OGIP Memo OGIP/94-010

ROSAT data analysis using xselect and ftools

Version: 1.1
(1996 April 2)

ROSAT Guest Observer Facility
Office of Guest Investigator Programs
Laboratory of High Energy Astrophysics
NASA Goddard Space Flight Center
Greenbelt
MD 20771
rosat@rosserv.gsfc.nasa.gov
edited by T.J.Turner

Acknowledgements

Contributors to the ROSAT software detailed here include Banashree Seifert, Rehana Yusaf, Ian George, Larry Brown, Bruce O'Neel, Jim Ingham, Alan Smale, Lorella Angelini, Bruce O'Neel, S.Snowden, K. Arnaud, T.J.Turner.

Contents

Chapter 1

Introduction

As the ROSAT data have been distributed as FITS files, one can easily and quickly analyze ROSAT data using the `xselect` analysis system developed for ASCA.

`xselect` is a command–line interface to `ftools`. Its major role is to allow easy manipulation of the data (storing intermediate data products internally during use). It allows quick and easy extraction of images, light curves and spectra from the events files, with easy application of filters, data screening and other selection criteria, and easy viewing of intermediate data products (images, light curves and spectra). At present, `xselect` runs under SunOS-4, DEC/Ultrix, DEC/OSF and VAX/VMS. `xselect` runs optimally with 24 MB of memory.

The `ftools` which you almost always need in your analysis are spawned from `xselect`. There are also a number of `ftools` which you may need, depending on your particular analysis requirements, these can be run at the system level.

This document outlines the analysis of ROSAT data using the `ftools` and `xselect` software. It assumes that the session is running on a terminal capable of supporting X-windows.

Please note that the `xselect` program is still evolving very fast so some things illustrated below will change in future releases. If you have suggestions for ROSAT–related improvements or spot bugs then please send them to turner@lheavx.gsfc.nasa.gov.

This document assumes that `ftools` and `xselect` are available standalone on your system. If they are not, you need to install them before you begin. To obtain the software, installation guide and full Users Manual for these tools please ftp to the anonymous account at legacy.gsfc.nasa.gov Software and documentation are available under /software/ftools/release/ and help is available by emailing ftoolshelp@athena.gsfc.nasa.gov

This document does not cover all of the sophisticated functionality provided by `ftools/xselect`, but covers some of the basics steps you are likely to need during your ROSAT data analysis. Please see the `xselect` Users Guide or online help for a detailed description of the more advanced features of the software.

1.1 ROSAT File formats

As ROSAT data processing software has been upgraded during the course of the mission, the file formats and the sequence-naming conventions have evolved. Data exist in three formats at the current time.

- Data processed in the US before mid-April 1994 were written in the so- called Rev0 FITS format. In the Rev0 format there are approximately 30 files for a typical PSPC dataset and 35 for a typical HRI dataset; the exact number of files depends on the details of the processing.
- Data processed after mid-April 1994 are written to tape as FITS files using the Rationalized Data Format (RDF), which uses a standard set of header keywords and data table structures to provide the data in a more user- friendly, multi-mission context. For data written in RDF there are 15 data files associated with each PSPC dataset, and 8 files associated with each HRI dataset.
- Data processed before mid-April 1994 in Germany are written in what we designate Rev0 German format. Data storage in the German files is significantly different to the oether two formats, and one must reformat the German events files before analysis is possible in the `xselect` or `ftools` environment. For data written in Rev0 German format there are 21 files associated with a PSPC dataset (the events file has a name like wp700272_events.tfits).

Because both MPE and the USRSDC have adopted RDF, the format of the ROSAT data products will be independent of where the data were processed; thus users of ROSAT data processed at MPE will easily be able to interpret and use ROSAT datasets processed in the US, and vice–versa. Eventually the ROSAT data archive will be overwritten such that all data are in the same Rationalized Data Format -RDF). In addition, RDF is being adopted by numerous other missions (EINSTEIN, ASCA and XTE) which means that software which can interpret and analyze ROSAT data should be able to interpret and analyze data from these and other missions.

The files distributed on the RDF data tape adhere to the FITS standard as described in the NASA/OSSA Office of Standards and Technology (NOST) draft 100–0.3b, *"Implementation of the Flexible Image Transport System (FITS),"* dated November 6, 1991. The ROSAT files use FITS primary images and FITS IMAGE extensions and ASCII and BINARY table extensions, although most data (apart from images) are stored in BINARY table (BINTABLE) extensions.

Detailed descriptions of each of the Rev0 and RDF files can be found in the Data Products Guide.

1.2 U.S. Sequence Naming Conventions

A key difference between early Rev0 format data and later Revo format data pertains to the way separate observation intervals (called "OBIs") are combined. Initially, all available OBIs were used to produce a single set of data products, later, OBIs which were part of the same sequence but which occurred in different "observing seasons" (i.e., 6 months apart) were processed and distributed separately.

All data processed by the U.S. RSDC using SASS versions 3_0 through 5_9 (Rev0) used file names of the form:

```
               r<detector-code><ROR><extensions>,

where:         r                - stands for ROSAT

               <detector-code>  - code for detector/filter:
                                        h - HRI
                                        p - PSPC
                                        f - Filtered PSPC.

               <ROR>           - 6-digit Rosat Observation Request number

              <extensions>     - additional characters specifying "type"
                                  of data.
```

In later Rev0 data, observations conducted over different "observing seasons" (i.e., over gaps of several months) were no longer combined to form a single data set. Additional observations, known as "add-ons", were processed separately. The sequence-naming convention became:

```
                r<detector-code><ROR><add-on number><extensions>
```

where the two additional digits added after the ROR number are:

```
                <add-on number> - 00 indicates the first segment,
                                - 01 indicates the second segment,
                                - 02 indicates the third segment,
                                    and so on.
```

An additional identifier has been inserted into the RDF file names to indicate both "split" Rev0 sequences and "mispointed" observations (those for which the telescope was not pointed at the intended target). The RDF FITS file names have the form:

```
        r<detector-code><ROR><obs-code><extensions>
```

where obs-code has replaced add-on number:

```
  <obs-code> =
```

- n00 - observed as scheduled, first segment ("n" = Normal)

- a01,a02,... - observed as scheduled: first, second, and subsequent segments added to the "normal" segment ("a" = Add-on)
- a00,a01,... - indicate 1st, 2nd, and subsequent segments for an observation which was first processed as a single observation in rev 0 processing.
- m01,m02,... - mispointed segment (targeting was not as scheduled), numbering begins at 1 ("m" = Mispointed)

1.3 Coverting Old style German events files to RDF

`xselect` can handle both the old style US FITS files (hereafter called US Rev0), and the new RDF FITS format files. In the Rev0 case the events for sequence "rp700105" for example are in the file rp700105.fits or rp70010500.fits, depending on what version of SASS was used for the data processing. In the RDF case the events are in a file with a name like rp700105n00_bas.fits or rh110267n00_bas.fits, the "basic" file.

Old format German events files must be converted to RDF format prior to starting `xselect`. To do this run

```
gevt2rdf
Enter file name: wp700559_events.tfits
Infile is wp700559_events.tfits
ONTIME :    2393.0000000000
Number of rows in STDGTI table:   2
Number of rows in STDEVT table:   28845
New file wp700559.fits written
```

Old German format archive images can also be converted to RDF FITS format, the converted image can then be displayed via saoimage

Example:

```
img2us
Name of German ROSAT image FITS file[ ] wp700559_image1.ifits
```

This creates a FITS image of name wp700559_im1.fits which can be displayed using saoimage at the system level i.e.

```
saoimage wp700559_im1.fits
```

Chapter 2

Reading the data into Xselect

```
xselect
```

starts up `xselect`. First thing you'll see is:

```
** xselect V1.3 **

> Enter session name >[xsel] session1
```

The `xselect` program was originally written for the analysis of ASCA data but use of FITS formats for the ROSAT data files has made it relatively easy to adapt the `xselect` software to handle ROSAT data.

Session names are used as the first letters of temporary files so that you can run multiple sessions of `xselect` from the same directory so long as you give each session a different name. You can save the `xselect` session (you will be asked if you wish to save the session upon exiting it) and later start it up again and continue from where you left off: in the above example, if you had saved an `xselect` session called 'session1' before, then it will ask you if you wanted to restore the saved session.

It should be stressed that images, light curves, spectra and events files extracted are temporary files and will be lost when you exit from `xselect` unless you have saved each (using "save spectrum", "save curve" etc) or unless you save the `xselect` session.

