Chandra X-Ray Center

Transmission Grating Spectroscopy Data Processing Tool and File Specifications

Level 1.5: TG/ACIS Order-Sorting

David Huenemoerder

(`http://space.mit.edu/ASC/docs/TBD`)

7 April 2000

1 Purpose

Given the real CCD performance with gain and CTI degradation, resulting in strong spatial variation in the redistribution function peak width, an improved order-sorting algorith is required which accounts for this spatial dependence. This algorith is needed in `tg_resolve_events`, and a new tool is needed to make the "`rmfile`" from FEF and gain files. The table is also required as input by `mkgarf` to compute an accurate effective area for HETG/ACIS-S.

Here I will define a new table, indicate how the `tg_resolve_events` algorithm must be modified, and outline how the table can be generated. In addition, I will provide a prototype table for test purposes containing the old data. Finally, a file-format specification will be given, which will become part of an ICD.

The section contents are organized as follows:

2 Current implementation overview.

3 Required enhancements.

4 New tools needed.

5 File specification documents.

6 File format specification.

7 Sketch of new algorithms.

2 Current implementation

`tg_resolve_events`: apply the spatial mask, apply aspect, assign diffraction coordinates.

If the detector is ACIS, the CCD pulse-height is used to discriminate diffraction orders, and m and λ are added to the event record. If a photon lies in the diffracted spatial region but does not match a valid CCD PHA, it is labeled as background. To do ACIS-PHA order sorting, a special table (the reference data file) is used which is derived from the response matrices for each CCD. The table gives the low and high CCD energy (scaled PI) bounds at each energy and the fractional enclosed probability.

The lookup table used by `tg_resolve_events` (its "`rmfile`" parameter) contains four columns: `ENERGY`, `ENERGY_LO`, `ENERGY_HI`, and `FRACPROB`.

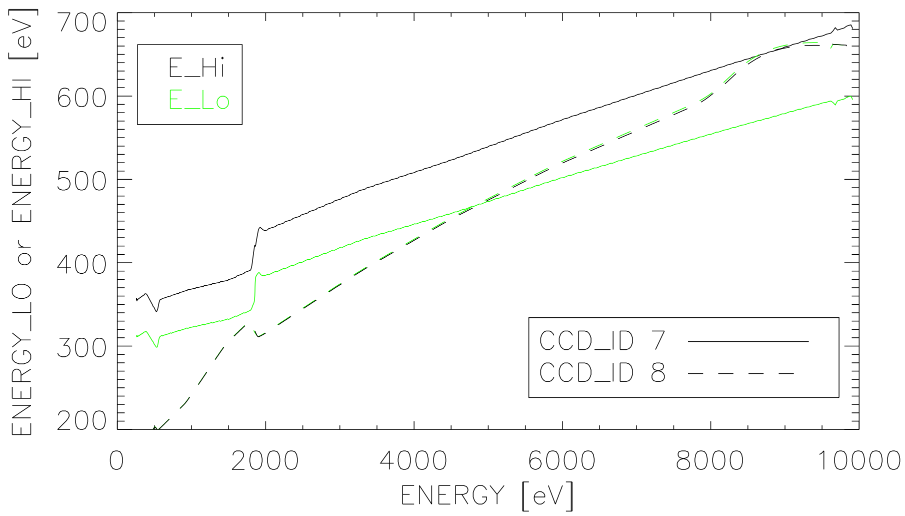

Figure 1: *This graph shows the* `ENERGY_LO` *and* `ENERGY_HI` *columns from the order-sorting table as a function of energy. Note that these represent the* difference *between the peak and the order-sorting limits. The limits are chosen by fitting a Gaussian to the* RMF *main peak, then finding limits specified in factors of σ, the dispersion of the peak. Hence, as the* RMF *profile changes, the enclosed probability in those fixed limits can change.*

