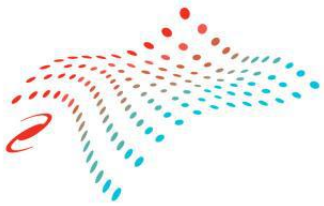

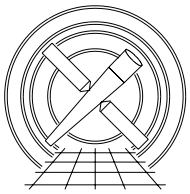

MIT Kavli Institute **Chandra X-Ray Center**

MEMORANDUM
March 27, 2022

To:	Jonathan McDowell, SDS Group Leader	(_____)
	Dale Graessle, DS Calibration Data Coordinator	(_____)
From:	David P. Huenemoerder, SDS	(initial / date)
Subject:	Interface Control Document: ACIS/Grating Order-Sorting Reference Data & Algorithm	
Revision:	1.0 (March 2022)	
URL:	`https://space.mit.edu/cxc/docs/osip_data_alg_icd-1.0.pdf`	
File:	`osip_data_alg_icd-1.0.tex`	

1 Overview of Order-Sorting

This memo will define the `CALDB` files and their generation algorithm for order-sorting with the *Chandra* gratings used in conjunction with the ACIS detectors. The OSIP tabulates `ENERG_LO` and `ENERG_HI` vs `ENERGY` and the enclosed energy fraction within those limits, `FRACRESP`, for each detector element and location. These values are used

- by `tg_resolve_events` to assign events to orders by determining whether the CCD energy falls between the low and high limits;
- by `mkgarf` to apply the `FRACRESP` factor to the effective area for each position in the spectrum.

To briefly illustrate the need for order sorting, we can look at the events in a few different coordinates and relate them to the simple diffraction equation (for the small angles appropriate here): $m\lambda = P\theta$, where m is the integral diffraction order, λ is the wavelemgth, P the grating period, and θ is the diffraction angle. Figure 1 shows a typical distribution of events on the ACIS-S detector array for an observation using HETG. Since the HETG has two types of gratings (high and medium energy gratings, called HEG and MEG, respectively) which disperse at different clocking angles, the dispersed photon events form a shallow "X". Each or HEG and MEG disperse to both positive and negative orders; they also have an un-diffracted portion in a common zeroth order. The diffraction angle for each arm is measured along the dispersion with $\lambda = 0$ being the zeroth order position. Using the known detector geometry, grating parameters, and aspect solution, we can convert the detector location for each event to an $m\lambda$ coordinate very accurately (< 0.01 Å). However, we would like to "resolve the orders" to know m and λ independently. To do that, we use the CCD energy resolution.

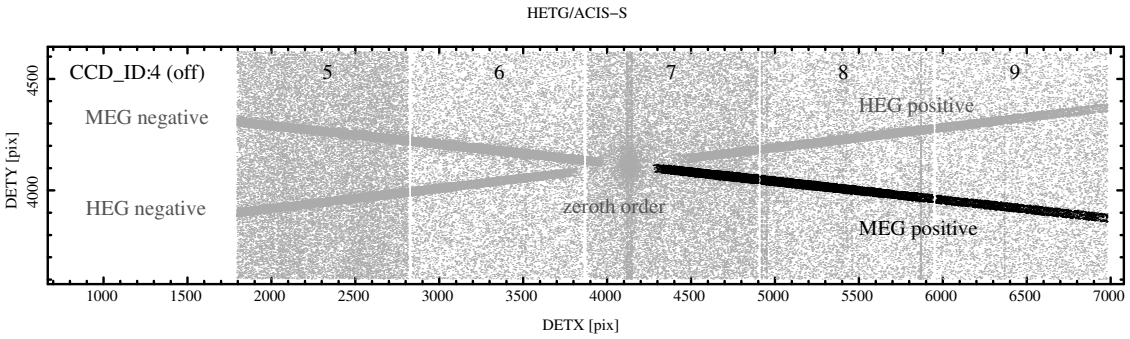

Figure 1: The spatial distribution of events for an HETGS observation (ID 1451, the star II Pegasi). We have used detector coordinates so that the dithered events are spread out so it is easier to see the spectral regions, as labeled. We have highlighted the MEG positive order events in darker coloring; these are used in Figure 2.

