Compute LETG/ACIS−S Grating ARFs

CIAO 3.4 Science Threads

Table of Contents

Compute LETG/ACIS−S Grating ARFs

CIAO 3.4 Science Threads

Overview

Last Update: 26 Feb 2007 − fullgarf v4.0.1: Four new parameters have been added, all of which are used by the `mkgarf` tool: `pbkfile`, `dafile`, `osipfile`, and `maskfile`; see "ahelp fullgarf" for details on each new parameter. Added ACIS dead area correction section

Synopsis:

`fullgarf` is a script that creates a grating ARF for a particular order and grating of an observation. While the `mkgarf` tool will create a grating ARF for an individual chip given an aspect histogram, this script creates ARFS for each chip, creating aspect histograms as necessary. The script then combines the individual ARFS into one for the full array.

Purpose:

To create grating ARFs for an LETG/ACIS−S observation.

Read this thread if:

you are working with an LETG/ACIS−S observation and intend to model and fit the spectra in *Sherpa*.

Calibration Updates:

- ***CALDB v2.26*** ***(2 Feb 2004):*** The ACIS contamination files were added to the CALDB. The Account for the ACIS QE Degradation section of this thread describes how it is taken into account in the data analysis.
- ***CALDB v2.8*** ***(13 Sep 2001):*** New OSIP files make it possible to analyze −120 degrees S3 grating data with the gainmaps and FEFs that were in the prevous CALDB release. If you are working with an observation taken under these conditions, run the Apply an ACIS Gain Map and Obtain Grating Spectra from HETG/ACIS Data threads before running this thread to ensure the use of the correct gainmap and OSIP files.

Related Links:

- Analysis Guide for Chandra High Resolution Spectroscopy: an in−depth discussion of grating analysis.
- Continuous Clocking Mode why topic: additional information for users working with CC−mode data.

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

Get Started

Sample ObsID used: 1198 (LETG/ACIS−S, 3C 273)

File types needed: evt2; pha2; asol1; bpix1; msk1

If you created a new bad pixel file by running the Create a New ACIS Bad Pixel File: Identify ACIS Hot Pixels and Cosmic Ray Afterglows thread, use that file in this analysis. Otherwise, use the `bpix1.fits` file from the Archive. Make sure that you have set up ardlib to use the same bad pixel file.

It is assumed that you have created a gRMF for your observation by running the Create Grating RMFs for ACIS−S Observations thread, which is called `rmf.fits`.

Download the script

This thread uses the `fullgarf` script; for information about the script, consult the help file ("`ahelp fullgarf`"). The most recent version of `fullgarf` is v4.0.1 (12 February 2007):

```
unix% grep version_ `which fullgarf`
version_str="4.0.1"
version_date="12 February 2007"
```

Please check that you are using the most recent version before continuing. If you do not have the script installed or need to update to a newer version, please refer to the Scripts page.

Using the mask file

To make sure that the gARF file is correct, set the `maskfile` to the observation−specific msk1.fits file. ***If you are working with continuous−clocking (CC) mode data***, be sure to read the ACIS Pipeline−Processed Mask Files caveat as well.

The `maskfile` parameter was added to the `fullgarf` parameter file in v4.0.1. Prior to that version, it had to be set in `mkgarf.par` before running this script.

The ACIS Dead Area Correction

There is a fractional area loss per unit time due to cosmic ray flux incident on the ACIS detector. Calibration to account for this ACIS "dead area" was included in CALDB 3.3.0 on 15 December 2006. The correction is non−zero for the 8 front−illuminated ACIS chips; the effect is not detectable for the BI chips, so the nominal calibration value is 0.0. The resulting chipy−dependent reduction in the EA will be approximately 2.2% at the readout, and 4.0% at the top of the chip. Refer to the ACIS Dead Area Correction why topic for technical details.

In CIAO 3.4, the application of the dead area correction is ***turned off*** by default. However, users may opt to include it in the analysis by setting the `pbkfile` and `dafile` parameters in this script. Refer to the script help file for details on these parameters.

Account for the ACIS QE Degradation

The effective low−energy ACIS QE is lower now than it was at launch. This problem is thought to be associated with the deposition of one or more materials on the ACIS detectors or optical blocking filters. Since the depth of these contaminants is growing with time, the effective low−energy QE is becoming lower as time passes. A correction for this degradation should be incorporated when creating ACIS response files.

Refer to the ACIS QE Degradation why topic for more information.