There is a separate table for each CCD formatted as a FITS binary table extension, *with a single relation for the entire* CCD. The low and high energy columns are actually the *differences* in energy from the center, such that the fractional power enclosed between E$NERGY - ENERGY_LO$ to ENERGY + ENERGY_HI is FRACPROB. Figure 1 shows an example of

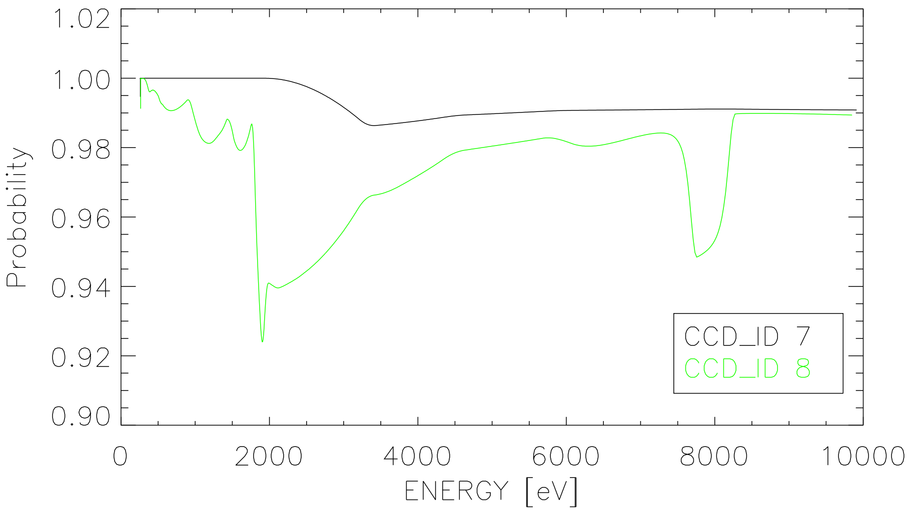

Figure 2: *This graph shows the* `FRACPROB` *column from the order-sorting table as a function of energy for two* CCDs*, for* $\pm 7\sigma$ *limits about the main peak. The sharp dip by about 8% just below 2 keV is due to the appearance of the silicon "escape peak". The dip at 8 keV is from an artifact in the response matrix.*

the low and high energy limits as a function of energy. Figure 2 shows the enclosed power versus energy.

For each photon, the grating wavelength is computed for a given order, m (`TG_MLAM`$/m$), and the CCD "wavelength" (from the event `ENERGY` column, which is a linearized PHA) is compared against the limits at the grating energy. If it falls between them, the event is assigned to order m. Figure 3 shows an example.

The current tables were constructed from pre-degradation mean PI response matrices. Default processing uses rather wide regions to avoid severe truncation of the effective area, at the possible expense of additional background.