Figure 2 shows the CCD instrumental energy (determined from the scaled "pulse height" of each event) against the $m\lambda$ coordinate as determined for just the MEG positive order arm's events (those shown as dark points in Figure 1). Here we can now see distinct bands, with higher CCD energy near zeroth order. The densest band is for positive first order, and we can easily see second and third above it (the lowest faint strip below 7 Å is from silicon fluorescence in the detector). The event density on the vertical axis at any x-value is given by the probability that the CCD redistributes an input photon with wavelength λ to any particular CCD instrumental energy.

We can provide a real-valued estimator of the order by dividing the very accurately known grating coordinate, $m\lambda$, by the CCD instrumental "wavelength". That is, $\tilde{m} = (m\lambda)_{\rm g}/\lambda_{\rm CCD}$. Here we use the inverse relationship between energy and wavelength to define the "CCD instrumental wavelength" as $\lambda_{\rm CCD} = hc/E_{\rm CCD}$).[1]

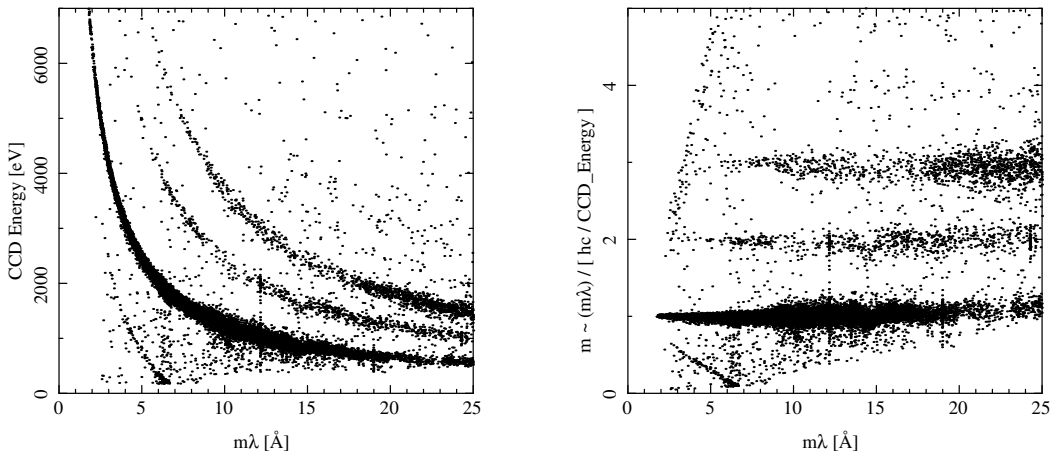

Figure 2: Event distributions for MEG positive orders in CCD instrumetal energy (left) and the same events rescaled to a real-valued order estimate (right), both against the grating scale, $m\lambda$.

The right panel of Figure 2 shows the real-valued order against the grating $m\lambda$ coordinate. The rest of

[1]For wavelength in Å and energy in keV the constant of proportionality is $hc \sim 12.3984185$.

this document decribes how we can use the CCD redistribution probability calibration to define the order boundaries used to assign the integral order values for each event (Order Sorting), and how to determine the integrated fraction of the CCD probability distribution included in that region (Integrated Probability; "OSIP").

1.1 Computation of the OSIP

OSIP values are computed from the CCD response vs energy for every calibration cell of every CCD. The CCD response used is after all corrections (gain, CTI) — it is effectively determined from response matrices (RMF) as computed by `mkacisrmf`. There are approximately 1000 calibration cells per CCD (nominally on a 32×32 pixel grid), and 6 CCDs in the ACIS-S array. Computation for ACIS-I (4 CCDs) is lower priority, but there are some observations using HETG or LETG with ACIS-I. There are also multiple epochs of distinct CCD calibrations, primarily due to different operating temperatures.

