Apply/Remove ACIS PHA Randomization

CIAO 3.4 Science Threads

Table of Contents

Apply/Remove ACIS PHA Randomization

CIAO 3.4 Science Threads

Overview

Last Update: 21 May 2007 − need to set `stop=none` if aspect solution is not provided

Synopsis:

For some observations, the way PHA channels are converted to PI channels will cause an aliasing problem in the final binned spectra. Randomizing the PHA distribution alleviates this effects; see the Background Information section for details. Similarly, one could *remove* the randomization from a file that has already had it applied. Since half an ADU is *much* smaller than the energy resolution at all energies, disabling the PHA randomization should not significantly affect CCD spectra.

Purpose:

To generate a new level=2 event file with PHA randomization applied (or removed). ***Removing the PHA randomization should only be done by advanced users who know exactly what they are doing.***

Read this thread if:

you are working with an ACIS observation and would like to apply or remove the randomization.

Related Links:

- Analysis Guide: ACIS Data Preparation
- Apply the ACIS CTI Correction thread: it is strongly recommended that you reprocess imaging data with this new calibration which became part of standard data processing in DS 6.11. Note that the CTI correction is on by default (`apply_cti=yes`) in the `acis_process_events`.

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

Background Information

For some observations, the way PHA channels are converted to PI channels will cause an aliasing problem in the final binned spectra. Since PHA is in discrete integer values, a single value of PHA corresponds to a single energy and therefore a single PI value for a given 32x32 pixel region in the gainmap file. Although it varies, a difference of one PHA bin corresponds to ~4 eV in energy space. Therefore, in the case of a sufficiently strong source in a small spatial region, the spectrum in energy will show a series of spikes and zeros. Since

```
  PI = 1 + (energy / 14.6)
```

the PI spectrum will also show a strange series of jumps. This problem primarily affects sources with a large number of counts; randomizing the PHA distribution alleviates this effects. PHA randomization became part of standard data processing with ASCDS version DS 6.1 (02 July 2001).

Get Started

Sample ObsID used: 1838 (ACIS−S, G21.5−09)

File types needed: evt1; flt1; bpix1

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.

Check the ASCDSVER keyword in the header, as explained in the Background Information section :

```
unix% dmkeypar acisf01838_000N001_evt1.fits ASCDSVER echo+
R4CU5UPD8.2
```

Since this data was processed with an ASCDSVER lower than DS 6.1, we need to complete this thread in its entirety in order to randomize the PHA channels.

Related acis_process_events threads

There are other threads that should be considered, since they may affect how `acis_process_events` is run. ***The Create a New Level=2 Event File thread shows how to combine all of these options into a single run of `acis_process_events`.***

- Apply an ACIS Gain Map
- Apply the Time−Dependent ACIS Gain Correction
- Remove Pixel Randomization
- Apply the ACIS CTI Correction

Generate A New Level=1 Event File

Determine the eventdef parameter

The `eventdef` parameter specifies the names and data types of the columns in the output event data file. Four predefined strings are included in the parameter file for acis_process_events:

READMODE	DATAMODE	event mode	eventdef string
TIMED	(V)FAINT	timed exposure (very) faint	stdlev1

TIMED	GRADED	timed exposure graded	grdlev1
CONTINUOUS	CC(33)_FAINT	continuous clocking (3x3) faint	cclev1
CONTINUOUS	CC(33)_GRADED	continuous clocking (3x3) graded	ccgrdlev1

If you are unsure of the event mode of your observation, the information can be found in the `READMODE` and `DATAMODE` values stored in the file header:

```
unix% dmkeypar acisf01838_000N001_evt1.fits READMODE echo+
TIMED

unix% dmkeypar acisf01838_000N001_evt1.fits DATAMODE echo+
FAINT
```

This is a timed exposure faint observation, so the proper `eventdef` parameter is "stdlev1." The full parameter syntax of each eventdef string may be found in plist acis_process_events.

Run acis_process_events

Running this tool with the SDP level=1 event file as the input will produce a *new* level=1 event file that has the latest CALDB applied, meaning that the newest gain map will be picked up. Since the CTI and Time−Dependent Gain corrections are on by default, they will both be applied (when possible).

There is only one parameter, `rand_pha`, that needs to be set specifically for this thread. In this example, we are applying the randomization, but using `rand_pha=no` will equivalently remove it if it had already been applied.

```
unix% punlearn acis_process_events
unix% pset acis_process_events infile=acisf01838_000N001_evt1.fits
unix% pset acis_process_events outfile=acis_1838_new_evt1.fits
unix% pset acis_process_events badpixfile=acis_1838_new_bpix1.fits
unix% pset acis_process_events eventdef=")stdlev1"
unix% pset acis_process_events stop=none
unix% pset acis_process_events rand_pha=yes
unix% acis_process_events
Input event file or stack (acisf01838_000N001_evt1.fits):
Output event file name (acis_1838_new_evt1.fits):
aspect offset file ( NONE | none | <filename>) (NONE):
```

It is important to note the unusual syntax of the `eventdef` parameter; the tool will not access the predefined string if the leading ")" is missing (see example 6 of `ahelp parameter`).

The content of the parameter file may be checked using plist acis_process_events.

You may see a warning about the number of event islands that contain one or more bad pixels:

```
# acis_process_events (CIAO 3.4): The following error occurred 26941
  times: dsAFEBADPCNTERR −− WARNING: Event island contains 1 or more bad pixels.
```

It is explained in this FAQ and may be ignored.