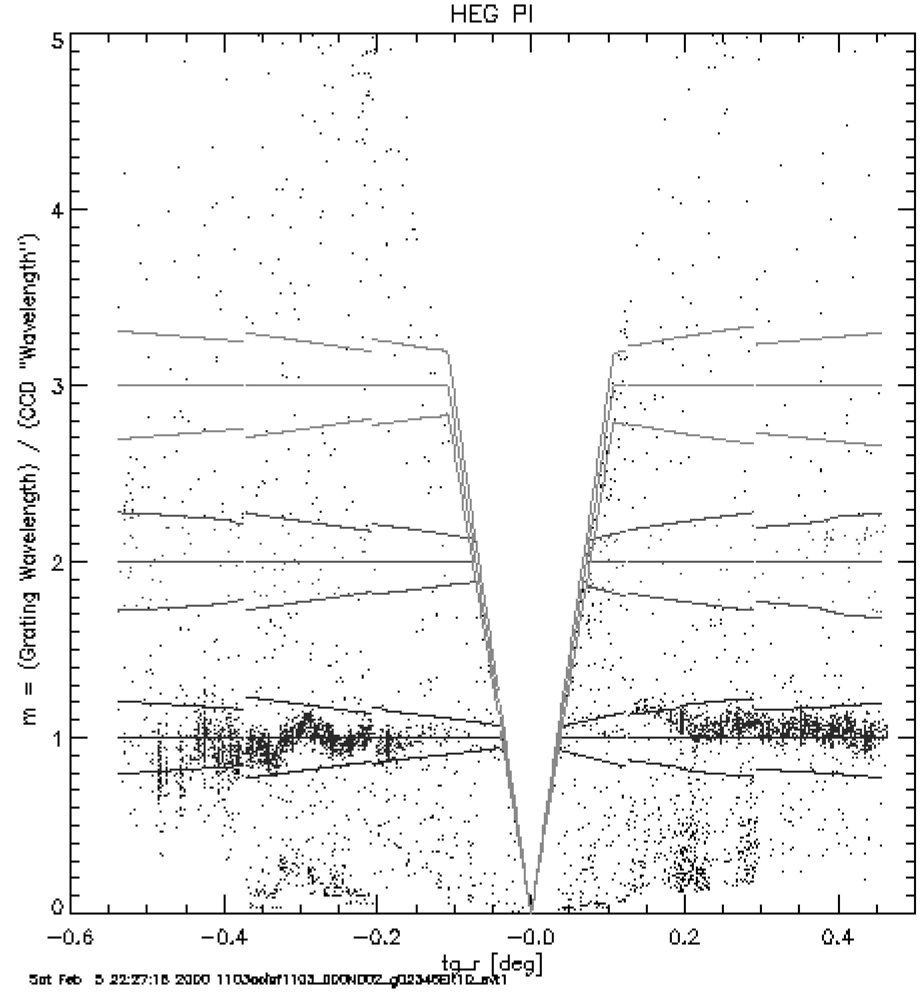

Figure 3: *This graph shows the limits of the order-sorting table vs diffraction angle as solid lines, in terms of the ratio of grating wavelength to* Ccd *"wavelength". The points plotted are from a Capella observation. Note the shifts and wiggles in the events' distribution. These will be suppressed if gain and* Cti *spatial dependence is incorporated into* `tg_resolve_events`, *and the boundaries can be made smaller to reduce background.*

3 Required enhancement

Since the CCD resolution depends strongly upon position (primarily `CHIPY`, but also `CHIPX`), and since the HETGS spectrum spans a large range in `CHIPY`, extra dimensionality is required in the lookup-table. For each event, the energy limits must be looked up *for the specific location on the chip.*

The new file definition closely follows the ACIS gain file format, and is also very similar to the MARX CCD gain/resolution table. For convenience, all CCDs are enclosed in one FITS extension instead of ten, and the CCD_ID is given by a column. The region of the CCD is also given as a set of columns specifying the pixel boundaries. And like the PHA-energy mapping of the gain table, the energy, energy limits, and probability arrays will be given as array columns. One significant difference from the old table is that *the energy limits are now given absolutely, instead of as differences from the peak.* This (and the slight column name change) is to more closely conform to the existing FITS structure of the `EBOUNDS` table. The specific file definition is given in a following section.

4 New Tools

A new tool will be needed to make the new IRM (Integrated Response Matrix) table. That tool should take as input the FEF files and the gain-table. For each cell of each CCD:

– read the $N_{Hi,ccd} \times \sigma$, $M_{Lo,ccd} \times \sigma$ parameters (one pair for each CCD)
– identify the coefficients of the main peak in the FEF;
– read the PHA and σ of the main peak;
– read the gain-table for the same cell from the gain-table file;
– convert the peak position and dispersion to energy;
– apply the $N_{Hi,ccd} \times \sigma$, $M_{Lo,ccd} \times \sigma$ limits to determine the absolute limits (`ENERG_LO, ENERG_HI`);
– evaluate the FEF total area, the main peak area within the given limits, and divide to get the fractional area (`FRACPROB`).

