

*AHELP for CIAO 3.4*

dither_region

Context: [tools](#)*Jump to:* [Description](#) [Examples](#) [Parameters](#) [See Also](#)

Synopsis

Compute fraction of region area that covers chips

Syntax

```
dither_region infile region outfile [maskfile] [psffile] [gtifile]
[dtffile] [tolerance] [resolution] [maxpix] [convex] [geomfile]
[ardlibpar] [detsubsysmod] [verbose] [clobber]
```

Description

'dither_region' takes a region on the sky and computes its location on the detector. When part of the region falls onto a bad-pixel, goes off the detector, or goes outside the bounds of the mask (optional) the area is decremented.

dither_region will take in either the aspect solution or the aspect histogram as inputs.

- If the aspect solution (stacks accepted) is supplied, the output file will have 1 row for each row in the input with the fractional area computed. This is useful for timing analysis since certain frequencies can be attributed to the source dithering across bad pixels or into chip gaps. One can change the tolerance parameter to speed up the calculations at the expense of skipping rows in the aspect solution file. There will also be a DELTA_T column with the time length between points when the fractions are computed. The fraction is computed at the start of the time 'bin'.
- If the aspect histogram is supplied, then the fractional area is computed for each point in the histogram. There is no timing information; only information about where in the dither pattern the region dithered off the chip.

The outfile will contain either a TIME or CAH_REC column depending on the type of aspect file provided: solution or histogram respectively. It will contain a column with the total fractional area, FRACAREA. It will also contain an array column with data for each chip (3 HRC plates, 10 ACIS chips) with the fraction on each chip; this is useful if the region dithers across chips since the response of the chips can be very different. There is TOTLAREA keyword with the total area of the region (area in units of sky pixels).

If the psffile is provided the PSF fraction will also be computed. It is assumed that the PSF file is correctly normalized before being fed into the tool. The PSF fraction for each chip will also be computed and stored in an array column.

Users can control the speed of execution by adjusting one of two parameters. First, they can pick the resolution of the region (in sky pixels). A small resolution provides a better estimate of the fraction but will slow down the execution time considerably. If the aspect solution is used, the tolerance parameter controls how much the aspect solution has to change before the next fractional area is computed. Another option when an aspect histogram is supplied is to use the points that make up a convex-hull around the aspect histogram rather than all the points. By checking the outer boundary you can tell if any part of the region goes on/off a chip.

The information about badpixels is retrieve through ARDLIB; the parameters should be set for the observation specific badpixel file before running.

The STATUS column in the output indicates why the fraction is less than 1. '00000100' means in the masked out region, '00000010' means it covers a badpixel, and '00000001' means it dithers off the chip. Various combinations of above are possible.

Example 1

```
(1) dither_region infile=pcad0000_asol1.fits
region="circle(4096,4096,2)" outfile=fracarea_vs_time.fits
```

Aspect solution. For every point in the aspect solution, map the sky region (circle) to chip coordinates and determine what fractional part falls off the chip or onto badpixels.

On can then use this information to identify which frequencies are attributed to (part of) the source region dithering across multiple chips.

Example 2

```
(1) dither_region infile=asphist_7.fits region="region(ds9.reg)"
outfile=fracarea_vs_histbin.fits psffile=myspf.fits
```

Aspect solution. The syntax is currently a bit cumbersome; but will be worked. For each aspect histogram bin, the sky region (taken from external file 'ds9.reg') will be mapped to chip coordinates and the fraction of good area will be reported. The output contains the histogram bin number and the fracitonal area.

Once can use 'dmpaste' (or dmjoin) to put the FRACAREA column onto the aspect histogram and then use 'dmtcalc' to correct the 'DURATION' based on the area lost.

The fraction of the PSF that is on a good part of the detector (and for each chip) will also be computed.

Parameters

name	type	ftype	def	min	max	units	reqd	stacks
<u>infile</u>	file	input					yes	yes
<u>region</u>	string						yes	
<u>outfile</u>	file	output					yes	
<u>maskfile</u>	file	input						
<u>psffile</u>	file	input						
<u>gtifile</u>	file	input						

<u>dtffile</u>	file	input						
<u>tolerance</u>	real		INDEF			arcsec		
<u>resolution</u>	real		1	0				
<u>maxpix</u>	integer		1000	1				
<u>convex</u>	boolean		no					
<u>geomfile</u>	file		geom					
<u>ardlibpar</u>	file		ardlib					
<u>detsubsysmod</u>	string							
<u>verbose</u>	integer		0	0	5			
<u>clobber</u>	boolean		no					

Detailed Parameter Descriptions

Parameter=infile (file required filetype=input stacks=yes)

Aspect solution or histogram file

Either can be input. The aspect solution will give information about the time variations (and possibly false periods) that are introduced by the dither pattern. The aspect histogram (see example for correct syntax) gives information about what part of the aspect solution the chips dithers over.