There are two modes of OSIP generation to support:

Normal Mode: We usually contstruct the OSIP for a given minimum `FRACRESP` goal. For each cell, we compute the `ENERG_LO` and `ENERG_HI` limits vs energy by requiring a minimum specified `FRACRESP`, such as 98.5% enclosed energy.

Custom Mode: Alternatively, we can compute `FRACRESP` from *a priori* given `ENERG_LO` and `ENERG_HI` limits (vs energy). This mode can be useful for custom analysis when there are either confusing sources to avoid, or if there excess background in some regions of energy-dispersion space.

1.2 Computation Issues

Brute-force computation of 6000 ACIS-S RMFs with `mkacisrmf` (to provide a database of CCD response vs energy vs `CCD_ID`, `CHIP_X`, and `CHIP_Y`) would take about 5 hours on a typical workstation. Storage of 6000 RMFs would require about 144 GB (though computation could be done serially by region—all RMFs are not required simultaneously). This would also have to be done for any epoch or temperature-dependence in responses resulting in significantly different CCD resolution.

As a user tool, one could consider computing the OSIP for only the region covered by the spectrum. The standard HETG spatial mask, when dithered, covers about 25% of the detector area, so does not represent a large saving in time. Such use would also make an observation- and source-dependent file, and limit applicability.

Hence, `mkosip` is likely to be a CXC in-house tool to make the tables for the `CALDB`. The mode in which a custom order-sorting region is required would be used for rare special-case user analysis, and then the computation time will not be a severe impact.

2 OSIP File Format Specification

The OSIP files will follow standard CXC calibration FITS file standards. Here we define the current as-implemented format.

2.1 File Names

The `CALDB` type is `osip`. File names will be of the form

`acisD`*yyyy-mm-dd*`osipN`*nnnn*`.fits`

in which $nnnn$ is a 4-digit version counter (which starts at `0001`, and the date-stamp is to be filled in with the start date of file applicability).

2.2 HDU Components

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

HDU	HDU	EXTNAME	EXTVER	CONTENT	HDUCLASS	Description
0	PRIMARY	N/A	N/A	N/A	N/A	NULL
1 (*)	BINTABLE	AXAF_OSIP	1	CDB_ACIS_OSIP	ASC RESPONSE CCD	Binary table extension listing order-sorting energy bounds and enclosed energy fraction vs CCD and calibration region.

2.3 Columns and Coordinate Systems

Columns for OSIP are given in the following table. Column ordering in the FITS file is arbitrary.

TTYPE	TUNIT	TFORM	TLMIN	TLMAX	Description
`CCD_ID`	1	1I	0	9	Numeric index for each CCD in the ACIS-I and ACIS-S arrays.
`CHIP_XMIN`	pixel	1I	1	1024	spatial bin low edges
`CHIP_XMAX`	pixel	1I	1	1024	spatial bin high edges
`CHIP_YMIN`	pixel	1I	1	1024	spatial bin low edges
`CHIP_YMAX`	pixel	1I	1	1024	spatial bin high edges
`NPOINTS`	1	1I	1	32768	Number of valid bins in the `ENERGY`, `ENERG_LO`, `ENERG_HI`, and `FRACRESP` columns.
`ENERGY`	eV	n_gE	0	20000	Energy grid, for the true photon energy. The column is a fixed-length array whose length is the maximum length required for any energy, but only contains valid data in bins 1 to `NPOINTS`. n_g is the maximum array length required.
`ENERG_LO`	eV	n_gE	100.0	12000.0	Low energy limit of instrumental (CCD) energy.
`ENERG_HI`	eV	n_gE	100.0	12000.0	High energy limit of instrumental (CCD) energy.
`FRACRESP`	1	n_gE	0.0	1.0	Fractional enclosed energy between `ENERG_LO` and `ENERG_HI`.