5 File Formats

Input files (Analysis Reference Data) are specified in detail by the following Interface Control Document:

- "Analysis Reference Data for: Exposure Maps and Grating Levels 1.5–2 Tools"

Interface Control Documents are available at:
`http://hea-www.harvard.edu/asclocal/sds/development/ICD/index.html`.
This document will be updated with the following specification:

6 New ICD Material: RMF Energy-Limit Table

Data Product Summary

Analysis Reference Data Product	IRM Table
Instrument(s)	HETG/ACIS-S, LETG/ACIS-S
Level	1.5
Scientist/SDS	D.Huenemoerder
Scientist/Cal	N.Schulz
Filetype	FITS bintable
Created by tool	mkirmfxy
Used by tool(s)	tg_resolve_events, mkgarf
Sample file	`acisD1999-08-26irmfxy_N0000.fits.gz`

In order to perform order-sorting during extraction of grating spectra detected with ACIS, a pulse-height region (in "pulse invariant" or PI bins, scaled to energy units) is used to specify energy offsets versus energy. The table is generated for a specified region about the main peak of the redistribution function, and specifies the low and high energy for this power as a function of energy It is created using a detector redistribution matrix function ("RMF"), as encoded in FITS Embedded Function (FEF) files.

6.0.1 File Names

The `CALDB` type for the RMF energy width table is `irmfxy` (which is very short for "integrated redistribution matrix function energy limits vs. x, y"). File names will be of the forms:

```
acisD1999-08-26irmfxyN0001.fits
```

6.0.2 File Structure

The following table describes the file structure by Header-Data Unit number, type, extension name, content, and HDU classes. An asterisk (*) denotes the CXC principal HDU.

HDU	HDU Type	EXTNAME	CONTENT	HDUCLASS	Description
0	PRIMARY	N/A	N/A	N/A	NULL
1 (*)	BINTABLE	AXAF_IRMXY	CDB_ACIS_IRMXY	ASCRESPONSECCD	Redistribution function energy limits and enclosed energy vs energy and position.

6.0.3 Column Descriptions

Columns for *RM_** HDUs

TTYPE	TUNIT	TFORM	TLMIN	TLMAX	TNULL	Comment
CCD_ID		1I	4	9	-1	CCD identifier (same as event tag)
CHIPX_MIN	pixel	1I	1	1024	-1	Minimum `CHIPX` pixel for the cell.
CHIPX_MAX	pixel	1I	1	1024	-1	Maximum `CHIPX` pixel for the cell.
CHIPY_MIN	pixel	1I	1	1024	-1	Minimum `CHIPY` pixel for the cell.
CHIPY_MAX	pixel	1I	1	1024	-1	Maximum `CHIPY` pixel for the cell.
ENERGY	eV	nE	0	20000	NaN	Energy array; x-axis for the tabulated `ENERG_LO`, `ENERG_HI`, and `FRACPROB` columns
ENERG_LO	eV	nE	0	20000	NaN	Low boundary energy array, at `ENERGY`, which encloses `FRACPROB`.
ENERG_HI	eV	nE	0	20000	NaN	High boundary energy array, at `ENERGY`, which encloses `FRACPROB`.
FRACPROB		nE	0.0	1.0	NaN	Enclosed energy fraction array; the integrated probability in the redistribution function between `ENERG_LO` and `ENERG_HI` as a function of `ENERGY`.

6.0.4 Special Header Keywords

A useful item of history are the limits used about the center of the redistribution's main peak, in terms of the Gaussian dispersion, σ. The current specification only has one pair per CCD. Since they don't need to be machine readable, they will be encoded in `HISTORY` keywords.

```
HISTORY CCD_IDs:   4   5   6   7   8   9  / CCD list (example)
HISTORY N_sig_Lo: 3.0 4.0 3.0 5.5 4.5 8.0 / Low limit (example)
HISTORY N_sig_Hi: 3.0 2.0 3.0 4.0 3.2 6.0 / High limit (example)
```

6.0.5 CALDB Keywords

The table depends strongly upon focal-plane temperature. This will be stored in a CALDB keyword.