`xselect` initially comes up with the mission set to ASCA, this and other `xselect` defaults will change when you read in the events file, and `xselect` reads the header.

```
xselect> read events
```

```
> Enter the Event file dir >[] ./
> Enter Event file list >[] rp100000n00_bas.fits
```

(the data need not be in the current working directory and you can reset the default data directory using 'set datadir' /whatever/dir at any time)

`xselect` will display the new defaults and a summary of the datafile you have read in, the following shows an example of reading an RDF events file:

```
        xselect> read events



  /testdata/rp100000n00_bas.fits
Got new mission: ROSAT and new instrument: PSPCC
> Reset the mission and instrument? >[yes]
 This is ROSAT

 Notes:  xselect set up for      ROSAT
 Keywords for time and pha are  TIME       PI
 Units of time are              SECONDS
 Keywords for images are          X          Y


Setting: IMAGE binning    =   15
         WMAP binning     =   15
         Energy Column    =  PI
         Energy rebinning =    1
         WMAP keywords = DETX       DETY


Getting Min and Max for Energy Column...
Got min and max for PI:     1    500

Number of files read in:   1

******************** Observation Catalogue ********************

Data Directory is: /testdata/
HK Directory is: /testdata/


        INSTRUME OBJECT   DATE     RA_NOM   DEC_NOM  ROR_NUM  LIVETIME
      1 PSPCC    XRT/PSPC 11/04/94 .310590 .188E+04 110590 .188E+04
```

If you are reading in a US Rev0 format dataset then you will see a slightly different output, the differences are in the PI range and in the names of the keywords containing the information shown in the Observation Catalogue.

```
Getting Min and Max for Energy Column...
Got min and max for PI:     0    255

Number of files read in:   1

******************** Observation Catalogue ********************

Data Directory is: /testdata/
HK Directory is: /testdata/


        INSTRUME DATE-OBS XS-RAPT  XS-DECPT XS-OBSID XS-LIVTI
      1 PSPC     07/10/93 .208E+03 .693E+02 US701424 .962E+04
```

In some cases xselect cannot fit the information into the formated observation catalog display, and a field of wild cards (*) may result. If this occurs you can allow an unformated display (which may wrap around to 2 lines) by using

```
xselect> read events filename tdisp=no
```

Notes & Warnings:

- The RDF data have PI 1–500, and when you extract a spectrum you will have 500 channels which you can examine using 'plot spectrum'. There is not much data above channel ~ 300 and channels above 256 are not calibrated well enough for use in data analysis so the 'save spectrum' command will truncate your spectrum to 256 channels for `xspec`. Also note that the Rev0 data have PI 0–255 but this will **not** result in a 1 channel offset in your extracted spectrum (`xselect` understands what to do with these data).
- You can find out things about `xselect` by typing :

  ```
  xselect> help
  ```

  and find out the current state of play by :

  ```
  xselect> show status
  ```

  to reset things you can reset the instrument, e.g.

  ```
  xselect> set instrument PSPC
  ```

  or reset any parameter by itself, e.g.

  ```
  xselect> set binsize 200
  ```

  resets the time bin size to 200 seconds for subsequent extracted light curves.

- To examine the deafult settings of any parameters you can use 'lparm' followed by the name of a command which uses the parameter you are interested in, e.g.

```
xselect> lparm extract
```

```
NAME          TYPE    PROMPT                                   DEFAULT
======================================================================
continue      BOOLEAN > Continue? >                            no
bin_what      STRING  > Give parameter to be binned >
[binsize_t]   REAL    > Give bin size in seconds >             -1.0
[pharebin_t]  INTEGER > Temporary Bin size for PHA >           -1
[xybinsize_t] INTEGER > Rebinning factor for the image >       -1
[phalcut_t]   INTEGER > Temporary Lower cutoff for PHA >       -1
[phahcut_t]   INTEGER > Temporary Upper cutoff for PHA >       -1
[exposure]    DOUBLE   > Threshold exposure value for the lig 0.99
[use_qdp]     BOOLEAN > Use QDP light curves? >                no
```

  task defaults are then listed (parameter names ending in _t, such as binsize_t, are temporary parameters which can be used if you wish to set a parameter for one run only, and the normal default to be used thereafter. Many of the _t parameters will not be set if you don't normally use them).

- You can abbreviate commands so long as the string you type is unambiguous

Chapter 3

Image Analysis

3.1 Making an Image from the Events file

You may wish to start your analysis by looking at the ROSAT image, and while 'standard' images are provided with the archival dataset you will probably wish to make your own spatial binning and energy band selections.

```
xselect> extract image
```

```
xselect> saoimage
```

you will now have extracted a full-band image with the default spatial binning and displayed it in an saoimage window.

You can save this image by :

```
xselect> save image sourceimage.fits
```

3.1.1 Image subselections

Image can be extracted under pha, time, spatial or housekeeping filters, pha is loosely used to mean pha or pi and so

```
xselect> extract image phalcut=12 phahcut=200
```

extracts an image using PI channels 12 to 200 etc.

```
xselect> filter time file xsel_cursor_gti.xsl
```

```
xselect> extract image
```

extracts an image using only the good times previously saved in the file xsel_cursor_gti.xsl (see Chapter 4 for details of how to make such a time filter file).

As the raw sky image has 0.5 arcsec pixels, and the binning factor is 15, for example the default image is created as 1024 x 1024 for the PSPC.

To change the resolution of the image you must reset the spatial binning

`xselect> set XYBINSIZE 8` (4 arcsec bins)

Care is necessary when switching the xybinsize and using regions defined in saoimage. `xselect` will assume that the regions that it is using for any extraction were defined on an image binned up by the factor given by the current xybinsize so if you create a region then change the xybinsize you will not get correct results.

To extract a sub-image at high resolution one can reset xysize. Images are constrained to be square and so xysize requires the dimension, in output pixels, of the side of the output image. Each output pixel is of the size determined by xybinsize. For example, if you are working with the sky image (i.e. Coordinates for the image are X Y) xybinsize is set to a factor of 8, so each output pixel will be 8x0.5 = 4 arcsec. Setting xysize to 1000 will yield an image 1000x1000 4 arcsec pixels, or 66.7 arcmin on each side.

As the limit of xysize is 1200, you may wish to recenter your output image, in order to get high resolution information about sources at the edge of the field–of–view. To do this, set xycenter in **raw** pixels. For the *ROSAT* PSPC the center pixel in raw 0.5 arcsec sky pixels is 7680 7680.

Example:

```
xsel:ROSAT-PSPC > set xycenter 6530 6240

Remember to SET XYSIZE, or the extractor will ignore the centering

xsel:ROSAT-PSPC > set xysize 800
xsel:ROSAT-PSPC > extract image
 Extractor 1.0q
 XPI 2.1t
 Doing file: /testdirectory/rp700105.fits
 Getting FITS WCS Keywords

show st100% completed
    Total      Good    Bad: Region      Time      Phase       PHA
   124810    109163          14930         0          0       717
================================================================================
```

```
  Grand Total      Good   Bad:  Region      Time     Phase       PHA
       124810    109163           14930         0         0       717
 Total Counts for image :   109163
 Total Time for image :     9625.0000000000
 Total Counts/Time for image :     11.341610389610
```

To make an image in detector coordinates

```
xselect> set image detector
```

and then proceed as before. Both PSPC and HRI datasets are oversampled to 0.5 arcsec pixels in the sky image (raw images sizes are thus 15360 and 8192 respectively). In detector cordinates they are 0.934 and 0.600 arcsec per pixel, for the PSPC and HRI, respectively.

3.1.2 Region selection

Once the desired image is displayed in the saoimage window you can select a subregion of interest. Click on the "Cursor" button. There are two sets of cursor buttons available; toggle between them using the "Region" button at the righthand end. To define a region click on the appropriate geometric shape and then manipulate the region using the cursor. The left button changes the center of the region, the middle button changes the size of the region, and the right button changes the orientation. To save the region first type "s" to save as an included region then go to the region "write" button and at the prompt give the filename for the region specifier. **The current version of xselect only supports circles, boxes and triangles, please do not use more complex regions**. If you want to specify a region to be excluded then type "e" rather than "s". To tell `xselect` to use your region file :

```
xselect> filter region mysource.reg
```

You can add other region files simply by repeating this command, but remember to use the reset button to clear old saved cursors. To remove a region file that you don't want to use :

```
xselect> filter region "-mysource.reg"
```

and to clear all the regions in filter :

```
xselect> clear region all
```

Assuming that you have set the region filter to be mysource.reg you can now remake the image using photons only from the mysource.reg area :

```
xselect> extract image

xselect> saoimage
```

Region files are simple ascii format files and can be edited (with care). An example of a region file from saoimage is

```
# /test_data/xsel_image.xsl
# Tue Sep  30 17:18:03 1994
# shape x, y, [x dimension, y dimension], [angle]
 CIRCLE(502.50,569.00,6.80)
```

For detailed image analysis the extracted image, or the events file, must be read into `ximage`, which is described briefly in Chapter 6.