Grids are required to be monotonic and in ascending order but are not necessarily uniform.

2.3.1 Relevant Header Keywords

Relevant keywords are:

`TELESCOP`: The only allowed value is `CHANDRA`.

INSTRUME: The only allowed value is ACIS.

DETNAM: ACIS-s, where is is a numeric string listing the relevant CCD_ID values (e.g., ACIS-456789) for the ACIS-S array.

2.3.2 CALDB Keywords

The required CALDB keywords are as follows:

```
CCLS  CPF
CDTP  DATA
CCNM  OSIP
CDES  Order-sorting and integrated probability
CBO1  ENERG(100.0-12000.0)eV
CVSD  2000-01-29T20:00:00*
CVST  20:00:00*
CBD   CTI_APP(PPPPPPBPBPP)*
CBD   FP_TEMP(151.16-164.16)K*
```

* These are representative values to be replaced as necessary to describe specific data.

2.4 Size and File Number Estimates

There will be one OSIP file per ACIS response epoch, as defined by the number of "p2_resp" files, currently about 5. Each OSIP file, with an optimized energy grid of about 100 bins will be of order 5 MB. (Current files are about 20 MB each, larger than the above estimate due to denser energy grids.)

3 OSIP Computation Algorithm Specification: "mkosip"

3.1 Parameters

osip_mode: (required) compute_limits (default) or compute_fracresp .

For compute_limits, we use the fracresp_goal and compute the corresponding energy bounds (ENERG_LO, ENERG_HI)

For compute_fracresp, we read an order_bounds_table and compute the FRACRESP.

obspar: (optional) observational parameters, which includes the date and instrumental configuration required to resolve to a specific CALDB file. Ignored if a named p2_resp file is given by the p2resp_file parameter.

p2resp_file: (optional) explicit CALDB file of type "p2_resp". Ignored if the epoch or OBSPAR file is specified

fracresp_goal: (optional; default 0.985 (TBR)) If osip_mode is compute_fracresp, try to reach or exceed this value of the enclosed energy fraction at each energy. It might not be feasible to reach this goal, in which case the clipping parameters (see below) will be imposed.

This parameter is ignored if the mode is `compute_limits` and an `order_bounds_table` is given.

`order_bounds_table`: (optional) A table of order limits vs energy. That is, `ENERG_LO`/`ENERGY` and `ENERG_HI`/`ENERGY` (required if `osip_mode` is `compute_fracresp`). (Note: a CCD "detected energy" divided by the first-order equivalent photon energy (i.e., from the high-resolution diffracted position) is equivalent to a real-valued diffraction order.)

`energy_grid`: (optional) default to internally defined grid. If specified, filename of custom table of energies (single column). The default grid is comprised of 60 (TBR) energies with variable spacing, designed to have finer grids over regions of rapidly changing spectral response. Computation time will scale with the number of energy grid bins.

`ccdids`: (optional; default ACIS-456789) list of CCDs to process. CCDs are identified by integers from 0–9.

`region`: (optional; default to entire area) Regions of CCDs (in chip coordinates) over which to compute the OSIP. Region specification must include `CCD_ID`, `CHIP_X`, and `CHIP_Y`. Resolution will only be at that in the `p2_resp` file.

`high_limit_clip`: (optional; default = 1.3) maximum allowed ratio of `ENERG_HI`/`ENERGY`. This prevents the order-sorting region from becoming too wide, at the expense of possibly not meeting the `fracresp_goal`.

`low_limit_clip`: (optional; default = 0.8) minimum allowed ratio of `ENERG_LO`/`ENERGY`. This prevents the order-sorting region from becoming too wide, at the expense of possibly not meeting the `fracresp_goal`.

`fracresp_high`: (optional; default = `fracresp_goal`): Allow the initial high-side fractional enclosed energy to be higher than the general goal set by `fracresp_goal`. This is used to set the high limit as if the response were a symmetric gaussian. The low limit then follows until `fracresp_goal` is reached, as long as the limit is greater than the `low_limit_clip` value. This is to help obtain a higher `FRACRESP` by extending the limit on the Gaussian-like high energy side of the CCD response.

