MIT
Center for Space Research Chandra X-Ray Center

MEMORANDUM
March 29, 2004

To:	Martin Elvis, SDS Group Leader
From:	Glenn E. Allen, SDS
Subject:	Removing hot pixels and cosmic-ray afterglows
Revision:	1.6
URL:	http://space.mit.edu/CXC/docs/docs.html#hotpix
File:	/nfs/cxc/h2/gea/sds/docs/memos/memo_acis_detect_hotpix_1.6.tex

1 Hot Pixels and Cosmic-Ray Afterglows

1.1 Description

Parts or all of the tools acis_build_badpix, acis_find_hotpix, acis_classify_hotpix and acis_process_events are designed to identify and flag hot pixels so that they can be excluded from Level 2 ACIS event data files. Pixels known to be bad or that are outside the valid event detection region are excluded from the search. The algorithm used to identify potential hot pixels also identifies pixels affected by cosmic-ray afterglows and pixels at the corners of the dither pattern for bright sources. However, it is possible to distinguish between hot pixels, afterglows and real source events. This specification describes how to distinguish between these three cases. Pixels identified as bad (and their neighbors) are added to the bad-pixel file and the events on these pixels have a STATUS bit set to one. Events associated with afterglows have an afterglow STATUS bit set to one and the pixel is not added to the bad-pixel file. Events associated with bright sources are not flagged as bad and the pixels on which these events occur are not added to the bad-pixel file.

The tool acis_classify_hotpix is more efficient at removing afterglows than acis_detect_afterglow because the algorithm in acis_detect_afterglow does not identify afterglow events if the events are separated by one or more frames with no events at the location of the afterglow. The tool acis_detect_afterglow (but not acis_classify_hotpix) also misidentifies piled source events as afterglow events. For these reasons, the algorithm described in this specification supersedes the algorithm in acis_detect_afterglow.

1.2 Input

1. A Level 0, 1, 1.5, or 2 event data file (acis*evt0.fits, acis*evt1.fits, acis*evt1a.fits, or acis*evt2.fits)

2. An optional Level 0 bias map file(s) (acis*bias0.fits)

3. An optional Level 1 bad-pixel file (acis*bpix1.fits)

4. An optional Level 1 window mask file (acis*msk1.fits)

1.3 Output

1. An event data file
2. A bad-pixel file

1.4 Parameters

1. infile,s,a,“”,,,“Name of input event file (⟨filename⟩)”
2. outfile,s,a,“”,,,“Name of output event file (⟨filename⟩)”
3. badpixfile,s,a,“”,,,“Name of input bad-pixel file (⟨filename⟩ — CALDB — none — NONE)”
4. outbadpixfile,s,a,“”,,,“Name of output bad-pixel file (⟨filename⟩)”
5. maskfile,s,a,“”,,,“Name of input mask file (⟨filename⟩ — default — none — NONE)”
6. biasfile,s,a,“”,,,“Name of input bias map file(s) (⟨filenames⟩ — none — NONE)”
7. logfile,s,h,“stdout”,,,“Destination of output messages (stdout — STDOUT — ⟨filename⟩)”
8. probthresh,r,h,1.0e−3,1.0e−10,1.0e−1,“Minimum significance of potentially bad pixels after trials (e.g. $1\sigma = 0.159$, $2\sigma = 0.0228$, and $3\sigma = 0.00135$)”
9. biasthresh,i,h,6,3,100,“Minimum bias offset for invalid pixels in adu”
10. expnothresh,i,h,10,2,10000,“Minimum difference in frame numbers for hot pixels”
11. regwidth,i,h,7,3,255,“Size of reference region for comparison (e.g. 7 pixels $\times$ 7 pixels)”
12. clobber,b,h,“no”,,,“Overwrite output files if they already exist?”
13. verbose,i,h,0,0,5,“Amount of messages produced (0=none, 5=most)”
14. mode,s,h,“ql”,,,

1.5 Processing

1. Verify that the specified input files exist. If the parameter clobber = “no,” then verify that the output file does not exist. Verify that the values specified for the parameters probthresh, biasthresh, expnothresh, regwidth, and verbose are in the allowed ranges from 1.0e−10 to 1.0e−1, 3 to 100, 2 to 10000, 3 to 255, and 0 to 5, respectively. If the value of regwidth is even, then add one to regwidth and write a warning message.
2. Create the three-dimensional arrays $A[i, j, k]$ (long integer) and $B[i, j, k]$ (short integer) for the event data and bias maps, respectively. The indices $0 \leq i \leq 9$, $1 \leq j \leq 1024$, and $1 \leq k \leq 1024$ (i.e. i, j, and k correspond to CCD_ID, CHIPX, and CHIPY, respectively). If no bias files are specified as input, then do not create the array $B[i, j, k]$.
3. Read the values of CCD_ID, CHIPX, and CHIPY for each event in the input event file and add one to the appropriate element of A[CCD_ID,CHIPX,CHIPY].
4. If bias files are specified as input, then read the bias for each CCD and store the bias data in array B.
5. If $B[i, j, k] = 4094$, 4095, or 4096, then set the values of $A[i, j, k]$ and $B[i, j, k] = -8192$. These pixels are known to be bad and are excluded from the search for hot pixels and afterglows.