Determine the Orders of the Observation

An ARF needs to be calculated for each order in the observation. We can use *Prism* to examine the pha2 file and determine how many orders there are:

```
unix% prism acisf01198N002_pha2.fits &
```

The `tg_m` column indicates the order of the observation (+/− 1, +/− 2, +/− 3) and the `tg_part` column indicates the grating (1 = HEG, 2 = MEG, 3 = LEG). In this example there are six rows (all +/− orders for LEG) for the LETG observation.

Run fullgarf

We will have to run `fullgarf` six times, once for each row in the PHA file.

For row 2, LEG, order = −2:

```
unix% punlearn fullgarf
unix% pset fullgarf phafile=acisf01198N002_pha2.fits
unix% pset fullgarf pharow=2
unix% pset fullgarf evtfile=acisf01198N002_evt2.fits
unix% pset fullgarf asol=pcadf063832996N002_asol1.fits
unix% pset fullgarf engrid="grid(rmf.fits[cols ENERG_LO,ENERG_HI])"
unix% pset fullgarf dtffile=")evtfile"
unix% pset fullgarf badpix=acis_1198_new_bpix1.fits
unix% pset fullgarf maskfile=acisf01198_000N002_msk1.fits
unix% pset fullgarf rootname=acisf01198
```

In many cases there will be more than one asol1.fits file for an observation. ***All*** the files must be input to the `asol` parameter, either as a list or as a stack (see ahelp stack for more information).

If you wish to include the ACIS dead area correction (not applied in this thread), set the `pbkfile` and `dafile` parameters as well:

```
unix% pset fullgarf pbkfile=acisf063833861N002_pbk0.fits dafile=CALDB
```

Now run the tool:

```
unix% fullgarf
Will use /home/username/cxcds_param/fullgarf.par for the parameter file.

/home/username/cxcds_param/fullgarf.par contains 15 parameters . . .

Input PHA file (Type I or II) (acisf01198N002_pha2.fits):
Row in Type II PHA file (ignored if Type I) (0:99) (2):
Event file  (acisf01198N002_evt2.fits):
Aspect offsets file (pcadf063832996N002_asol1.fits):
Energy grid spec (grid(rmf.fits[cols ENERG_LO,ENERG_HI])):
Dead time correction factor; ACIS−>evt file; HRC −> dtf file (')evtfile'):
Bad pixel file; (filename|NONE|CALDB) (acis_1198_new_bpix1.fits):
Output rootname (acisf01198):
NONE, or name of ACIS window mask file (acisf01198_000N002_msk1.fits):
Getting the pha file type . . .
Grating arm is LEG, order=−2
Source location is X=4162.160156250, Y=4084.7800292969

Detector is ACIS
Will run asphist for ccd_id= 4 5 6 7


asphist infile=pcadf063832996N002_asol1.fits outfile=acisf01198_ah4.fits
evtfile=acisf01198N002_evt2.fits[ccd_id=4] dtffile=acisf01198N002_evt2.fits

asphist infile=pcadf063832996N002_asol1.fits outfile=acisf01198_ah5.fits
evtfile=acisf01198N002_evt2.fits[ccd_id=5] dtffile=acisf01198N002_evt2.fits

asphist infile=pcadf063832996N002_asol1.fits outfile=acisf01198_ah6.fits
evtfile=acisf01198N002_evt2.fits[ccd_id=6] dtffile=acisf01198N002_evt2.fits

asphist infile=pcadf063832996N002_asol1.fits outfile=acisf01198_ah7.fits
evtfile=acisf01198N002_evt2.fits[ccd_id=7] dtffile=acisf01198N002_evt2.fits

Finished creating aspect histograms for ccd_id= 4 5 6 7
Will run mkgarf for the same ccd_id list

mkgarf detsubsys=ACIS−S0 order=−2 grating_arm=LEG outfile=acisf01198_S0_LEG_−2.fits asphistfile=acisf01198_
4.fits[ASPHIST] engrid=grid(rmf.fits[cols ENERG_LO,ENERG_HI]) osipfile=CALDB dafile=NONE pbkfile=NONE
maskfile=acisf01198_000N002_msk1.fits mode=hl verb=0

mkgarf detsubsys=ACIS−S1 order=−2 grating_arm=LEG outfile=acisf01198_S1_LEG_−2.fits asphistfile=acisf01198_
5.fits[ASPHIST] engrid=grid(rmf.fits[cols ENERG_LO,ENERG_HI]) osipfile=CALDB dafile=NONE pbkfile=NONE
maskfile=acisf01198_000N002_msk1.fits mode=hl verb=0

mkgarf detsubsys=ACIS−S2 order=−2 grating_arm=LEG outfile=acisf01198_S2_LEG_−2.fits asphistfile=acisf01198_
6.fits[ASPHIST] engrid=grid(rmf.fits[cols ENERG_LO,ENERG_HI]) osipfile=CALDB dafile=NONE pbkfile=NONE
maskfile=acisf01198_000N002_msk1.fits mode=hl verb=0

mkgarf detsubsys=ACIS−S3 order=−2 grating_arm=LEG outfile=acisf01198_S3_LEG_−2.fits asphistfile=acisf01198_
7.fits[ASPHIST] engrid=grid(rmf.fits[cols ENERG_LO,ENERG_HI]) osipfile=CALDB dafile=NONE pbkfile=NONE
maskfile=acisf01198_000N002_msk1.fits mode=hl verb=0


Finished creating grating arfs for ccd_id= 4 5 6 7

dmarfadd @thisfile.istemporary acisf01198LEG_−2_garf.fits

fullgarf finished.
```