Summary

This thread is now complete; the new level=1 event file is named `acis_1838_new_evt1.fits`. See the final section of the ACIS Gain Map thread to generate a new level=2 event file.

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


#---------------------------------------------------------------------------
#
#  acis_process_events.par- Parameter file for acis_process_events program
#
#---------------------------------------------------------------------------
        infile = acisf01838_000N001_evt1.fits   Input event file or stack
       outfile = acis_1838_new_evt1.fits Output event file name
    acaofffile = NONE             aspect offset file ( NONE | none | <filename>)
    (apply_cti = yes)             Apply CTI adjustment?
  (apply_tgain = yes)             Apply time-dependent gain adjustment?
(alignmentfile = )acaofffile -> NONE) sim/fam alignment file ( NONE | none | <filename>)
      (obsfile = NONE)            obs.par file for output file keywords ( NONE | none | <filename>)
      (geompar = geom)            Parameter file for Pixlib Geometry files
      (logfile = stdout)          debug log file ( STDOUT | stdout | <filename>)
    (gradefile = CALDB)           grade mapping file ( NONE | none | CALDB | <filename>)
     (gainfile = CALDB)           acis gain file ( NONE | none | CALDB | <filename>)
   (badpixfile = acis_1838_new_bpix1.fits) acis bad pixel file ( NONE | none | <filename>)
   (threshfile = CALDB)           split threshold file ( NONE | none | CALDB | <filename>)
      (ctifile = CALDB)           acis CTI file ( NONE | none | CALDB | <filename>)
    (tgainfile = CALDB)           gain adjustment file ( NONE | none | CALDB | <filename>)
     (eventdef = )stdlev1 -> {d:time,s:ccd_id,s:node_id,i:expno,s:chip,s:tdet,f:det,f:sky,s:phas,l:pha,l:ph
f:energy,l:pi,s:fltgrade,s:grade,x:status}) output format definition
   (doevtgrade = yes)             Determine event flight grade?
 (check_vf_pha = no)              Check very faint pixels?
(calc_cc_times = no)              Estimate the times of arrival for CC-mode observation?
        (trail = 0.027)           Trail fraction
     (spthresh = 13)              Default split threshold level (overridden by values in threshfile)
  (time_offset = 0)               Offset to add to event time field to synch w/ fam data
   (docentroid = no)              Determine pixel centroid for coord. conversion?
 (calculate_pi = yes)             perform pha->pi conversion? (requires gain file)
 (pi_bin_width = 14.6)            Width of Pi bin in eV
  (pi_num_bins = 1024)            Number of values to bin energy into
 (max_cti_iter = 15)              Maximum iterations for the CTI adjustment of each event
 (cti_converge = 0.1)             The convergence criterion for each CTI-adjusted pixel in adu
       (tstart = TSTART)          header key containing start/default time value
        (tstop = TSTOP)           header key containing time of last event
      (clobber = no)              Overwrite output event file if it already exists?
      (verbose = 0)               level of debug detail (0=none, 5=most)
         (stop = none)            end transformations at [chip,tdet,det,tan,sky,none]
     (instrume = acis)            axaf instrument- used for instrument parameter file
    (rand_seed = 1)               random seed (for pixlib), 0 = use time dependent seed
     (rand_pha = yes)             Randomize the pha value used in gain calculations
(rand_pix_size = 0.5)             pixel randomization width (-size..+size) 0=no randomization
      (stdlev1 = {d:time,s:ccd_id,s:node_id,i:expno,s:chip,s:tdet,f:det,f:sky,s:phas,l:pha,l:pha_ro,f:energ
l:pi,s:fltgrade,s:grade,x:status}) TE faint modes event definition string
      (grdlev1 = {d:time,s:ccd_id,s:node_id,i:expno,s:chip,s:tdet,f:det,f:sky,l:pha,l:pha_ro,s:corn_pha,f:e
l:pi,s:fltgrade,s:grade,x:status}) TE graded event format definition string
       (cclev1 = {d:time,s:ccd_id,s:node_id,i:expno,s:chip,s:tdet,f:det,f:sky,f:sky_1d,s:phas,l:pha,l:pha_
f:energy,l:pi,s:fltgrade,s:grade,x:status}) CC faint event format definition string
    (ccgrdlev1 = {d:time,s:ccd_id,s:node_id,i:expno,s:chip,s:tdet,f:det,f:sky,f:sky_1d,l:pha,l:pha_r
f:energy,l:pi,s:fltgrade,s:grade,x:status}) cc graded event format definition string
         (mode = ql)
```

History

16 Dec 2004	updated for CIAO 3.2: use ACIS bad pixel file (`badpixfile` parameter); removed reference to running on a level=2 event file
01 Feb 2005	added note about "Event island contains 1 or more bad pixels" warning
20 Jun 2005	CIAO 3.2.2 patch: minor `acis_process_events` parameter change (default value of `threshfile` is `CALDB` instead of `NONE`)
09 Dec 2005	updated for CIAO 3.3: output filenames include ObsID
01 Dec 2006	reviewed for CIAO 3.4: no changes
21 May 2007	need to set `stop=none` if aspect solution is not provided

URL: http://cxc.harvard.edu/ciao/threads/acispharand/ Last modified: 21 May 2007