6. Read the input bad-pixel file (if one is specified). If the parameter badpixfile = "CALDB," then use the appropriate bad-pixel ARD file. If one or more of the bad-pixel STATUS bits 0–6, 11, or 13 is set to one for a pixel or column, then set the corresponding elements of $A[i,j,k]$ and $B[i,j,k] = -8191$. These pixels are known to be bad and are excluded from the search for hot pixels and afterglows. However, if the only STATUS bits set to one are 8, 9, 10, or 12, then do not set the corresponding elements of $A[i,j,k]$ and $B[i,j,k] = -8191$. These pixels and columns are included in the search because they are not truly bad.

7. Read the input mask file if one is specified. If the parameter maskfile = "default," then use the appropriate mask described in the acis_build_mask specification (http://space.mit.edu/CXC/docs/docs.html#msk). No events can be reported for the "masked out" region of a CCD. Therefore, these regions are excluded from the search for hot pixels and afterglows by setting the corresponding values of $A[i,j,k]$ and $B[i,j,k] = -8190$.

8. Set the appropriate values of $A[i,j,k]$ and $B[i,j,k] = -8190$ for CCDs that are not used. For example, if the CCD_ID for a device is not included in the keyword DETNAM in the specified infile, the values of $A[i,j,k]$ and $B[i,j,k] = -8190$ for all of the pixels on the device.

9. Compute the total number N_{tot} of pixels to be searched. This value is the total number of pixels for which $A[i,j,k] \neq -8190$, -8191, or -8192. For example, if six CCDs were used for an observation, then $N_{\mathrm{tot}} \approx 6 \times 10^6$.

10. Compute the adjusted bias values by subtracting an offset from each column of each bias. For each CHIPX of each CCD_ID set

$$B\,[\mathrm{CCD\_ID}, \mathrm{CHIPX}, k] = B\,[\mathrm{CCD\_ID}, \mathrm{CHIPX}, k] - \mathrm{median}\,(B\,[\mathrm{CCD\_ID}, \mathrm{CHIPX}, k])\,. \tag{1}$$

When computing the median, exclude the elements k whose values are equal to -8190, -8191, or -8192. Do not change the value of $B[i,j,k]$ if its value is -8190, -8191, or -8192. If all of the pixels in a column have one of these three values, then do not change the values of B[CCD_ID,CHIPX,k] for the column.

11. Identify suspicious pixels in the bias maps. If $B[i,j,k] >$ biasthresh or $B[i,j,k] < -$biasthresh, then set the values of $A[i,j,k]$ and $B[i,j,k] = -8189$. Set the event data STATUS bit 4 (of 0–31) to one for events occurring on these pixels.

12. Identify suspicious pixels in the event data.

    - Create a two-dimensional real-valued array $M[i,j]$ for the mean number of events per node. The indices $0 \le i \le 9$ and $0 \le j \le 3$ correspond to CCD_ID and NODE_ID, respectively.
    - For each CCD_ID and NODE_ID, compute the mean number of events per pixel.

$$M[\mathrm{CCD\_ID}, \mathrm{NODE\_ID}] = \frac{1}{n}\left(\sum_{j}\sum_{k=1}^{1024} A[\mathrm{CCD\_ID}, j, k]\right), \tag{2}$$

    where j is in the range of 1–256, 257–512, 513–768, or 769–1024 for NODE_ID = 0, 1, 2, and 3, respectively, and n is the total number of valid pixels on the node. For each CCD_ID, find the smallest value of M[CCD_ID,l] and set the four values of M[CCD_ID,l] = min(M[CCD_ID,l]), where $l = 0$, 1, 2, and 3 for NODE_ID = 0, 1, 2 and 3, respectively. Pixels for which $A = -8189$, -8190, -8191, or -8192 are excluded from the computation of the values of n and M.
    - Compute the expected number of events R for each CCD_ID, CHIPX, and CHIPY such that $A[\mathrm{CCD\_ID}, \mathrm{CHIPX}, \mathrm{and CHIPY}] \neq -8189$, -8190, -8191, or -8192. The expected number is the average number of events on the valid pixels surrounding the pixel in question.