You may see an aspect−related error from `mkgarf`; it is explained in this FAQ and may be ignored. The content of the parameter file may be checked using plist fullgarf.

Summary

The thread is now complete. The grating ARFs for this dataset are:

```
  acisf01198LEG_−1_garf.fits    acisf01198LEG_1_garf.fits
  acisf01198LEG_−2_garf.fits    acisf01198LEG_2_garf.fits
  acisf01198LEG_−3_garf.fits    acisf01198LEG_3_garf.fits
```

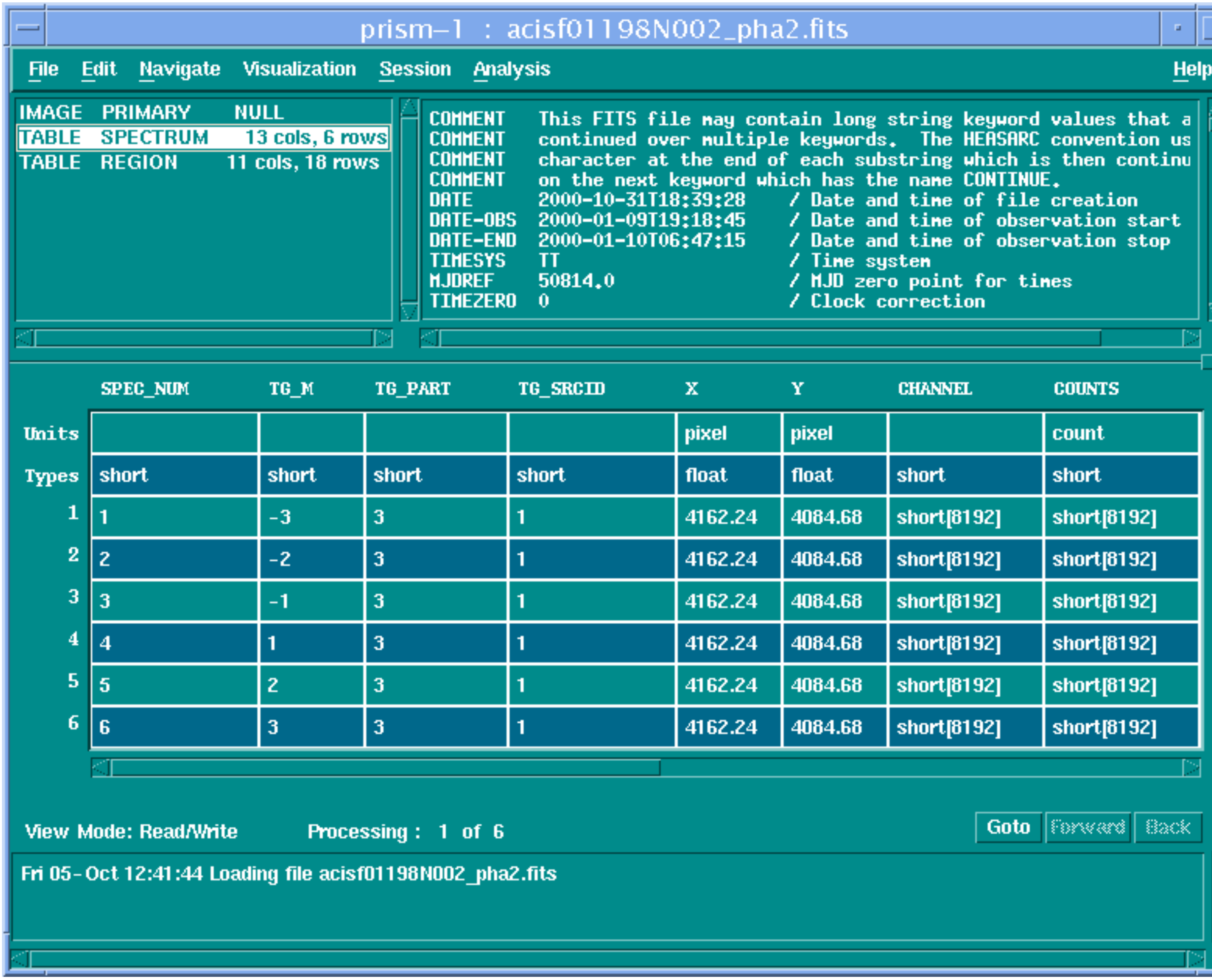
```
Parameters for /home/username/cxcds_param/fullgarf.par


       phafile = acisf01198N002_pha2.fits Input PHA file (Type I or II)
        pharow = 2                Row in Type II PHA file (ignored if Type I)
       evtfile = acisf01198N002_evt2.fits Event file
          asol = pcadf063832996N002_asol1.fits Aspect offsets file
        engrid = grid(rmf.fits[cols ENERG_LO,ENERG_HI]) Energy grid spec
       dtffile = )evtfile −> acisf01198N002_evt2.fits Dead time correction factor; ACIS−>evt file; H
        badpix = acis_1198_new_bpix1.fits Bad pixel file; (filename|NONE|CALDB)
      rootname = acisf01198       Output rootname
      maskfile = acisf01198_000N002_msk1.fits NONE, or name of ACIS window mask file
      (pbkfile = NONE)            NONE, or the name of the parameter block file
       (dafile = NONE)            NONE, CALDB, or name of ACIS dead−area calibration file
     (osipfile = CALDB)           NONE or Name of fits file with order sorting info
      (clobber = no)              Clobber existing output files? This is passed to ALL child process
      (verbose = 0)               Control the level of diagnostic output. 0=>least.
         (mode = hl)              Mode flags.  Set to 'ql' to enable querying, 'hl' to suppress.
```