3.2 Graphical Examples

In order to guide interpretation of the specifications below, it is useful to have some graphical representation of the OSIP as well as the CCD response. Figure 3 shows the CCD response functions for ACIS-S4 (`CCD_ID` 8) at one location for 3 photon energies. Figure 4 shows detail for the CCD response at one energy as well as the end result of an OSIP evaluation over input photon energy for that same calibration cell.

3.3 Algorithm Overview

1. Form an input photon energy (E_p) grid.

   The grid is sparsely spaced where spectral changes are smooth, and denser where there are strong features in the response (such as the silicon edge near $2\,\mathrm{keV}$).

   The default grid is given by a piecewise evenly spaced energy grid whose segments are defined in units of eV by:

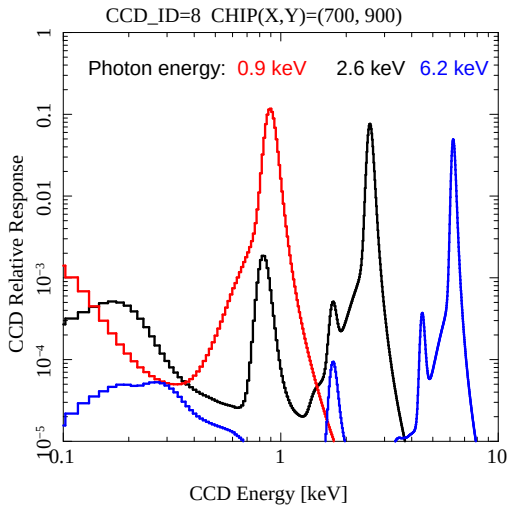

Figure 3: This graph shows the relative responses (redistribution functions) of one CCD at one chip location for three input photon energies. The main peak position follows the input photon energy (which is what is used to calibrate the gain and define the "CCD energy" scale on the x-axis). The main peak is near-Gaussian in shape, but the long tail and its energy dependent structure is clear.

E_{min}	E_{max}	#bins
253	1740	20
1750	1890	10
1900	4990	20
5000	12000	10

2. For each CCD calibration cell, determine the CCD response (e.g., RMF redistribution function) for each input photon energy grid point, E_p.

   Note: A CCD response for a mono-energetic source is represented as a probability vs "energy", where that "energy" scale is assigned to the channel (PHA or PI) and is determined by the position of the peak vs photon energy (the CCD "gain"). We'll call this scale the "CCD energy", E_c, to distinguish from the input photon energy, E_p, if context is not clear.

   The response is represented as a matrix, the "RMF" which encodes the redistribution function from input energy on one axis to the output "energy" (or channel) on the other axis. The response's calibration cells are defined in the `p2_resp` files in FITS table blocks `AXAF_CTI_SCATTER_L`, `AXAF_CTI_SCATTER_G` columns `CCD_ID`, `CHIPX_LO`, `CHIPX_HI`, `CHIPY_LO`, and `CHIPY_HI`.

3. Given the CCD redistribution curve (probability vs CCD energy) for a given photon energy E_p, evaluate the high and low CCD energy limits which give the desired enclosed fraction of the probability (`FRACRESP`).

There is no unique solution for high and low energy limits which give the desired (or greater) value for `FRACRESP`. The CCD redistribution function is asymmetric, having a long low-energy tail, so we cannot necessarily have low and high limits placed symmetrically about the peak. Hence, we will make some assumptions about the distribution and impose some constraints on the limits:

- the high energy side of the distribution is nearly gaussian;
- the low energy side has long, structured tail;