Chapter 4

Timing Analysis

4.1 Light Curve Extraction

As for images, light curves can be extracted for the whole field, or filtered through a region descriptor and /or additional housekeeping filters and phalcut and phahcut specifications. The time bin that will be used is "binsize", the current default can be seen in the "show status" summary and it can be reset using e.g. "set binsize 1000" for 1000 second bins. You can specify a minimum limit for partial bins, as a fraction of "binsize". For example

```
xselect> extract curve exposure=0.2
```

will include partial bins of 200 seconds exposure or greater, while binsize is set at 1000 seconds.

```
xselect> set device /xw
```
(/xterm will also work)

```
xselect> plot curve
```

will plot the temporary light curve file. The x-axis is seconds since the beginning of the observation and the start time in spacecraft clock seconds is written at the top of the plot.

You will be left with the "PLT>" prompt. At this point you can use standard PLT commands, (i.e. those used in the QDP plotting software– which some users may be familiar with). Useful commands include

```
PLT> r x 1.234e5 2.468e5

PLT> r y -0.1 5.
```

which reset the x and y axes to zoom into some interesting part of the lightcurve.

In some cases the user will wish to define a region of interest and extract the events from that region without binning the data. The extracted and saved mini-events file has the highest time resolution possible for those data, and can be read by XRONOS. This is the preferred method when doing complicated timing analysis since it avoids the danger of introducing spurious binning effects.

4.2 Simple Time filtering

To define time intervals for inclusion/exclusion having already extracted the light curve

```
xselect> filter time cursor
```

plots the light curve in cursor time selection mode; the user will see

```
Type:
  - click to include a time region
c - to cancel cursor selection
e - makes the selection excluded
l - puts cursor on the left
r - puts cursor on the right
x - to exit cursor selection
p - to return to PLT prompt
  - click outside the plot to cancel a selection

Enter QUIT at PLT prompt to continue
```

and after exiting from PLT> a file of the time selections will be already added to the filters (type 'show status' to see that this is so).

To enter times by hand use 'filter time' MJD,SCC,UT which allow you to enter timing filters by HAND in MJD, Spacecraft Clock time, or UT. To use times from a (previously saved) file 'filter time' FILE. The associated 'save' commands are now grouped into 'save time cursor', and 'save time hand'. The associated clear commands are grouped under 'clear time'. Once you have saved (say) cursor times to a file, you can use those time selections again in the current or in future sessions by the command 'filter time file' thus

```
xselect> filter time file gti1.xsl
```

will apply the times saved in the file gti1.xsl, while

```
xselect> clear time file gti1.xsl
```

will clear that filter.

You can check your work by creating a lightcurve using just the selected time intervals :

```
xselect> extract curve

xselect> plot curve
```

4.3 Barycentric correction

For detailed timing analysis you will wish to apply a barycentric correction to events of interest.

The correction programs work directly on FITS files and are called are `bct` (Barycenter Correction Table: corrects the times in the orbit table at the barycenter) and `abc` (Apply Baycentric Correction: applies the correction to the arrival time of the photons in the events list at the barycenter). Invoking a script, `rosbary`, the two tasks can be run in sequence.

The most likely application is that users will extract a mini-events file for the photons of interest and then correct this before entering XRONOS.

bct

This task computes the barycentric correction on the times contained in the orbit file. The times in the orbit file are sampled every 60 seconds. This task takes as input:

- the orbit file, corresponding to the observation for which the correction is needed. Both format old style corresponding to REV0 and the RDF style file are supported. The orbit data can be found in a file containing the extension SPORB, for US Rev0 this is the file with extension .so, i.e. file number 14); in the RDF case the data are in the _anc.fits file i.e. that containing the extension EPHEM (_anc.fits; in the old style German case the file required is _orbit.tfits.
- the equatorial coordinates of the source given either as hh:mm:ss.s, dd:mm:ss.s or in degrees (epoch 2000.0).
- output filename, this will contain the uncorrected and the corrected orbit time. The output FITS file consists of 4 columns, containing the integer and the fractional part of the uncorrected and corrected times. The times are written as JD. The output file is used by `abc`.

This program makes use of the JPL DE200 ephemeris given as a FITS binary table (Standish,M.,1982 Astr. Ap.,114,297).

abc

This task corrects the arrival times of the photons, in the EVENT extension and the times in the GTI extension, for the SSC-UTC relation and uses the correction table created from `bct` to correct the arrival time at the barycenter. The Barycentric correction is computed by a linear interpolation between the corrected orbit times (output from `bct`). If a photon falls outside of the orbit interval a warning is issued together with the photon time. This task takes as input:

- the event list data file (in FITS), corresponding to the observation for which the correction is needed. It is recommended to input the event data file obtained after having applied a region selection, which contains only the photons from the source for which the correction is required. Both format old style produced by REV0 processing and the RDF are supported. Old style German evenets files must be converted to RDF first, using **`gevt2rdf`**.
- input correction table: table containing the corrected and uncorrected orbit times (in FITS) created by `bct`. It consists of 4 columns, containing the integer and the fractional part of the uncorrected and corrected times, written as JD.
- the equatorial coordinates of the source (epoch 2000.0) given either as hh:mm:ss.s, dd:mm:ss.s or in degrees. The coordinates are not used for computation but they are requested in order to test the compatability with the input correction table.
- output file: event output file (in FITS) with the arrival times of the photon corrected at the barycenter.

  The correction of the times to UTC is computed with the following formula :

  $$UTC = JDref + (Aj \times (SCC - SCCo)^{j-1})/86400.0$$

  where the Aj coefficients are stored together with the time interval validity in the ssc_to_utc2.fits file (available from the ROSAT caldb area) and spacecraft clock (SCC) photon arrival times. Each row in the ssc_to_utc2.fits file contains the JDref, SSCo and Aj values together with the start and end time for which the listed value are applicable. The start and end time for the interval are not corrected to UTC.

`bct` needs to be run before `abc`.

<u>rosbary</u>

This script runs `bct` and `abc` in sequence The input paramters are:

orbit filename (FITS ext SPORB or EPHEM) (input file)

output correction table filename

RA hh mm ss.s or degrees in 2000 equinox

DEC dd mm ss.s or degrees in 2000 equinox

event filename (FITS EXTNAME EVENTS or STDEVT) (input file)

output corrected event filename

`rosbary` (and `bct` and `abc`) will work on the Rev0 US files, the RDF format files and will handle the German orbit files. However, users need to convert the German events files to RDF format before running these tasks.

For detailed timing analysis the time series or events file can be read in `xronos`, which is brielfy described in Chapter 6.

Chapter 5

Spectral Analysis

5.1 Extracting a Spectrum

When all desired filters are in place, then you are ready to extract a spectrum. This spectrum should be created from the PI column in the event file (the keyword for pha should be set to PI, if this is not the case, 'set phaname PI') the option for the data to be regrouped is offered when the spectrum is saved (`grppha` will allow you to reset the grouping at a later stage, so answering yes to the grouping is not going to constrain your spectral analysis).

```
xselect> extract spectrum

xselect> save spectrum source.sp
```

When using a RDF PSPC dataset you will see that the temporary spectral file made by `xselect` is 500 channels, and can be examined using 'plot spectrum'. As noted in section 3.1, when you save an RDF spectrum it is truncated to 256 channels which can be fit in `xspec`.

When extracting an HRI spectrum, you will extract the full 16 channels spectrum, and you will be offered a default rebinning down to a single channel, on saving this spectrum. It is recommended that this rebinning be used, if the user wishes to fit the HRI data using the standard single channel HRI matrix (seee section 6.1 for details on response matrices available from legacy).