$$R = \frac{1}{n}\left[\left(\sum_{j=\mathrm{CHIPX}-w}^{\mathrm{CHIPX}+w}\ \sum_{k=\mathrm{CHIPY}-w}^{\mathrm{CHIPY}+w} A[\mathrm{CCD\_ID}, j, k]\right) - S\right], \tag{3}$$

where $S = A[\mathrm{CCD\_ID}, \mathrm{CHIPX}, \mathrm{CHIPY}]$, $w = (\mathrm{regwidth} - 1)/2$ (e.g. $w = 3, 4, 5, \ldots$ if regwidth $= 7, 9, 11, \ldots$), and n is the number of valid pixels surrounding the suspicious pixel. Normally, $n = 48$ (i.e. $7 \times 7 - 1$) if regwidth $= 7$. (The pixel in question is not included in the sum.) However, the number of pixels to include in the computation of R (and n) is smaller near the edges of a node. For example, if the pixel for which R is being computed is (CHIPX,CHIPY) $= (2, 3)$, the valid pixels are those for which $2 \leq \mathrm{CHIPX} \leq 5$ and $2 \leq \mathrm{CHIPY} \leq 6$ ($n = 4 \times 5 - 1 = 19$). Similarly, for the pixel (CHIPX,CHIPY) $= (256, 1023)$ at the node boundary, the valid pixels are $253 \leq \mathrm{CHIPX} \leq 256$ and $1020 \leq \mathrm{CHIPY} \leq 1023$ ($n = 4 \times 4 - 1 = 15$). The values of R and n also exclude pixels for which $A[i, j, k] = -8186, -8187, -8188, -8189, -8190, -8191$, or -8192. R is a real number, not an integer.

- For each CCD_ID, CHIPX, and CHIPY such that $A[\mathrm{CCD\_ID}, \mathrm{CHIPX}, \mathrm{andCHIPY}] \neq -8189$, $-8190, -8191$, or -8192, compute the Poisson probability of obtaining at least $A[\mathrm{CCD\_ID}, \mathrm{CHIPX}, \mathrm{CHIPY}]$ events if the expected number of events is R.

$$P = 1 - e^{-R} \left[\left(\sum_{i=0}^{S-1} \frac{R^i}{i!} \right) + \frac{1}{2} \frac{R^S}{S!} \right]. \tag{4}$$

  where R is the value from equation 3 and $S = A[\mathrm{CCD\_ID}, \mathrm{CHIPX}, \mathrm{CHIPY}]$. Equation 4 is only appropriate if $S > 0$ and $R > 0$. If $S = 0$ and $R > 0$, then equation 4 is simply $P = 1 - 0.5e^{-R}$. If $S > 0$ and $R = 0$, then use equation 4 with R replaced by the appropriate value of M from equation 2. If $S = 0$ and $R = 0$, then use $P = 1 - 0.5e^{-M}$. The probability P is a double-precision real number.

- If $P < \mathrm{probthresh}/N_{\mathrm{tot}}$ or $P > (1 - \mathrm{probthresh}/N_{\mathrm{tot}})$ for the pixel CCD_ID, CHIPX, CHIPY, then set $A[\mathrm{CCD\_ID}, \mathrm{CHIPX}, \mathrm{CHIPY}]$ and $B[\mathrm{CCD\_ID}, \mathrm{CHIPX}, \mathrm{CHIPY}] = -8188$.

13. Compute the total number of suspicious pixels N_{sus}. This number is the sum of all those for which $A[\mathrm{CCD\_ID}, \mathrm{CHIPX}, \mathrm{CHIPY}] = -8188$.

14. Classify the events on the suspicious pixels as being associated with hot pixels, afterglows, or bright sources. For each CCD_ID, CHIPX, and CHIPY such that $A[\mathrm{CCD\_ID}, \mathrm{CHIPX}, \mathrm{CHIPY}] = -8188$,

    - Recompute the values of R (and n) using equation 3.
    - Compute the probability of obtaining R expected events if the mean number of expected events is $M[\mathrm{CCD\_ID}, \mathrm{NODE\_ID}]$.

$$P_{\mathrm{exp}} = 1 - e^{-nM} \left[\left(\sum_{i=0}^{nR-1} \frac{(nM)^i}{i!} \right) + \frac{1}{2} \frac{(nM)^{nR}}{(nR)!} \right]. \tag{5}$$

      Although R is a real number, the value nR is an integer. If $R = 0$, then set $P_{\mathrm{exp}} = 0.5$.