The required CALDB keywords are as follows:

```
CCLS0001  'CPF'
CDTP0001  'DATA'
CCNM0001  'IRMXY'
CBD10001  'ENERG(0.1-10.0)keV'
CBD10001  'FP_TEMP(158.2-168.2)K'
CVSD0001  1999-08-26T00:00:00
CDES0001  'ACIS integrated redistribution matrix limits'
```

6.0.6 Size Estimate

There are nominally 32×32 x, y cells for FI Ccds (Ccd_IDs 4, 6, 8-9), and 4×32 for BI Ccds (5, and 7), for a total of 4352 records. For $n = 128$ energies, there are 2058 bytes per record. The total file size is about 8 MB. There will be one file per Acis FP_TEMP (about 4).

6.1 Unresolved Issues

1. none

7 Algorithms

The tool, `tg_resolve_events` reads the order-sorting table and uses it to assign orders to photons. A schematic of the current algorithm is:

```
For event,
    Read event.CCD_ID
    Read event.TG_MLAM
    Read event.ENERGY
    Read rmfile.ENERGY[CCD_ID]
    Read rmfile.ENERGY_LO[CCD_ID]
    Read rmfile.ENERG_HI[CCD_ID]
    For order -max_order to max_order (except 0)
       Compute  order_estimate == event.tg_mlam / (hc/event.ENERGY)
       Compute  nominal energy: E_0 =
                round(order_estimate)*hc/event.tg_mlam
       Look up  rmfile.energ_lo(E_0), rmfile.energ_hi(E_0)
       Compute  CCD hi,low boundaries:
                 m_lo = order*hc/(energy-rmfile.energy_lo(E_0))
                 m_hi = order*hc/(energy+rmfile.energy_hi(E_0))
       Test order_estimate against boundaries, m_lo, m_hi;
            break if resolved.
     Next(order)
     Write tg_m, tg_lam
Next(event)
```

The new algorithm becomes:

```
For event,
    Read event.CCD_ID
    Read event.TG_MLAM
    Read event.ENERGY
    Read event.CHIPX
    Read event.CHIPY

    NEW- look up arrays by CHIPX,CHIPY, as well as CCD_ID:
      For each event, find cell its chipx, chipy falls in:
          (rmfile.CHIPY_MIN < event.CHIPY <= rmfile.CHIPY_MAX )
            && (event.CCD_ID == rmfile.CHIP_ID)

    Ditto for event.CHIPX

    Read rmfile.ENERGY[CCD_ID][CHIPX][CHIPY]
    Read rmfile.ENERG_LO[CCD_ID][CHIPX][CHIPY]
    Read rmfile.ENERG_HI[CCD_ID][CHIPX][CHIPY]
```

```
      (...rest is nearly identical to before...)

      For order -max_order to max_order (except 0)
         Compute  order_estimate == event.tg_mlam / (hc/event.ENERGY)
         Compute  nominal energy: E_0 =
                  round(order_estimate)*hc/event.tg_mlam
         Look up  rmfile.energ_lo(E_0), rmfile.energ_hi(E_0)
         Compute  CCD hi,low boundaries:
                   m_lo = order*hc/rmfile.energ_lo(E_0)        /*diff*/
                   m_hi = order*hc/rmfile.energ_hi(E_0)        /*diff*/
         Test order_estimate against boundaries, m_lo, m_hi;
              break if resolved.
       Next(order)
       Write tg_m, tg_lam
  Next(event)
```

In the tool, `mkgarf`, the table must be read to compute the effective area for each detector position along the spectra:

```
      Select grating, order, zero-order position;
      Read energy grid;
      For each energy bin,
        project to detector, using aspect and zero-order sky pos
        determine chip_id, chipx, chipy
        calculate efficiency
        lookup up rmfile.FRACPROB by CHIP_ID, CHIPX, CHIPY, ENERGY
        scale efficiency by rmfile.FRACPROB
      next energy bin
```