The saved spectrum can be read into `xspec` directly. You will see that the spectral file contains a small map image in the primary array, showing where on the detector the events in the PHA or PI file originated. This weighted map is known as the WMAP, and is used by `pcarf` for generating an accurate ancilliary file which is used in conjunction with the photon redistribution matrix in the spectral fitting process.

5.1.1 Spectral filters

A variety of filtering commands are available as qualifiers for a spectral extraction. These are

```
xselect> filter time
```

For example using

```
xselect> filter time file good_times.xsl
```

allows a spectrum to be extracted using the 'good time intervals' previously defined and saved to a file. The command

```
xselect> filter region
```

is almost always required in analysis of imaging data, to define the region for which your spectrum is being extracted. Region files can be defined using an saoimage display of the field of interest, as detailed in Chapter 3. For example

```
xselect> filter region binary_bgd.reg
```

will use the region file binary_bgd.reg as a spatial filter for subsequent extracted spectra.

```
xselect> filter phase
```

allows a phase to be defined, and thus good time intervals are calculated and used as a time filter for the extracted spectrum. Similarly,

```
xselect> filter intensity
```

allows the user to extract a spectrum over a user-defined count rate

Example:

```
xsel:ROSAT-PSPC > filter intensity
> Enter ranges for intensity filtering (i.e. .01-.02,1-5) >[] 10-15
Making GTI to implement intensity selection with Boolean expression:
 * RATE>=  0.1000E+02&&RATE<=  0.1500E+02
```

These filters can be cleared in the usual way, e.g.

```
xselect> clear time good_times.xsl
```

Extracted spectra can be model–fit in `xspec`, or manipulated (e.g. to make ratios of high versus low state spectra) using `mathpha` (Chapter 7).

Chapter 6

Preparing for XSPEC, XRONOS and XIMAGE

Together, `xspec`, `xronos` and `ximage` are known as the `xanadu` software. These programs allow the detailed spectral, timing and spatial analysis of your data (in FITS format). A very brief introduction follows.

The xanadu software and installation guide can be found on the anonymous ftp area of legacy.gsfc.nasa.gov under the `/software/xanadu/` directory. For help finding or installing `xspec, xronos` or `ximage` software or manuals please email xanprob@athena.gsfc.nasa.gov.

6.1 Prior to XSPEC

There are just a few considerations prior to reading your FITS spectrum into `xspec`. If your source was observed on-axis, you may wish to use the standard on-axis response matrices which are already convolved with the on-axis mirror effective area and the PSPC gas transmission and window functions.

The ROSAT PSPC response matrices

The anonymous ftp /caldb/data/rosat/pspc/cpf/matrices/ directory tree on the legacy.gsfc.nasa.gov machine contains various response matrices (and related products) for the ROSAT PSPCs.

<u>256 PI channel detector redistribution matrices</u>

The following matrices are 2-dimensional arrays (energy vs channel) containing the probabilities that a photon of a given incident energy which enters the detector will give rise to an event in a given PI channel. As such they are VALID FOR ALL REGIONS of the detector, but also require an appropriately constructed "Ancillary Response File" (ARF) to enable spectral analysis to be performed in `xspec`. (ARFs can be constructed using `pcarf`, as detailed above)

```
pspcb_gain1_256.rmf
        - valid for PSPCB data taken BEFORE 1991 Oct 14
pspcb_gain2_256.rmf
        - valid for PSPCB data taken AFTER 1991 Oct 14
pspcc_gain1_256.rmf
        - valid for ALL PSPCC data (detector destroyed on 1991 Jan 25)
```

On-axis 256 PI channel response matrices (including the on-axis effective areas)

The following matrices are 2-dimensional arrays (energy vs channel) containing the above detector redistribution (probability) matrices BUT MULTIPLIED by the XRT effective area, window transmission etc appropriate for ON-AXIS sources.

```
pspcb_gain1_256.rsp
        - valid for PSPCB data taken BEFORE 1991 Oct 14
          (formerly known as pspcb_92mar11.rmf)
pspcb_gain2_256.rsp
        - valid for PSPCB data taken AFTER 1991 Oct 14
          (formerly known as pspcb_93jan12.rmf)
pspcc_gain1_256.rsp
        - valid for ALL PSPCC data (detector destroyed on 1991 Jan 25)
          (formerly known as pspcc_92mar11.rmf)
```

Compressed, 34 channel matrices are also available in the same area, for use with the archival spectra stored in the RDF `_src.fits` files.

The ROSAT HRI response matrix

The HRI matrix can be found under /caldb/data/rosat/hri/cpf/hri_90dec01.rmf

6.1.1 Off-Axis Spectra

If your source was observed off-axis then you must make an ancilliary response file (ARF) for your spectrum and you should use `grppha` to group and set bad channel flags for the source spectral data (if you did not allow to the deafult grouping to be set when you saved the spectrum from `xselect`).

pcarf

This program generates a Ancillary Response File (ARF) for the ROSAT PSPC instrument. The ARF consists of a simple 1-d list of the corrections required to be applied to the input detector response matrix during the spectral analysis of input PHA dataset. The ARF contains the weighted effective area, PSPC gas transmission, window and filter (where applicable) information, and is used in `xspec` in conjunction with the photon redistribution *only* matrix (in the past we have used matrices which held the convolved area and photon redistribution information).

```
pcarf
** PCARF 2.0.0
Name of input PHA file[] target.sp
Name of input RMF file[/caldb/data/rosat/pspc/cpf/matrices/pspcb_gain2_256.rmf]
Name of output ARF file[target.arf]
Name of input SPECRESP or EFFAREA file[/caldb/data/rosat/pspc/cpf/pspcb_v2.
spec_resp]

** PCARF 2.0.0   Finished
```

Notes & Warnings:

- Spectra extracted from Rev0 PSPC data temporarily have the keyword values for some quantities related to correct use of the WMAP. Please run `fix_rev0_pha` on Rev0 spectra before you run `pcarf`, this will fix the problem.
- pcarf takes the effective area, PSPC gas transmission, PSPC window and filter files (filter where applicable) and makes an energy dependant correction file for use with the photon redistribution response matrix, during spectral fitting.
- pcarf does not include a correction for occultation by the PSPC ring or ribs, nor does it contain a point–spread–function correction for the extraction cell size
- Some old (Rev0) datasets have the instrument keyword PSPC, whilst the matrices have PSPCC or PSPCB, this may cause some warning messages.
- US Rev0 format data (but not RDF data) does not properly specify the size of the pixels in the Detector image. Because of this, the extractor (the program that `xselect` uses to make a spectrum) puts the wrong pixel size into the PHA file. The extractor will soon be modified to cope with this deficiency in US Rev0 data but it isn't fixed yet. You can run the `fix_rev0_pha` script to fix the extracted spectrum, if you intend to use `pcarf`.
- The HRI equivalent of PCARF is not yet available.

6.1.2 Grouping or Rebinning the Spectral Data

grppha

The raw 256 channels oversamples the PSPC spectral resolution many times so you may want to group up the FITS spectrum before analyzing it within `xspec`. This can be done using the `grppha` ftool. `grppha` sets a data grouping rather than actually binning the data. This means that the grouping can be changed as many times as required, while the original counts and error information remain available. In fact the data must not be rebinned using `rbnpha` or the resultant spectra will fail to match the dimension of the response matrices you will be using (`xspec` internally handles the binning of the response matrix to match the data grouping produced using `grppha`). `xselect` spawns the `grppha` ftool as you "save spectrum", it offers a reasonable setting for the bad channels, and a reasonable grouping, but you may wish to set your own grouping (**note** even if you allow xselect to set an initial grouping, you can reset this later using the 'reset group' command in `grppha`, so its a good idea to use the xselect default for the first quick–look)

```
grppha target.sp target_group.sp
```

`grppha` will first give you a summary of the keyword values set in your file. You may wish to change these in order to set defaults for the response matrix, ARF , background spectrum etc.

```
GRPPHA (): chkey ancrfile new_target.arf

GRPPHA (): chkey backfile target_bgd.sp
```

set default arf and background files for the spectral file.

Note Do not group up the background spectrum, `xspec` will handle the area rescaling and data grouping using the header information in both spectral files. `xspec` also automatically bins up the response matrix to have the same number of channels as the spectrum.

```
GRPPHA (): bad 1-8, 200-256

GRPPHA (): group 9-199 10

GRPPHA (): exit
```

The target_group.sp file will now give 19 channels when read into `xspec`.