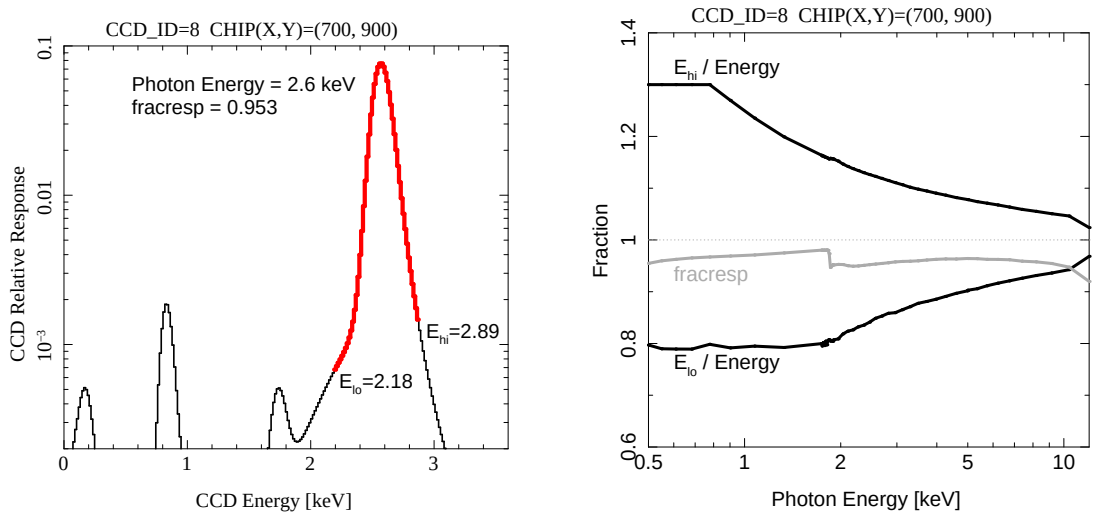

Figure 4: The left panel shows the CCD response function for CCD 8 at one location for an input photon energy of 2.6 keV, where the low energy tail is significant. The the OSIP region determined for the FRACRESP goal of 0.9995 is shown in red, but due to the tails and the imposed clipping limits, the actual FRACRESP obtained is 0.953. The right panel shows both the FRACRESP (gray curve) and the scaled ENERG_LO and ENERG_HI limits for a FRACRESP goal of 0.9995. The imposed clipping limits are apparent at low energy at values of 0.8 and 1.3.

- there are hard low and high energy limits we will not exceed, since they would then cause the order regions to be too wide (and possibly overlap, or allow too much background).

Given a goal for the enclosed energy fraction (FRACRESP in the OSIP file), we will first assume that the distribution near the peak is gaussian and we will evaluate the corresponding high energy limit for the desired enclosed fraction for this shape. Then given that high side limit, we will then determine the low energy limit which meets or exceeds the desired FRACRESP.

3.4 Algorithm Details: How to Determine the CCD Energy Limits for a Given Photon Energy

1. Impose general constraints on the CCD ENERG_LO, ENERG_HI:

   Define m as the ratio of E_c/E_p; where E_c will be ENERG_HI or ENERG_LO, and E_p is our photon energy grid value.

   Impose hard constraints on the limits:

   $m_{min} = 0.5$; hard limit for minimum $m \geq$ ENERG_LO/ENERGY

   $m_{max} = 1.5$; hard limit for maximum $m \leq$ ENERG_HI/ENERGY

   m should never be outside these limits. If it is, the limit should be clipped and a warning issued.

   Control parameters will generally restrict these further, to, say, 0.8 and 1.3, via the low_limit_clip and high_limit_clip values (but need not have a warning when clipped).

   Define minimum limits (that is, the region between ENERG_HI and ENERG_LO should not be too narrow):

$m_{min\_hi} = 0.90$; m should be $\leq$ this value on the low side.

$m_{max\_lo} = 1.10$; $m \geq$ this value on the high side.

2. Form cumulative distribution functions (CDF) of the CCD response function, $R(E_c)$, over E_c (for the assumed E_p) and compute some statitistics of the distribution.

   Assuming E_c grid represents the low side of the bin values, create the high and central bin values, with some over-sampling to help with numerics.