The instrument is determined based on the SIM_Z location; SIM_Z less than 0 implies ACIS, greater than 0 implies HRC.

The header is copied from the first header. Note that the DETECTOR and INSTRUME will probably be set incorrectly.

Parameter=region (string required)

Region specification

The sky region specification for the area to dither across the chip. Region should be in physical coordinates.

Parameter=outfile (file required filetype=output)

Output file name

Parameter=maskfile (file filetype=input)

Mask file

Name of detector mask file. Windows with SAMP_CYC=0 are probably being treated incorrectly right now.

Parameter=psffile (file filetype=input)

An image of the PSF.

An image of the PSF for the source location. Pixels within the sky region ('region' parameter) will be integrated over all the good pixels on the detectors. The fraction (integrated fraction) is output.

This is mostly useful when doing point source analysis. Extended source analysis will benefit mostly from the fractional area, not the PSF weighted value.

The PSF image should be normalized so that the sum(pixels) is ≤ 1.0 . (Although the clever user will surely find a use for an unnormalized PSF.)

Parameter=gtifile (file filetype=input)

Good time interval table

A file with GTIs attached. The time period when the aspect doesn't change more than 'tolerance' arcsec is intersected with the GTI for each chip. The fraction of time is reported in the output file (only when the aspect solution is supplied).

For ACIS, the GTI should be part of a standard DSS that also contains the CCD_ID. For HRC, the same GTI will be used for all plates. If there are multiple GTIs for HRC and/or multiple GTIs for the same chip in ACIS the last GTI is the one that will be used.

Parameter=dtffile (file filetype=input)

Dead time factors file

DTF values averaged over the time interval the region is dither.

Parameter=tolerance (real default=INDEF units=arcsec)

Aspect solution tolerance

The tolerance of aspect solution records to compute. Computing the fraction at every aspect solution record is very time consuming. By picking time records where the aspect solution hasn't changed by 'tolerance' arcsecs we can greatly increase the speed without losing much information (since the dither is relatively slow).

The default value is INDEF which will get the sky pixel size for the given instrument and will divide by 2.

Parameter=resolution (real default=1 min=0)

Resolution of sky region to evaluate fractions

dither_region computes a pixel list for the given sky region and evaluates the fractions on that list. The smaller the resolution value, the more pixels will be evaluated: it will run slower but will give a better result. The bigger the number, the coarser the pixel grid but will speed up execution.

Parameter=maxpix (integer default=1000 min=1)

Controls the maximum number of pixels to evaluate

Sometimes adjusting the resolution can still produce long runs. This sets an absolute upper limit on the number of pixels that will be used to compute the fractions. If the number of pixels is too many, the resolution will essentially be incremented +1 (so 4 times area); if that's too many it will be incremented again (so 9 times area) until the number of pixels falls below this limit. A value of "INDEF" removes any upper limit cap.

Parameter=convex (boolean default=no)

If set to yes, then only those points that makeup the convex–hull around the histogram are used and output.

Parameter=geomfile (file default=geom)

Pixlib parameter file

Points to the pixlib parameter file where various calibrations about the instruments are stored.

Parameter=ardlibpar (file default=ardlib)

Ardlib parameter file

Points to the ARDLIB parameter file. Specifically the badpixel information is needed from ARDLIB and should be set to the observation specific files prior to running the tool.

Parameter=detsubsysmod (string)

Ardlib detector sub–system modifier

Unlike some other ARDLIB enabled tools; dither_region runs on multiple chips and as such does not have a detsubsys parameter. This parameter allows one to modify the internal detsubsys value to allow the response product to be modified. Things such as skipping badpixels be overridden (see "ahelp ardlib" for more information).

Parameter=verbose (integer default=0 min=0 max=5)

Tool verbosity or chatter level.

Parameter=clobber (boolean default=no)

Remove existing files?

See Also

calibration

[ardlib](#)

concept

[subspace](#)

dm

[dmregions](#)

tools

[acis bkgrnd lookup](#), [acis fef lookup](#), [acis set ardlib](#), [acis spec](#), [add grating orders](#), [add grating spectra](#), [asphist](#), [dmarfadd](#), [dmcontour](#), [dmfilth](#), [dmimg2jpg](#), [dmmakereg](#), [dmregid](#), [fullgarf](#), [get src region](#), [mkacisrmf](#), [mkarf](#), [mkexpmap](#), [mkgarf](#), [mkgrmf](#), [mkinstmap](#), [mkpsf](#), [mkrmf](#), [mkwarf](#), [psextract](#), [psf project ray](#), [rmfimg](#), [specextract](#), [tg create mask](#)



Ahelp: dither_region – CIAO 3.4

Smithsonian Institution, Copyright © 1998–2006. All rights reserved.