If ARF, RESP and BACKFILE are not set in the source spectrum file header, then one can read them in at the `xspec` prompt.

rbnpha

In the case of the HRI spectra, the energy resolution is negligable and the response matrix for the HRI is a single channel matrix, while the data have 16 raw channels. In this case one should use `rbnpha` to bin up the 16 raw channels into the 1 resultant channel for which the HRI response matrix is calibrated. `xselect` offers this as a default upon saving HRI spectra, but you can run the task standalone

```
RBNPHA ():
```

```
Please enter PHA filename[agn.pha]
Please enter output filename[] agn_bin.pha
Please enter Resultant number of Channels[10] 1
Please enter Compression mode[linear]
 Main RBNPHA Ver 1.0.9
 RBNPHA ver 1.0.9 completed
```

6.2 Reading the Data into XSPEC

Having prepared the spectrum, and any arf file if required, invoke `xspec` by typing its name

```
>xspec

               XSPEC 9.00 16:25:16 14-Mar-96

  Plot device not set, use "cpd" to set it

  Type "help" or "?" for further information
 XSPEC> data agn.pha
  Net count rate (cts/cm^2/s) for file   1  0.2669    +/-  7.7344E-03
 XSPEC> ignore bad
 XSPEC> cpd /xw
```

the data are read in and the plot device is set. If you put the information as to where to find the response matrix, the background spectrum and (if in use) the arf file in the primary spectrum file header, then you are now ready to define and fit a model spectrum to the data. If you did not define these things, then you can do so within `xspec`

```
 XSPEC> resp /cpf/matrices/pspcb_gain2_256.rmf
 XSPEC> arf agn.arf
 XSPEC> back background.pha
  Net count rate (cts/cm^2/s) for file   1  0.2441    +/-  8.2032E-03
( 92.6% total)
```

You are now ready for model definition and spectral fitting. For more details about `xspec` see the xspec users guide, available from legacy under /software/xanadu/xandis/manuals/xspec_7/

6.3 Reading the events files or time series into XRONOS

xronos can read the FITS binned light curves and the photon time files written by xselect. xronos can be used for detailed analysis of time series and photons events files, it contains routines to calulate power spectra, hardness ratios, cross correlation functions etc. The xronos tasks lc1, lc2 and lc3 are used to plot light curves in 1 to 3 bands (respectively). The user can set the number of output bins, and the data window. The tasks efs and psd allow data folding and fft calculation, while acf, acs, ccf and ccs allow auto-correlation and cross- correlation analysis and sta and tss allow a statistical analysis of the dataset.

A detailed description of xronos is beyond the scope of this guide, but further information can be found in the xronos users guide available from legacy under /software/xanadu/xandis/timing/xronos/help/

In this simple example, a time series is read into the lc1 program, which allows binning and plotting of a light curve (which can be made up of several files) as follows:

```
% xronos

--------------------------------------------------------------------------
Welcome to Xronos
--------------------------------------------------------------------------

                       Type "help" for information


[1]xronos> lc1
> Program lc1         18-OCT-1994 11:48:13         Xronos vers. 4.02 Jul 94

Enter up to  50 input filenames and options for series 1 (or rtn)
lc1> Ser. 1 filename   1 +options[.rbf]=> test.curve

Selected FITS extensions: 1 - RATE TABLE;

Source ....... NGC 0000  Start Time (d) ....   601 16:22:03.000
FITS Extension  1 - `RATE` Stop Time (d) ...   605 02:38:43.000
No. of Rows ...         18       Bin Time (s) ......    200.0
Right Ascension                  Internal time sys.. Literal
Declination .....                Experiment ........ ROSAT    PSPC
Filter ........ none
Corrections applied: Vignetting-No ; Deadtime-No ; Bkgd-No ; Clock-No

Selected Columns: 1-Time;   2-Y-axis;  3-Y-error;

File contains binned data.

 lc1> Ser. 1 filename   2 +options[.rbf]=>
```

```
 lc1> Window Options ?          {y/n}[n]=>

   0 Time,    0 Phase,    0 Intensity,    2 Exposure Windows

 Expected Start ...  601.68197916667  (days)  16:22: 3:0  (h:m:s:ms)
 Expected Stop ....  605.11021990741  (days)   2:38:43:0  (h:m:s:ms)
 Minimum Newbin Time    200.00000      (s)
 Maximum Newbin No..             1482
 Default Newbin Time  579.194 s (to have 1 Intv. of512 Newbins)

 lc1> Enter Newbin Time (s) or neg.rebinning
      of Min. Newbin Time [   579.2    ]=>

 Newbin Time ......     579.194     (s)
 Maximum Newbin No.     512    (or 1 Intvs. of 512 Newbins)

 lc1> No. of Newbins/Intv.  (max: 65536)
      {value or neg.pow.of 2}[      512]=>

 Maximum of  1 Intvs. with  512 Newbins of 579.194 s

 lc1> Order of polynomial trend removal
            {0=none,1=1st,....,4=4th}[0]=>
 lc1> Time axis units
        {0=s from start,1=s,2=h,3=d} [3]=>
 lc1> Enter output filename
      [xronos.qlc1                     ]=>
 lc1> Plot options ?           {n/y}[n]=>

Intv 1   Start  601 16:25:12
     Ser.1 Avg  13.01  Chisq  3608. Var  12.83  Newbs. 9
     Min  8.973    Max  20.08  expVar 0.3804E-01  Bins 18
             Light curve ready !
 Frme    1  Written to output file : xronos.qlc1

 To plot  vs. Time (days), please enter
 PGPLOT file/type: /xterm
```

Thus a light curve was rebinned to 512 output bins, plotted in an xterm window and saved to xronos.qlc1, ready for further analysis.

6.4 Reading the events files or images in XIMAGE

The FITS image output from XSELECT can be read straight into XIMAGE using the command :

```
XIMAGE> read/fits goodimage.fits
```

The events file requires some rebinning in order to read in the entire field-of-view

```
XIMAGE> read/fits/rebin=30 events.fits

XIMAGE> cpd /xw

XIMAGE> disp
```

sets the plot device to xwindows and displays the data.

If you don't have an X device then you could use the contour option:

```
XIMAGE> smooth[/sigma=3]
```

smooths the image with a Gaussian of sigma 3 bins. Then make a contour plot :

```
XIMAGE> contour[/first=2/no_of=10]
```

The optional qualifers make 10 linearly-spaced contours but this does not plot the lowest contour.

```
XIMAGE> contour/overlay
```

will overlay the contour on a previously displayed image.

There are many other useful things that ximage can do, e.g.

```
         slice/x - make a slice in the x direction (use the left,
                   middle and right mouse buttons to select, erase
                   and choose a y region to do the slice accumulation)

background/box=n - measure the background in a box of size n pixels

          detect - source detection (use /back=n, to vary the background
                   box size)
             cct - change the color table

          circle - defines a circular region (can write to file)

            grid - draw a coordinate grid

ra_dec_to_pixel  - mark the image and return pixel for given RA and DEC

         psf/cur - measure the point spread function
```

```
  saoimage - spawn saoimage to display the image

write/fits - write the current image as a FITS file
```

Notes & Warnings:

- Region descriptors written by `ximage` are the same format used by `saoimage` and thus `xselect`, but in the typical case, `ximage` writes the region descriptor file in raw coordinates, and so "xybinsize" should be set to 1 in `xselect` just prior to using any ximage region file.

For further information regarding XIMAGE please see the XIMAGE users guide, available from legacy.gsfc.nasa.gov anonymous ftp account under /software/xanadu/xandis/image/ximage/doc/

Chapter 7

More Advanced Analysis

7.1 Screening/Rescreening the Data

`xselect` includes a facility to select times based on the values of some auxiliary quantities ie those not associated with each event. Examples could be angle between the pointing direction and the Earth's limb, some housekeeping parameter, or the total counting rate. This information must reside in a FITS file whose first extension is a BINTABLE with time column called TIME. For ROSAT, the housekeeping information is stored in the seq_anc.fits (the ancilliary file), the seq_anc.fits file does not, however, store all the information in the most useful way for `xselect` screening and users need to run PCFILT for PSPC data, or HRIFILT for HRI data.

Using PCFILT as an example (HRIFILT is very similar):

The FTOOL PCFILT can be run prior to starting an `xselect` session (or spawned from within `xselect`). PCFILT will resample the housekeeping data and calculate many useful parameters from the raw information contained in seq_anc.fits, making the ROSAT equivalent of the ASCA makefilter file.