   Make a linear energy grid, E_1, with 8 times the number of bins in the native response, $R(E_c)$, array, and make the corresponding high-bounds grid, E_2. The CCD response nominally has 1024 bins.[2]

   Make an array of the bin size: $\delta = E_2 - E_1$.

   Make an array of the bin center: $E_0 = (E_2 + E_1)/2.0$.

   Interpolate the reponse to this new grid, $R(E_c) \rightarrow R(E_0)$.

   Normalize the distribution: $R_n = R/\delta/\Sigma R_n$;

   Compute the cumulative distribution function:

   $CDF = \mathrm{cumsum}(R_n \times \delta)$

3. Determine some statistics of the redistribution profile:

   $\mathrm{mode}(R_n)$: the value of E_0 where R_n is a maximum:

   $E_{mode} = E_0[\mathrm{where}(R_n = \max(Rn))]$ (Ideally this should be the same as E_p, the input photon energy.)

   For the region about the mode (where the shape is fairly Gaussian), approximate the mean, standard deviation, and skewness of the peak. Limit region and renormalize:

   Over the range $x = E_{mode} - 0.25$ to $E_{mode} + 0.25$

   Re-normalize the limited distribution: $y = R_n(x)/\Sigma R_n(x)$

   Compute moments:

   $x_{mean} = \Sigma(y \times x)$

   $\sigma_{R_n} = \sqrt{\Sigma(y \times (x - x_{mean})^2)}$, the characteristic Gaussian width;

   $S_{Rn} = \Sigma(y \times ((x - x_{mean})/\sigma_{R_n})^3)$, the skewness.

   The most important quantity is the standard deviation (or σ) for the Gaussian-like peak.

   The skewness value could be used to assess how similar to a Gaussian the peak is (e.g, $\mathrm{skew} < -2.5$ if the low energy tail is significant), but we don't implement that in any of the following. It can be kept for future use, if desired.

4. Given the CDF and the statistics, evaluate the one-sided cumulative distribution functions centered on the mode:

   $CDF_{lo} = CDF(R_n; x_{mode} - E_0)$, for $E_0 \leq x_{mode}$;

[2] CIAO ahelp on `mkacisrmf` says: "Bugs: Non-standard channel grids: The mkacisrmf tool does not work correctly with arbitrary, non-standard channel grids. For chantype=PI the channel parameter should always be channel=1:1024:1, for chantype=PHA the channel parameter should always be channel=1:4096:1." So we cannot construct RMFs with higher sampling.