History

23 Dec 2004	updated for CIAO 3.2: canned gRMFs are no longer available in the CALDB
06 Dec 2005	updated for CIAO 3.3: the `fullgarf` script has been updated to version 3.3.1 for the new asphist tool syntax; corresponding changes to screen output
01 Dec 2006	reviewed for CIAO 3.4: no changes
26 Feb 2007	fullgarf v4.0.1: Four new parameters have been added, all of which are used by the `mkgarf` tool: `pbkfile`, `dafile`, `osipfile`, and `maskfile`; see "ahelp fullgarf" for details on each new parameter. Added ACIS dead area correction section

URL: http://cxc.harvard.edu/ciao/threads/mkgarf_letgacis/ Last modified: 26 Feb 2007

Image 1: Viewing the pha2 file in prism



```
IMAGE  PRIMARY     NULL
TABLE  SPECTRUM    13 cols, 6 rows
TABLE  REGION      11 cols, 18 rows
```

```
COMMENT   This FITS file may contain long string keyword values that a
COMMENT   continued over multiple keywords.  The HEASARC convention us
COMMENT   character at the end of each substring which is then continu
COMMENT   on the next keyword which has the name CONTINUE.
DATE      2000-10-31T18:39:28     / Date and time of file creation
DATE-OBS  2000-01-09T19:18:45     / Date and time of observation start
DATE-END  2000-01-10T06:47:15     / Date and time of observation stop
TIMESYS   TT                      / Time system
MJDREF    50814.0                 / MJD zero point for times
TIMEZERO  0                       / Clock correction
```

	SPEC_NUM	TG_M	TG_PART	TG_SRCID	X	Y	CHANNEL	COUNTS
Units					pixel	pixel		count
Types	short	short	short	short	float	float	short	short
1	1	-3	3	1	4162.24	4084.68	short[8192]	short[8192]
2	2	-2	3	1	4162.24	4084.68	short[8192]	short[8192]
3	3	-1	3	1	4162.24	4084.68	short[8192]	short[8192]
4	4	1	3	1	4162.24	4084.68	short[8192]	short[8192]
5	5	2	3	1	4162.24	4084.68	short[8192]	short[8192]
6	6	3	3	1	4162.24	4084.68	short[8192]	short[8192]

View Mode: Read/Write   Processing : 1 of 6

```
Fri 05-Oct 12:41:44 Loading file acisf01198N002_pha2.fits
```