Chandra X-ray Center

AHELP for CIAO 3.4

mkgarf

Context: tools

Jump to: Description Examples Parameters CHANGES IN CIAO 3.4 CHANGES IN CIAO 3.0 Notes Bugs See Also

Synopsis

Generate a Chandra Grating ARF for one detector element.

Syntax

```
mkgarf  asphistfile outfile order sourcepixelx sourcepixely engrid
obsfile osipfile maskfile detsubsys grating_arm [mirror] [pbkfile]
[dafile] [ardlibparfile] [geompar] [verbose] [clobber]
```

Description

The grating ARF (auxiliary response function) is a table of effective area vs. energy for a specific observation. It depends upon the observational details through the aim–point, which determines where the mean chip–gap locations are in the spectrum, and through the aspect, which moves the image around on the detector.

This program computes the ARF for a single detector element. They can be combined with dmarfadd or a shell script which makes the grating ARF for an observation (including asphist and mkgarf for each chip, followed by dmarfadd); see help for fullgarf.

The mkgarf program generates an ARF file in standard format accepted for use in spectral analysis packages such as xspec or sherpa. It requires a parameter file and an aspect histogram as its minimum set of input files. Additionally, the energy grid may be specified via a grid parameter, an ASCII file, or by reference to a grating RMF. A scale factor due to order–sorting clipping of the PHA can be applied through the OSIP (Order Sorting, Integrated Probability) file parameter (see below, and tg_resolve_events).

The detailed definition of a grating ARF is beyond the scope of this document. The reader is referred to Davis (2001) for its definition and its use in spectral analysis.

Calibration information (QEs, effective areas, bad pixels) is computed by the ARDLIB software library that is linked to mkgarf. ARDLIB has its has its own parameter file, ardlib.par, which contains more than 50 parameters. Consult the ARDLIB documentation for information about specific ardlib.par parameters.

Although this program may be used to compute a type of zeroth order grating ARF, its use is not recommended for this purpose. Instead, the user should use the `mkarf' program.

While the output product is back−compatible with existing software and formats, we have extended the format to add some grating−specific content. The table's columns are:

Col Num	Name	Unit	Type
1	ENERG_LO	keV	Real4
2	ENERG_HI	keV	Real4
3	SPECRESP	cm**2	Real4
4	BIN_LO	angstrom	Real4
5	BIN_HI	angstrom	Real4
6	FRACEXPO	1	Real4

The BIN_LO and BIN_HI columns are CXC additions; they are by default filled with the bin wavelengths in Angstroms (which is usually a linear coordinate). FRACEXPO is also a CXC addition. It is the fraction of the exposure on good detector area (without dither, it would be 1.0 on good pixels, 0.0 for bad pixels or coordinates off the detector). FRACEXPO is required by pileup modeling programs.

The file header contains keywords identifying the grating, order, and zero−order source; for example:

GRATING	HETG	String	Grating
TG_SRCID	0	Int4	source index
TG_M	1	Int4	Grating Order
TG_PART	2	Int4	MEG(2), HEG(1), or LEG(3)

Note that mkgarf operates on one chip at a time. These individual responses are then weighted and added by dmarfadd. A shell script is available (fullgarf) which run the chip−dependent programs (asphist, mkgarf), and combines the responses into one ARF (superseded scripts are mkgarf_hetgs and mkgarf_letgs).

Example 1

```
mkgarf detsubsys="ACIS−S1" outfile="acis_s1_garf.fits" \
engrid="grid(acis_grmf.fits[2][cols ENERG_LO,ENERG_HI])" \
order=−1 asphistfile="acisf01318_ah5.fits[ASPHIST]"
```

Compute a grating ARF for order −1 on ACIS−S1 on an energy grid defined in acis_rmf.fits (in particular columns ENERG_LO and ENERG_HI of the second block in the file) and write it to acis_s1_arf.fits.

Example 2

```
mkgarf detsubsys="ACIS−S2" outfile="acis_s2_garf.fits" \
engrid="grid(acis_s2_grmf.fits[2][cols ENERG_LO,ENERG_HI])" \
order=−3 asphistfile="acisf00459_ah6.fits[ASPHIST]"
pbkfile=acisf063875928N002_pbk0.fits dafile=CALDB
```

                                                                    Example 1

Compute a grating ARF for order –3 on ACIS–S2. The pbkfile and dafile parameters are set to apply the dead area correction.

Example 3

```
mkgarf detsubsys="ACIS-S4;QUADRANT=1" outfile="acis_s4q1_arf.fits" \
engrid="0.1:10.0:0.01"
```

Compute a grating ARF for use with events that fall into the first quadrant. The last example illustrates the use of ARDLIB qualifiers or options. In particular, this causes the QE computed by ARDLIB to be 0 for pixels not in the first quadrant of ACIS–S4. The energy grid runs from 0.1 keV to 10.0 keV in steps of 0.01 keV.