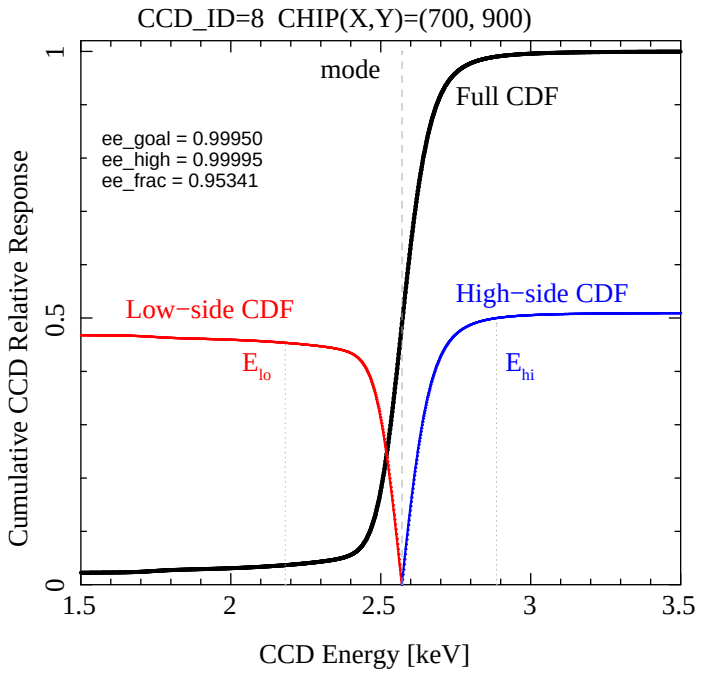

Figure 5: We show the cumulative distribution functions (CDF) for the CCD response for CCD 8 for one calibration cell. The black curve shows the full CDF, while the blue and red curves show the high- and low-side of the mode $CDFs$. The CCD energy limit on the high side, E_{hi} is where a single Gaussian profile would reach the given `ee_high` value. The actual fraction of the response enclosed there is around 0.50. The goal was `0.99950`, but that cannot be reached with a reasonable E_{lo}, which instead resulted in value of about 2.2 keV for a total `FRACRESP` ~ 0.95.

$$CDF_{hi} = CDF(R_n; E_0 - x_{mode}), \text{ for } E_0 \geq x_{mode}.$$

These can be formed from the full CDF via:

$$CDF_{hi}(x - x_{mode}) = CDF(x; x \geq x_{mode}) - CDF(x = x_{mode})$$
$$CDF_{lo}(x_{mode} - x) = CDF(x = x_{mode}) - CDF(x_{mode} - x; x \leq x_{mode});$$

Figure 5 shows and example of the full CDF and the one-sided CDF terms (but on the E_c scale, not scales relative to x_{mode}).

5. Find the high-side limit, in terms of a Gaussian profile with enclosed energy fraction equal to the value of `fracresp_high`. The limit can be found by determining the argument of the error function, $\mathrm{erf}(x)$, which gives the desired `fracresp_high`, as if we had a single Gaussian response. For instance, for a 1.6σ limit, the fractional area of the Gaussian is $\mathrm{erf}(1.6/\sqrt{2}) = 0.890$. Hence, the desired limit in Gaussian sigmas is the inverse error function of:

   $$n_\sigma = \sqrt{2} \times \mathrm{ierf}(\texttt{fracresp\_high}).$$

   Thus

   $$\texttt{ENERG\_HI} = n_\sigma \sigma_{Rn} + E_p.$$

   But we clip by our imposed maximum:

   $$\texttt{ENERG\_HI} = \min(\texttt{ENERG\_HI}, E_p \times \texttt{high\_limit\_clip})$$

6. Find the low side limit such that satisfies the `fracresp_goal`.

   The actual energy enclosed on the high side is

   $$f_1 = CDF_{hi}(\texttt{ENERG\_HI} - x_{mode})$$

   Thus we want to find the value of `ENERG_LO` such that

$\texttt{fracresp\_goal} = f_1 + CDF_{lo}(x_{mode} - \texttt{ENERG\_LO})$

Compare ENERG_LO to the clip limit:

$\texttt{ENERG\_LO} = \max(\texttt{ENERG\_LO}, E_p \times \texttt{low\_limit\_clip})$

7. Also check ENERG_LO and ENERG_HI against the minimum width allowed:

   $\texttt{ENERG\_HI} = \max(\texttt{ENERG\_HI}, m_{min\_hi} \times E_p)$

   $\texttt{ENERG\_LO} = \min(\texttt{ENERG\_LO}, m_{max\_hi} \times E_p)$

8. Finally, re-evaluate the FRACRESP for the given limits (in case clipping altered them):

   $\texttt{FRACRESP} = CDF_{lo}(x_{mode} - \texttt{ENERG\_LO}) + CDF_{hi}(x_{mode} + \texttt{ENERG\_HI})$

9. Save the results for this CCD_ID, CCD cell (and any other relevant observational configuration): ENERGY ($= E_p$ grid point), ENERG_LO, ENERG_HI, and FRACRESP.

3.4.1 compute_fracresp Mode

The compute_fracresp mode follows a similar path right to the end (through Step 4) where instead of determining the limits from the CDF, one takes the specified limits provided in a file referenced by a parameter and evaluates the corresponding FRACRESP (Step 8).