The output (makefilter) file should have a .rmkf extension for `xselect` to pick it up automatically when it is required.

Example:

```
pcfilt
Please enter Ancillary filename[] rp700111_anc.fits
Please enter output filename[] rp700111.rmkf
Main PCFILT Ver 1.0.0
PCFILT ver 1.0.0 completed
```

The data is actually filtered within `xselect` as follows

```
xselect> select mkf
```

> Boolean expression for filter file selection >[]

at this prompt give a selection expression. This expression can be in either Fortran or C syntax. For instance, if you wanted all times when OBI_NUM exceeded 1 then the expression would be "OBI_NUM .gt. 1". `xselect` will then calculate and store all time ranges for which this is true. These time ranges will be used for any extractions performed by `xselect`.

After running "select mkf", a "show status" command will show the filter; any data product extracted while this is in place will be filtered accordingly.

The makefilter file can be used to screen down the "standard events", ie those already screened by SASS, or to rescreen the total set of events. In the latter case, you first need to make a file of the total events, by summing the SASS selected events, and the rejected events, and generating a set of good time intervals which are applicable to this total dataset.

If you wish to bypass the SASS data screening, and do your own screening from the master set of events, you can do this for an RDF dataset by combining the rejected and good events, and calculating the full set of time intervals associated with these data

```
mk_allevt
mk_allevt 0.2.0:  Type 'mk_allevt -h' for instructions.
***
File: "" is illegal for output
You have the following likely looking file groups:
1       rh110267n00
2       rh701446n00
3       rh701447n00
4       rp110590n00
5       rp110231n00
Type corresponding number to select a set of files.
Or type a specific file or pathname.
Output (abas) Filename: 5
```

will combine the RDF rejected and good events into a single events extension in a new file (by default this will be named rp110231n00_abas.fits in this example). This new file can be analyzed in `xselect` as normal.

7.1.1 Saving screened events

If you have screened your events file, then you may wish to save the screened events for later analysis, first you must extract the events under all desired filters

```
xselect> extract events
```

```
xselect> save events good.events
```

After the second command you will be asked whether you want to read this data back in. If you answer yes then `xselect` uses the file which you just created rather than the original data files.

7.2 Exposure Correcting an image - pcexpmap and hriexpmap

Creates an exposure map for user specified time period and energy range for any US Rev0 or RDF format ROSAT datasets

EXAMPLE - To exposure correct a PSPC image:

- Clean the events file to exclude events which follow within 0.35 ms of a "precursor" event. This excludes some of the low pulse-height "after-Pulse" signal which affects data collected after 1992 May. Such a cleaning can be achived using the ftool `burst`.
- Create the image in the desired energy range and at the same spatial resolution as the exposure map (the sky data will require a factor of 30 rebinning to match the template maps currently available - there will be an unavoidable slight mismatch due to the 14.94733 arcsec/pixel size used in the detector maps, but this should not be a significant problem in the data analysis). If you wish to make a selection based on time, create a new events file which contains your good events, then `pcexpmap` will pick up the correct GTIs from this file for use in creating the exposure map.
- Create the exposure map in the appropriate energy range
- Use `farith` to perform the division of image by exposure map
- Display the corrected FITS image using saoimage

pcexpmap

To use `pcexpmap` the detector maps created from the ROSAT All-Sky Survey data are required. These may be obtained from the /caldb/data/rosat/pspc/cpf/detmaps/ directory on legacy.gsfc.nasa.gov. The detector maps are available in a number of energy bands, and the filenames contain the PI channel ranges used, where channel 11 is $\sim$ 0.11 keV, channel 200 is $\sim$ 2 keV etc. The detector maps currently available have the following rootnames (one map each for PSPCB and PSPCC):

```
det_11_19
det_20_41
det_42_131
det_132_201
det_42_201
```

```
det_42_51
det_52_69
det_52_90
det_70_90
det_8_19
det_8_41
det_91_131
det_91_201
```

The output from `pcexpmap` is a 512x512 FITS image of the whole PSPC field of view (with pixels 14.94733 arcsec per side; representing a blocking factor of 16 over the raw [0.9341875 arcsec] pixelsize) of the effective exposure time (in seconds) at that position. The effects of vignetting (for a spectrum equal to the mean spectrum of the X-ray background in the PSPC band) and spatial variations in the efficiency of the detector are included (via the detector maps), along with detector deadtime effects (applied by `pcexpmap`).

The program follows the suggestions of Snowden et al. (1992, ApJ, 393 819) and Plucinsky et al. (1993, ApJ, 418, 519) to exclude regions of the PSPC near the edges of the PSPC which are strongly affected by the particle background, the "bright line" regions. These regions are set to have zero exposure time.

In brief, the attitude and GTI files are used to construct a matrix of the time the instrument spent at each pointing position (X,Y relative to the nominal pointing position defined by the optical axis) and roll angle. (The X,Y are in units of 14.94733 arcsec for historical reasons, the ROLL steps are in units of 0.2076 degrees.) The event-rates file is then used to calculate the live-time fraction at each of these positions. Finally the output exposure map is constructed by moving the detector map to each off-axis position, rotated to each roll angle, and adding the detector map with the appropriate weighting factor to the exposure map under construction.

In the following example, the user removes afterpulse events, creates and image and exposure map, and corrects the image using `farith`.

Example:

```
burst
Name of FITS file and [ext][] rp100000n00_bas.fits[2]
Name of output FITS file[burst.out]
Maximum time between events in seconds[0.003]
Whether to keep first event in burst[yes]
```

Within xselect, extract an image

```
xselect> set xybinsize 30
```

to match the resolution of the detector maps

```
xselect> extract image phalcut=52 phahcut=90
```

extract our image in the $\sim 0.52 - -0.9$ keV band

```
xselect> save image image_52_90.fits
```

```
pcexpmap
Enter Event rate filename[]rp100000n00_anc.fits
Enter Attitude filename[] rp100000n00_anc.fits
Enter GTI filename[] rp100000n00_bas.fits
Enter Detector map file[] det_52_90_b.fits
Enter output filename[] expmap_52_90.fits
Enter Instrument/Detector name[PSPCB]
 MAIN : PCEXPMAP Ver 2.0.1
 ... Number of unique detector positions         378
 ... Number of entries when Detector ON         9634
 ... Number of entries when Detector OFF           0 (A1LL c/rate<10)
 ... Total ONTIME        9634.00000 s
 ... Total LIVETIME      9314.57227 s
 ... Average MV c/rate    111.51972 count/s
 PCEXPMAP Ver 2.0.1 completed
```

```
farith
Name of Ist FITS file and [ext#][] image_52_90.fits
Name of 2nd FITS file and [ext#][]expmap_52_90.fits
Name of OUTFIL FITS file[] corrected_52_90.fits
Name of operation[Enter Operation ADD,SUB,DIV,MUL(or +,-,/,*)]/
```

Please see the online help files for pcexpmap and hriexpmap for more details.

Notes & Warnings:

- It should be noted that the o/p data array is written as REALS in the Primary extension of the o/p FITS file. The image display and manipulation task, saoimage (v1.06), is unable to correctly read such datasets on DEC VMS and ultrix platforms.
- The current version of this task will only operate on i/p datasets in US Rev0 format or RDF format (ie cannot read German/UK Rev0 datasets). Furthermore all i/p datasets must be in the same format.
- To use the exposure map, extract an image at the same pixel size and use FARITH to flatten the field by dividing the image by the exposure map output from `pcexpmap`. The exposure corrected image can then be displayed using saoimage.

- `pcexpmap` will soon be enhanced to generate the image for you in the same energy band and at the same spatial resolution as the template detector map.
- US Rev0 events files must be time sorted (using FSORT) before `burst` can be run.
- Event rate filename should be seq.evr (Rev0) or seq_anc.fits (RDF) Attitude filename should be seq.cas (Rev0) or seq_anc.fits (RDF) GTI filename is the events file you used to make the sky image.
- `hriexpmap` will be available soon

7.3 Comparing the radial profile data to the psf

First the radil profile should be extracted from the events file using extrpsf

7.3.1 extrpsf

This caltools package task extracts a Radial Point-Spread Function (RPSF) dataset from an event file. An RPSF dataset consists of the number of counts per unit area as a function of radius (using series of concentric annulus). This is calculated using the X & Y (ie RA & dec) columns in the event file centered on a point defined by the user (in either RA & dec or pixel coordinates) for a user-defined number of annuli and outer-radius. If desired, the user can also specify the radius (within the outer radius) beyond which will be used to estimate the number of counts per unit area corresponding to the background. The user can also specify any of several prescriptions to be used to calculate the statistical errors associated with the RPSF data. The output psf file is FITS format, and can be directly compared to the theoretical psf model for the hri or pspc data as described below.