Example 4

```
mkgarf detsubsys="ACIS-S2;contam=no"
outfile="acis_s2_nocontam_arf.fits" \
engrid="0.1:10.0:0.01"
```

Compute a grating ARF without including the ACIS contamination. This is another example illustrating the use of ARDLIB qualifiers or options.

Parameters

name	type	ftype	def	reqd
asphistfile	string			yes
outfile	string			yes
order	integer			yes
sourcepixelx	real			yes
sourcepixely	real			yes
engrid	string			yes
obsfile	string			yes
osipfile	string			yes
maskfile	string			yes
detsubsys	string			yes
grating_arm	string			yes
mirror	string			
pbkfile	string	input		
dafile	string	input		
ardlibparfile	string			
geompar	string		geom	
verbose	integer			
clobber	boolean			

Example 3   3

Detailed Parameter Descriptions

Parameter=asphistfile `(string required)`

The name of file containing the binned aspect history in the form of an aspect histogram (see asphist).

Note that each ACIS CCD has its own GTI file. This means that each chip will have its own aspect histogram file.

Variations in the the mirror effective area are assumed to be small on the scale of the dither pattern. Hence, the mean aspect offsets will be used to deduce the mirror effective area. This means that the aspect offsets should be small, typically less than 100 ACIS sky pixels (50 arcsec) in magnitude.

Parameter=outfile `(string required)`

The name of the grating ARF file that this program will generate.

Parameter=order `(integer required)`

The diffraction order. The magnitude is limited by the orders' efficiencies tabulated in the calibration database grating files. (–11 to +11 for HEG and MEG, and –25 to 25 for LETG).

Parameter=sourcepixelx `(real required)`

Parameter=sourcepixely `(real required)`

These parameters specify the (x,y) sky position of the point source, in the appropriate tangent plane pixel system for the detector. This should be the same as the zero–order centroid in the region used by tg_resolve_events for the source, and which are tabulated in the binned spectrum file (`pha2.fits').

Parameter=engrid `(string required)`

Energy grid specification string. This string may specify either a file or an explicit energy grid. For example, to read a grid from the block MATRIX of an RMF file called rmf.fits, use:

```
"engrid=grid(rmf.fits[MATRIX][cols ENERG_LO,ENERG_HI])"
```

To specify an explicit grid running from 0.3 keV to 10.0 keV in 0.01 keV increment steps, use:

```
"engrid=0.3:10.0:0.01"
```

To use an ASCII file which contains two columns, the energ_lo and energ_hi, use:

```
"engrid=grid(myfile.tbl)"
```

Note that the preferred grid is linear in wavelength, since this reflects the natural dispersion of photons onto the detector. For back–compatibility, we write the grid in energy units in ascending order. To get a linear–wavelength grid, the first or third engrid forms must be used. Energy grids are required to be in ascending order.

Parameter=obsfile `(string required)`

The name of the FITS file that contains information about the observation via FITS keywords. This file should contain keywords that specify the mission, detector, SIM offsets, etc. The aspect histogram file should contain all the observational information. A level 1 event file will also suffice.

Parameter=osipfile `(string required)`

This parameter is used only when computing an ARF for an ACIS chip. Its value specifies the name of the order sorting table used extract nth order counts based upon the spatially dependent PHA fractions associated with the extraction region. "OSIP" stands for "Order Sorting and Integrated Probability". A value of "NONE" may be used to indicate that no such file is to be applied. If "NONE" is used, the it is assumed that the integrated probability is 1.0. The value, CALDB, indicates that the highest version for the most recent date before the observation date will be retrieved from the calibration data base.

Parameter=maskfile `(string required)`

The mask file (msk1.fits) for the observation. The mask file is needed in particular when a window or subarray was used. A value of "NONE" indicates that no mask file will be applied.

Parameter=detsubsys `(string required)`

The value of this parameter is passed to ARDLIB to select a particular detector subsystem so that the appropriate QE can be computed. It specifies the specific chip or MCP of the detector for which the exposure map is to be computed. For Chandra, it must specify one of the following detector subsystems:

```
HRC-I
HRC-S1, HRC-S2, HRC-S3
ACIS-I0, ACIS-I1, ACIS-I2, ACIS-I3
ACIS-S0, ACIS-S1, ACIS-S2, ACIS-S3, ACIS-S4, ACIS-S5
```

The following qualifiers or options are supported for these detectors:

```
QE=value
IDEAL (equivalent to QE=1)
WINDOW=xmin,ymin,xmax,ymax
REGION=BOX(xcenter,ycenter,xsize,ysize)
REGION=RECTANGLE(xmin,xmax,ymin,ymax)
```

Note that only simple rectangular regions are supported.