    - Identify valid X-ray events. If $P_{\mathrm{exp}} < \mathrm{probthresh}/N_{\mathrm{sus}}$, then the events on the pixel are most likely events from a bright source at one of the corners of the dither pattern because some of the pixels surrounding the suspicious pixel also have an unusually large number of events. In this case, set $A[\mathrm{CCD\_ID}, \mathrm{CHIPX}, \mathrm{CHIPY}]$ and $B[\mathrm{CCD\_ID}, \mathrm{CHIPX}, \mathrm{CHIPY}] = -8187$.
    - Identify real hot pixels. Compute the values of $\Delta\mathrm{EXPNO} = \mathrm{EXPNO}_i - \mathrm{EXPNO}_{i-1}$ for each event i on a suspicious pixel. For example, if there are six events on a suspicious pixel with EXPNO $= 3$, 6, 7, 8, 10, and 11, then the five values of ΔEXPNO are 3, 1, 1, 2, and 1. The median value of ΔEXPNO for this set is 1. For another pixel with eight events that have EXPNO $= 27$, 163, 345, 746, 839, 1337, 1593, and 1714, the seven values of ΔEXPNO are 136, 182, 401, 93, 498, 256, and 121. The median value of this set is 182. If the median value of $\Delta\mathrm{EXPNO} >$ expnothresh, then the pixel is most likely a real hot pixel instead of an afterglow because afterglow events have closely-spaced exposure numbers. In this case, set $A[\mathrm{CCD\_ID}, \mathrm{CHIPX}, \mathrm{CHIPY}]$ and $B[\mathrm{CCD\_ID}, \mathrm{CHIPX}, \mathrm{CHIPY}] = -8186$ and set the event data STATUS bit 4 (of 0–31) to one for events occurring on the pixel.

- The events on the remaining suspicious pixels (those for which A[CCD_ID, CHIPX, CHIPY] $= -8188$), are most likely dominated by afterglow events. However, not all of the events on these pixels are necessarily due to afterglows. Some of the events may be legitimate X-ray events. To distinguish between real X-ray events and afterglow events, use the values of ΔEXPNO. For simplicity, we assume that only one afterglow has occurred on any given suspicious pixel. In this case, there are two possible scenarios.

  The first event is part of an afterglow.
  If the value of ΔEXPNO between the first and second events is less than or equal to expnothresh, then set STATUS bit 16 (of 0–31) to one for both events. If the value of ΔEXPNO between the second and third events is less than expnothresh, then set STATUS bit 16 to one for the third event. Repeat this process until the value of ΔEXPNO for event i and event $i-1$ is greater than expnothresh. Set STATUS bit 16 to one for event $i-1$, but not for event i. Stop processing the events for this suspicious pixel. The remaining events (if any) are assumed to be valid.

  The first event is not part of an afterglow.
  If the value of ΔEXPNO between the first and second events is greater than expnothresh, then do not set STATUS bit 16 to one for the first event. Continue checking the value of ΔEXPNO until the value is less than or equal to expnothresh for events i and $i-1$. Set STATUS bit 16 to one for both events i and $i-1$. Continue to set the STATUS bit for events to one (as described in "*The first event is part of an afterglow*" until the value of ΔEXPNO for event i and event $i-1$ is greater than expnothresh. Set STATUS bit 16 to one for event $i-1$, but not for event i. Stop processing the events for this suspicious pixel. The remaining events (if any) are assumed to be valid.

15. Write the output event data to the file specified by outfile. The only difference between the input and output event data files (aside from minor changes to the header) is the contents of the column STATUS. Events associated with hot pixels or pixels that have bad bias values have STATUS bit 4 set to one. The events on the eight (FAINT) or twenty-four (VFAINT) surrounding pixels have STATUS bit 5 set to one. Events associated with afterglows have STATUS bit 16 set to one.

16. Write the output bad-pixel file (if one is specified). The new bad-pixel file should include all of the information in the input bad-pixel file as well as all the pixels for which A[CCD_ID, CHIPX, CHIPY] $= -8189$ or -8186. STATUS bit 7 (of 0–15) is set to one for the newly identified bad pixels. The eight (FAINT) pixels surrounding a bad pixel have STATUS bit 8 set to one. For VFAINT mode observations, STATUS bit 10 is set to one for sixteen of the surrounding pixels.

1.6 Special Array Values

The processing described in section 1.5 includes several special values for the arrays A and B. These special values are summarized in the following table.

1.7 Acknowledgements

This specification was developed after discussing the concept with Catherine Grant and reviewing the hot-pixel detection algorithm developed by Takashi Isobe for the mission and trends analysis group. The efforts of Kevin Tibbetts, who helped develop a prototype, and John Houck, who tested the code, are greatly appreciated.

Value	Condition
−8192	The bias value is 4094, 4095, or 4096 for the pixel.
−8191	One or more of the bad-pixel STATUS bits 0–6, 11, and 13 is set to one for the pixel.
−8190	The pixel is “masked out” or is on a CCD that was not used for the observation.
−8189	The pixel is suspicious because the adjusted bias value for the pixel is not within the specified limits.
−8188	The pixel is suspicious because the number of events on it is unusually high or low.
−8187	The events on the suspicious pixel are most likely associated with a bright source.
−8186	The pixel is most likely a hot pixel.