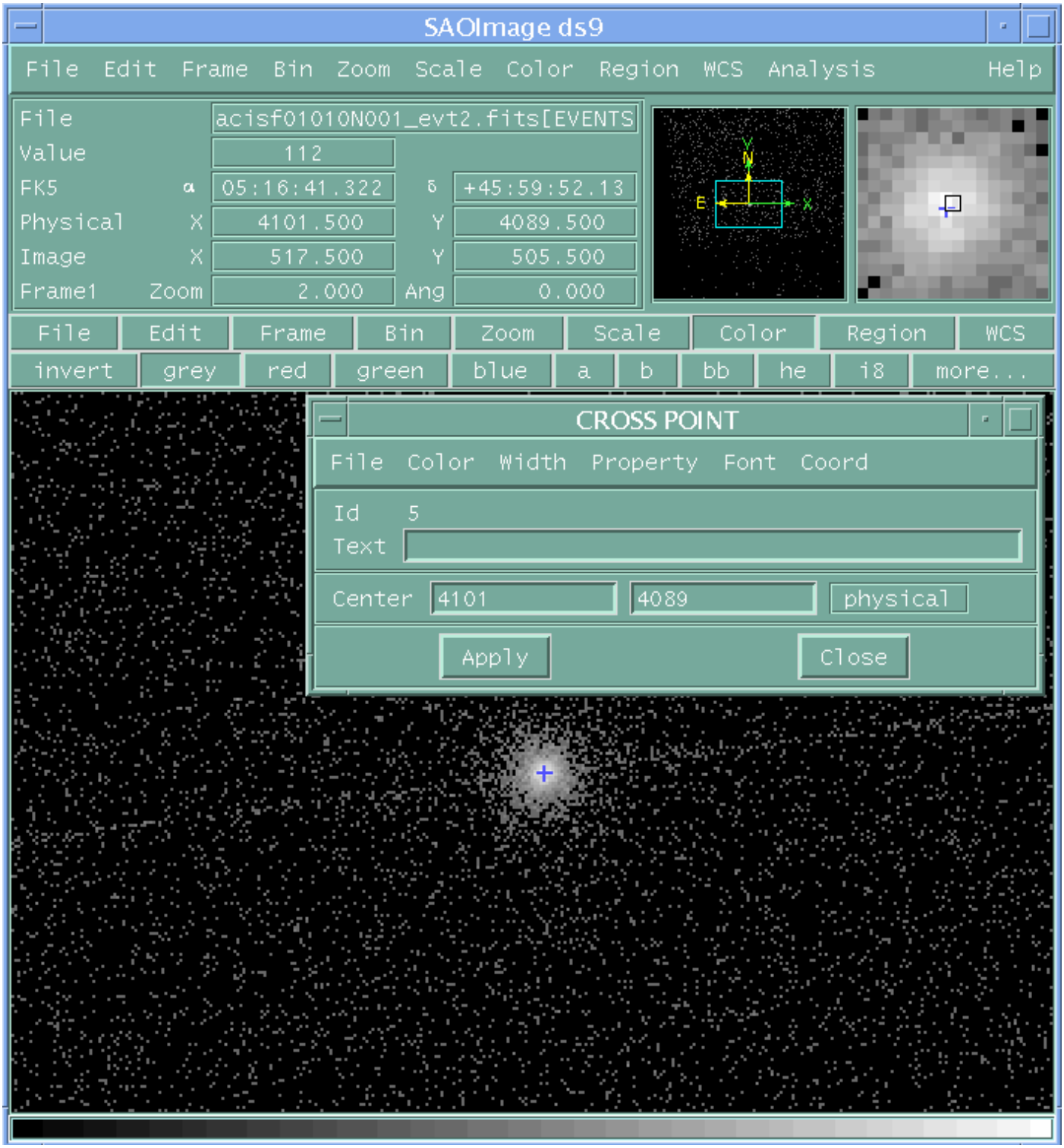


Image 3: Energy scale region file