For ACIS, one may select filter bad pixels based upon the status column in the bad pixel files by using the BPMASK qualifier:

```
BPMASK=hex value (Specified as, e.g., 0x39ff)
BPMASK=dec value (Specified as, e.g., 14847)
BPMASK=FAINT (=0x39ff, 14847)
```

Parameter=obsfile (string required)   5

```
BPMASK=VFAINT (=0x3fff, 16383)
```

If this option is not present, the default is FAINT.

Parameter=grating_arm `(string required)`

The grating arm to be used. The value of this parameter must be one of "HEG", "MEG", or "LEG".

Parameter=mirror `(string)`

For Chandra (a.k.a. AXAF), the mirror parameter must be set to "HRMA". However, the following ardlib qualifiers are also supported:

```
AREA=value
SHELL=value (possible values: 1,3,4,6)
BITMAP=value
```

For example, to compute an instrument map for using shells 1 and 3, which corresponds to the bitmap "1100", use "HRMA;bitmap=1100" for the value of this parameter. To compute using a mirror area of 1, use "HRMA;AREA=1".

Parameter=pbkfile `(string filetype=input)`

The parameter block file, which defines ACIS pixel clocking parameters, is a standard product of pipeline processing and is available for every observation. This file contains information which defines how long any pixel is exposed before being read−out, which is related to the probability that any pixel will be disabled ("deadened") by cosmic rays. See the description of the "dafile" parameter for more information on ACIS dead area.

Parameter block files have names of the form, "acisf146860615N001_pbk0.fits". The long string of digits refers to the time of observation (seconds since reference date) and "N001" is a revision number.

Since the pbkfile contents are observation−dependent, there is no default other than "NONE". In this case, no dead area correction is applied, and dafile is assumed to have the value "NONE". If pbkfile is set to a valid file, then the dafile parameter must also refer to a calibration file.

The pbkfile parameter is ignored for HRC data.

Parameter=dafile `(string filetype=input)`

ACIS "dead area" coefficients file, which may have the values "NONE" (no dead area computation), CALDB (for automatic lookup), or an explicit file reference to an ACIS "dead area" coefficients FITS table.

The ACIS dead area refers to a slight decrease in detector efficiency due to the background cosmic ray flux which temporarily renders some pixels useless. The calibration product is a coefficient (per CCD) which gives the fractional area lost (or "deadened") per second. Since the area lost increases with time, the magnitude of the effect depends upon the ACIS clocking parameters (number of rows, window location, frame−time) which determine how long a pixel was exposed to the background cosmic ray flux during the primary exposure and during electronic readout from the frame−store area. For full−frame timed−exposure, the dead area is about 4% at maximum CHIPY and about 2% at the readout. It is smaller for subarrays.

The ACIS clocking parameters required to scale the coefficients in the dafile are contained in the observation−specific parameter block file, which can be set by the associated parameter of this took, "pbkfile". If dafile=NONE, then pbkfile=NONE is assumed.

The dafile parameter is ignored for HRC data.

Parameter=ardlibparfile `(string)`

The name of the parameter file to be used by ARDLIB. Typically, "ardlib.par" will be used.

The parameter file contains only one explicit CALDB parameter (osip). Other such parameters are contained in ardlib.par. This parameter file is selectable using the `ardlibparfile' parameter. Calibration files are specified implicitely via the `DetSubsys' and `Mirror' parameter described below. Most default to CALDB, but can be made explicit for customized use.

Parameter=geompar `(string default=geom)`

The name of the Pixlib Geometry parameter file.

Parameter=verbose `(integer)`

The verbosity level for messages.

Parameter=clobber `(boolean)`

If set to yes, existing output files will be removed.

CHANGES IN CIAO 3.4

Parameter Changes

Two new hidden parameters have been added to mkgarf to allow the ACIS dead area correction to be taken into account: "pbkfile" and "dafile". By default, the correction is "off", i.e. both parameters are set to "NONE". See the individual parameter descriptions for more information.

CHANGES IN CIAO 3.0

The ACIS contamination functions is automatically included via a CALDB file pointed to by ardlib.par. This mitigates the need for the interim script, 'contamarf'. The application of the function may be turned off via an ardlib qualifier (see the EXAMPLES).

Notes

The command "mkgarf −−version" prints the version of mkarf and the associated libraries. This is useful when reporting problems with the tool.

Bugs

See the bugs page for this tool on the CIAO website for an up−to−date listing of known bugs.

See Also

calibration
    ardlib
dm
    dmbinning
tools

acis_bkgrnd_lookup, acis_fef_lookup, acis_set_ardlib, acisspec, add_grating_orders, add_grating_spectra, asphist, dither_region, dmarfadd, dmcopy, dmextract, dmfilth, dmgroup, dmimghist, dmregrid, fullgarf, get_sky_limits, mkacisrmf, mkarf, mkexpmap, mkgrmf, mkinstmap, mkpsf, mkrmf, mkwarf, psextract, psf_project_ray, rmfimg, specextract, tgextract

---

See Also