Notes & Warnings:

- The current version of this task cannot handle region masks.

Then the data should be compared to the model, using the script calcrpsf

7.3.2 calcrpsf

This routine is a multi-task wrapper for Radial Point Spread Function (RPSF) which allows several common steps to be performed by a single script. This task interfaces with radial profiles produced using either extrpsf or `pros` (after conversion of the pros files to FITS format via STWFITS and ST2RPSF).

calcrpsf spawns:

- `st2rpsf` - Reads stsdas.fitsio.stwfits radial profile and converts to OGIP fits format. This step is ONLY required if you are comparing PROS data to the model. If using extrpsf to extract the profile, please answer 'no' when asked if you want to run st2rpsf
- `rbnrpsf` - Rebins a radial PSF dataset (for example, to get 20 counts per radial bin)
- `pcrpsf` - Calculates a theoretical radial PSF for the ROSAT PSPC

  OR
- `hrirpsf` - Calculates a theoretical radial PSF for the ROSAT HRI
- `rpsfqdp` - Converts OGIP format radial profile file to ASCII QDP

The user is prompted as to which of these tasks are desired, the ftool `rpsf2eef` is also available, which converts a radial PSF (RPSF) dataset to a radial encircled energy function (REEF) dataset. The REEF file can be converted to qdp using `rpsfqdp`, in the same way the RPSF file can.

```
        calcrpsf
        ** CALCRPSF 1.3.0
        Input filename[temp] radial_profile.fits
        Output filename[calcrpsf.out]
        Run st2rspf ?[no]
        Run rbnrpsf ?[yes]
        Generate a theoretical RPSF dataset ?[no] yes
        Run rpsfqdp ?[no] yes

        Please enter Telescope name[ROSAT]
        Please enter Instrument name[PSPCB]
        Please enter Minimum PI channel for image[] 12
        Please enter Maximum PI channel for image[] 200
        Please enter background count rate (ct/pixel)[] 5e-4
Please enter PHA filename :[] NONE
Please enter Off Axis histogram filename :[%] NONE
Please enter RMF filename :[/caldb/data/rosat/pspc/cpf/matrices/pspcb_gain2_256.
rmf]
Please enter off_axis angle (arcmin) :[0.5]
 ** pcrpsf 3.0.1
 ** pcrpsf 3.0.1 completed successfully


 *** spawning RPSFQDP to convert RPSF dataset to QDP:
rpsfqdp chatter=5 datafile="rpsfpred.tmp" outfile=xxx.calcrpsf
 ** rpsfqdp 3.0.1
 ** rpsfqdp 3.0.1 completed successfully

** CALCRPSF 1.3.1   Finished
```

The resulting file can now be plotted with QDP (if you skip rpsfqdp then the resulting file can be plotted using `fplot`).

Notes & Warnings:

- The spectral weighting (using the FITS spectrum extracted via xselect) will be available in the next release, as will the facility to apply region masks.

7.4 Performing mathematical operations on PHA files

The ftool `mathpha` can be used to perform mathematical operations on PHA files. The calculation can optionally be carried out units of COUNTS or RATE, the user is advised to **carefully** read the help which is available before using this task.

Example:

```
mathpha
** MATHPHA 3.0.3
Expression to be evaluated[] spectrum1.pha / spectrum2.pha
Units algebraic expression to be performed in (C,R,F or ?)[C] ?


... This parameter indicates the physical units in
which the algebraic expression is to be evaluated,
and the units in which the output file will be written.
The allowed values are C (ie COUNTS), or R (ie RATE),
implying that the algebra will be performed in COUNTS
or COUNTS PER SECOND space respectively. The algebra
will be in this space irrespective of whether the input
files contain data stored in counts or in counts per
second (ie, if units=C, input PHA histograms stored in
counts per sec will be converted to counts prior to any
mathematical operations being performed). Similarly, this
flag gives the implied units of any numerical constants
within the input expression. A third option value is also
allowed, units=F (ie FILE) whereby the algebra is performed
in which ever units most of the files are stored in (COUNTS
in the event of a tie).

Units algebraic expression to be performed in (C,R,F or ?)[?] C
O/p PHA filename[] spectral.ratio
Exposure time flag/value ({value},{file},CALC,NULL or ?)[] ?

  This parameter controls the value written as the exposure time in the
  output file. The allowed values are:
```

```
          {a numerical value}
        - where the entered value (assuming it can be parsed as a real), is
        written to the output dataset. The value will be assumed to be in
        units of seconds. At the current time, numerals of the form  1E04
        or 1E-04 are NOT supported.
    {an input filename}
        - whereby the exposure time read from the specified input file is
        written to the output dataset
    CALC
        - where the exposure time is calculated from the input expression
        by substituting each filename with its exposure time, and performing
        the specified calculation.
    NULL
        - where an exposure of 1 second is written to the output dataset

        Exposure time flag/value ({value},{file},CALC,NULL or ?)[] NULL
        Number of comment strings to be added (up to 4)[1]
        Comment 1[] This is a ratio of spectrum1.pha/spectrum2.pha
        ...... processing file: spectrum1.pha
        ...... processing file: spectrum2.pha
        ... performing algebra in units of COUNTS
        ** FILOP 2.0.0   WARNING:
        ... Poissonian errors with N<20 assumed  for:
        ...... channels     1.-    7.
        ...... channels   201.-  201.
        ...... channels   205.-  208.
        ...... channels   210.-  213.
        ...... channels   216.-  256.
        ** FILOP 2.0.0   WARNING:
        ... Divide-by-zero encountered for:
        ...... channels     1.-    7.
        ...... channels   252.-  252.
        ...... channels   256.-  208.
        ... These channels have been set bad, and  value DIVZERO inserted
        ... written the PHA data Extension
        ** MATHPHA 3.0.3   Finished
```

This made a ratio of two spectra, the user can plot the output file using FPLOT (use FSTRUCT and FLCOL to examine the file first, so you know what column names to give to FPLOT).

7.5 Creating particle background spectra

Particle background spectra are required for use in background subtraction of extended sources, or in any case where the background must be estimated using a significantly different part of the PSPC detector than the region where the source data were taken. The sky X-rays have a different radial dependance than the particle background, so simply correcting the background spectrum for the different vignetting effect is inadequate

`pcparpha` generates a 256-channel PHA spectrum of the particle background in the ROSAT PSPC from the parameterization of Plucinsky etal (1993 ApJ 418, 519)

Example:

```
pcparpha
** PCPARPHA 2.0.1
O/p PHA filename[particle_spectrum.pha]
EVR filename[rp100000.evr]
GTI filename[rp100000.fits]                        (the events file)
WMAP/Off-axis histogram filename[target.pha]
No. pixels in source region (or '%' or '?')[%] ?
 ... This parameter is the number of pixels for which the
     particle background spectrum is to be calculated.
     This information should also be in the Off-axis histogram
     dataset. Enter "%" to use the stored values or enter
     an integer to override the stored values. In the
     latter case, you will be prompted for the pixel size next.
No. pixels in source region (or '%' or '?')[?] %
.... No. theta bins to process:          20
... No. EVR events to process:          323
                                  theta      weight (%)
...... finished theta bin   1 (1.5000    ,  0.000000)
... written the PHA data Extension
** PCPARPHA 2.0.1   Finished
```

Chapter 8

Quick Summary

8.1 Useful xselect commands

```
       clear - Clear input files, filters, selections, or everything
         cpd - Equivalent to set device
        exit - Exit from xselect
     extract - Extract image, spectrum or (light) curve from data
        help - Obtain help on commands and syntax
      filter - Set filters to be applied to the next extraction
        plot - Plot spectrum, light curve, image, HK parameter(mkf), or other
        quit - Exit from xselect
 read events - Read in the events data
        save - Save current session, product FITS file, spectrum, light
               curve, image
  select mkf - Apply Boolean expressions for data selection, for creating
               GTI intervals from HK or filter files, and for filtering the
               observation catalogue.
         set - Sets the mission, instrument and plot device, and the
               column names, and binsizes for extract.
        show - Show data files, obscat, parameters, selections, or status
        stop - Exit from xselect
```

8.2 ROSAT ftools

```
         abc - Apply Barycentric Corrections to ROSAT event data.
         bct - Produce a Barycenter Correction Table from ROSAT orbit data
      extpha - [P] Extracts source spectrum files from PSPC _src files
  fixrev0pha - Changes  the  CDELT1 and CDELT2 keywords in a ROSAT PHA file
```

```
 gevt2rdf - Converts a German _events dataset to RDF _bas format
   hiback - [P] Creates _qsrc file from _src file
hriaspcor - Use Morse's method to correct  aspect errors
hriexpmap - Creates exposure map for given ROSAT HRI observational dataset
  hrifilt - Creates a ROSAT HRI makefilter file
  hrirpsf - Calculates theoretical HRI PSF
   img2us - Converts from German Image file format to US RFITS format
   mkqsrc - Adds extra fields for quality screening to ROSAT source file
mk_allevt - [P] Combines standard \& rejected events tables of an RDF file
    pcarf - Generates a ROSAT PSPC ARF for an input PHA file
   pcecor - Corrects PI channel in Events file
 pcexpmap - Creates exposure map for ROSAT PSPC observation
   pcfilt - Creates a ROSAT PSPC makefilter file
 pcparpha - Generates a background PHA  spectrum  for ROSAT PSPC
  pcpicor - [P] Corrects PI channel in Events file
pcsasscor - Corrects PI channel in PSPC events file
   pcrpsf - Calculates a theoretical radial PSF for the ROSAT PSPC
   pctcor - Corrects PI channel in Events file
   pcecor - Applies energy dependant correction to PI channel in Events file
  rosbary - [P] Perform barycentric correction for ROSAT. (Runs abc \& bct)
  src2pha - Creates a PHA file from specified extensions in an RDF src file

    [P] Perl script.  You must have Perl installed to use this tool.
```

8.3 Other FTOOLS for ROSAT Analysis

Some FTOOLs used in ROSAT analysis are applicable to data from other missions, and so reside in a more general FTOOLS package:

```
  calcrpsf - Calculates a radial point spread function
 caldbinfo - Checks whether a local  CALDB  is  correctly  installed
             for a user
   rbnrpsf - Rebins a radial point spread function
  rpsf2eef - Converts  a  radial  PSF  (RPSF)  dataset  to  a  radial
             encircled energy function (REEF) dataset.
   rpsfqdp - Converts OGIP format radial profile file to ASCII QDP
   st2rpsf - Reads STWFITS file and writes an OGIP RPSF file
   stw2pha - Converts stw PHA file to OGIP FITS format
       bkg - Subtract a background from a binned light curve
   extrpsf - ExtractS radial PSF (RPSF) dataset from an event file.
    grppha - Manipulates OGIP standard PHA FITS file
   mathpha - Performs mathematical operations on PHA files
    rbnpha - Compresses  a  PHA dataset
     burst - Remove bursts of events from time ordered event list
```

```
    farith - Perform arithmetic on 2 FITS images
     fdump - Dump contents of a FITS table to an ASCII file
     fhelp - Display help pages for an FTOOL task
 fkeyprint - Display keyword(s) in FITS header(s)
     flcol - List column information in a FITS table extension
     fplot - Plot columns in a FITS file using QDP/PLT package
fstatistic - Calculate mean, standard deviation, min, and max for a column
   fstruct - List a description of the structure of a FITS file
     plist - List parameters from task parameter files
      pset - Set one or more parameters in task parameter file
  punlearn - Get a copy of the system parameter file (reset task pars)
    sky2xy - Transform from sky to pixel coordinates using WCS keywords
    xy2sky - Transform from pixel to sky coordinates using WCS keywords
```

Chapter 9

An Example Xselect Session

comments in italics

```
        xselect

        ** xselect V1.1a **

        > Enter session name >[xsel] session1


session1:ASCA > read events
> Enter the Event file dir >[./]
> Enter Event file list >[] rp700000.fits
Got new mission: ROSAT and new instrument: PSPC
> Reset the mission and instrument? >[yes]
 This is ROSAT

 Notes: xselect set up for      ROSAT
 Keywords for time and pha are  TIME       PI
 Units of time are              s
 Keywords for images are          X          Y


Setting: IMAGE binning    =   15
         WMAP binning     =   15
         Energy Column    =  PI
```

```
         Energy rebinning =    1
         WMAP keywords = DX         DY


Getting Min and Max for Energy Column...
Got min and max for PI:     0    255

Number of files read in:   1

******************** Observation Catalogue ********************

Data Directory is: /disk/user/agn/
HK Directory is: /disk/user/agn/


        INSTRUME DATE-OBS XS-RAPT  XS-DECPT XS-OBSID XS-LIVTI
      1 PSPC     07/11/93 .218E+03 .623E+02 US700000 .952E+04


session1:ROSAT-PSPC > extract image
  Extractor 1.0q
 XPI 2.1t
 Doing file: /disk/user/agn/rp700000.fits
 Getting FITS WCS Keywords

100% completed
    Total      Good    Bad: Region      Time     Phase       PHA
   124810    123847             0         0         0       963
============================================================================
   Grand Total      Good   Bad:  Region      Time     Phase       PHA
        124810    123847             0         0         0       963
 Total Counts for image :   123847
 Total Time for image :     9625.0000000000
 Total Counts/Time for image :     12.867220779221
session1:ROSAT-PSPC > saoimage
```

define regions and save to a file, as detailed in section 3.1.2

```
session1:ROSAT-PSPC > filter region target.reg
session1:ROSAT-PSPC > set binsize 200
session1:ROSAT-PSPC > extract curve exp=0.1
```

(*200 s time bins set*)

(*minimum partial bin 20s*)

```
 Extractor 1.0q
 XPI 2.1t
 Doing file: /disk/user/agn/rp700000.fits
100% completed
    Total      Good    Bad: Region      Time     Phase       PHA
   124810    123847             0         0         0       963
===========================================================================
   Grand Total      Good   Bad:  Region      Time     Phase       PHA
        124810    123847             0         0         0       963
 Total Counts for Fits Light Curve :   123847
 Total Time for Fits Light Curve :     9625.0000000000
 Total Counts/Time for Fits Light Curve :     12.867220779221
session1:ROSAT-PSPC > plot curve

Note: cursor timing selection is now done with the command
   FILTER TIME CURSOR
```

(*the light curve has a few bad points which i wish to exclude*)

```
session1:ROSAT-PSPC > filter time cursor
Type:
  - click to include a time region
c - to cancel cursor selection
e - makes the selection excluded
l - puts cursor on the left
r - puts cursor on the right
x - to exit cursor selection
p - to return to PLT prompt
  - click outside the plot to cancel a selection

Enter QUIT at PLT prompt to continue
```

(*2 clicks define a time for inclusion and these are written to* session1_cursor_gti001.xsl *save these times to a file so you can use them again in a future session*)

```
session1:ROSAT-PSPC > save time cursor
> Give output file name >[] good.times

session1:ROSAT-PSPC > extract spectrum
 Extractor 1.0q
```

```
 XPI 2.1t
 Doing file: /disk/user/agn/rp700000.fits
 Getting FITS WCS Keywords

100% completed
    Total      Good    Bad: Region      Time     Phase       PHA
   124810     17847              0    106781         0       182
==========================================================================
   Grand Total      Good   Bad:  Region      Time     Phase       PHA
        124810     17847              0    106781         0       182
 Total Counts for spectrum :   17847
 Total Time for spectrum :     1401.5493164062
 Total Counts/Time for spectrum :     12.733765263260
session1:ROSAT-PSPC > save spectrum
> Give output file name >[high_state]
The data will be grouped using the following:

Bad 1-11, 212-256&group 12-211 10&show group&show quality&exit

> Group ( or rebin ) the spectra before outputting? >[yes]
```

(*if the spectrum you are saving is a background spectrum then do not group the data*)

```
--------
 GROUPING
 --------
 ---------------------------------------------
 Channel Grouping (Channel-Channel) :
    1 -  11  are single channels
   12 - 211  are grouped by a factor   10
 ---------------------------------------------


 -------
 QUALITY
 -------
 Bad Channels (Channel - Channel)
 ---------------------------------------------
    1 -  11  have quality 5
  212 - 256  have quality 5
 ---------------------------------------------

Wrote spectrum to high_state.pha

session1:ROSAT-PSPC > exit
Do you wish to save this session?> no
```