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

MEMORANDUM

October 10, 2010

To:	Jonathan McDowell, SDS Group Leader
From:	Glenn E. Allen, SDS
Subject:	Adjusting ACIS Event Data to Compensate for the CTI
Revision:	7.1
URL:	http://space.mit.edu/CXC/docs/docs.html#cti
File:	/nfs/cxc/h2/gea/sds/docs/memos/memo_cti_correction_7.1.tex

1 CTI adjustment

The ACIS instrument teams at PSU and MIT have shown that a significant improvement in the energy resolution of existing ACIS event data can be obtained by compensating for some of the effects of the parallel and serial charge-transfer inefficiencies (CTIs) of the CCDs. To achieve this improvement, charge is added to, and possibly redistributed among, the pixels of an "event island.[1]" While some pixels in the island gain charge, others may lose charge. Yet, the net change for an event is alway positive. The algorithm described in section 1.4 is used to compute the CTI-adjusted values of `PHAS`.

1.1 Input

1. A Level 1 event-data file (acis*evt1.fits, acis*evt1a.fits)
2. A Level 1 mission time-line file (acis*mtl1.fits)
3. A CTI ARD file (acisD*cti*.fits)

1.2 Output

1. An event-data file that includes the keywords `CTI_APP`, `CTI_CORR`, `CTIFILE`, and `MTLFILE` and may contain the CTI adjusted pulse heights `PHAS_ADJ` and the CTI adjusted values for `ENERGY`, `FLTGRADE`, `GRADE`, `PHA`, and `PI`.

1.3 Parameters

1. `infile,f,a,"",,,"Input event-data file(s)"`
2. `outfile,f,a,"",,,"Output event-data file"`
3. `ctifile,f,h,"CALDB",,,"CTI ARD file ( NONE | none | CALDB | <filename> )"`

[1] An event island, the quantity named `PHAS` in a Level 1 event-data file, contains the pulse-height distribution in a 3 pixel × 3 pixel region centered upon the pixel in which an event is reported. In `VFAINT` mode, `PHAS` is a 5 × 5 array instead of a 3 × 3, but the CTI adjustment is only applied to the central 3 × 3 region.

4. `mtlfile,f,a,"",,,"Mission time-line file ( NONE | none | <filename> )"`

5. `apply_cti,b,h,"yes",,,"Apply CTI adjustment? ( yes | no )"`

6. `max_cti_iter,i,h,15,1,20,"Maximum number of iterations for the CTI adjustment of each event"`[2]

7. `cti_converge,r,h,0.1,0.1,1.0,"The convergence criterion for each CTI-adjusted pixel in adu"`[3]

1.4 Processing

The amount of charge added to each event is based on an estimate of the amount of charge that is lost as charge packets are clocked across the charge traps on the ACIS detectors. This estimate depends on (1) the location of the event on the detector, (2) the temperature of the detector, (3) the density of the charge traps, (4) the amount of charge in the charge packet, and (5) the number of traps that have already been filled.

Perform the following tests before processing begins.

- Verify that the `infile` exists. If it does not, then exit with an error message.
- If `apply_cti = yes`, then verify that
  - `ctifile` $\neq$ `NONE` or `none`,
  - the `ctifile` exists,
  - the `ctifile` contains a binary table with the columns `FRCTRLX`, `FRCTRLY`, `PHA`, `VOLUME_X`, and `VOLUME_Y`, and
  - `max_cti_iter` and `cti_converge` are in their valid ranges (e.g. from 1 to 20 and from 0.1 to 1.0, respectively).

  If one or more of these conditions are not satisfied, then change `apply_cti = no` and produce a warning message.
- If `apply_cti = yes` and `mtlfile` $\neq$ `NONE` or `none`, then verify that
  - the `mtlfile` exists and
  - the `ctifile` contains a binary table with the columns `TCTIX` and `TCTIY`.

  If one or more of these conditions are not satisfied, then produce a warning message.
- If `clobber = no`, then verify that the `outfile` does not exist. If it does, then exit with an error message.
- If `TIMEDEL` is not within the valid range for the specified `ctifile`, then produce a warning, but continue processing.

If `apply_cti = yes`, then perform the following steps, in sequence, for each event.[4]

1. Create the real-valued 3×3 arrays Δ_x, Δ'_x, Δ_y, Δ'_y, `PHAS_ADJ`, and `PHAS_ADJ`$'$. Initialize the arrays to zero.

2. Set `PHAS_ADJ` = `PHAS`.[5] The value of `PHAS` remains unchanged to ensure that it is possible to remove the CTI adjustment or to reapply the adjustment if the algorithm or calibration data are modified.

[2] When Catherine Grant tested the PSU CTI-adjustment tool, she found that the median number of iterations required to satisfy a convergence criterion of 0.1 adu is four. No event required more than ten iterations. Therefore, a default maximum of fifteen iterations should be sufficient to determine the values of `PHAS_ADJ`.

[3] The default convergence criterion is 0.1 adu because this is the default value used for the PSU CTI-adjustment tool.

[4] The algorithm described in this spec applies only to observations where the `DATAMODE = CC33_FAINT`, `FAINT`, `FAINT_BIAS`, or `VFAINT`. The CTI adjustment algorithm for `GRADED` and `CC33_GRADED` mode observations is described elsewhere.

[5] If `DATAMODE = VFAINT`, then set `PHAS_ADJ` equal to the central 3×3 region of `PHAS`. The outer sixteen pixels of the 5×5 array remain unchanged.

3. Perform an iterative loop:

   (a) Set

$$\begin{aligned} \texttt{PHAS\_ADJ}' &= \texttt{PHAS\_ADJ}, & (1)\\ \Delta'_x &= \Delta_x, \text{ and} & (2)\\ \Delta'_y &= \Delta_y. & (3) \end{aligned}$$

   (b) Set the serial CTI adjustment[6]

$$\begin{aligned} \Delta_{x,0j} &= c_{x,0j} s_x \rho_{x,0j} V_{x,0j}, {}^{7} & (4)\\ \Delta_{x,1j} &= c_{x,1j} s_x \rho_{x,1j} V_{x,1j} - c'_{x,0j} s_x \rho_{x,0j} V_{x,0j}, {}^{7} \text{ and} & (5)\\ \Delta_{x,2j} &= c_{x,2j} s_x \rho_{x,2j} V_{x,2j} - c'_{x,1j} s_x \rho_{x,1j} V_{x,1j}, {}^{7} & (6) \end{aligned}$$

   for every element j where

   - $\Delta_{x,ij}$ is an estimate of the amount of charge that should be added to pixel (i, j) to compensate for the effects of serial CTI,
   - the indices i and j of the 3×3 array are in the range from 0 to 2 and are associated with the coordinates `CHIPX` and `CHIPY`, respectively (see sec. 1.5 and Fig. 1),[5]
   - the temperature-dependent scaling factor[8]

$$s_x = 1 + \texttt{TCTIX}\,(T - 153.45 \text{ K})\,, \tag{7}$$

   - `TCTIX`, which depends on the `CCD_ID` of the event, is the fractional change in the serial trap density per degree C and is obtained from the column named `TCTIX` in the `ctifile`,
   - the time-dependent focal-plane temperature

$$T = \left(\frac{t' - t'_k}{t'_{k+1} - t'_k} \right) (\texttt{FP\_TEMP}_{k+1} - \texttt{FP\_TEMP}_k) + \texttt{FP\_TEMP}_k, \tag{8}$$

   - $t' = t + \texttt{TIMEDEL}_{\rm evt}\,(\texttt{TIMEPIXR}_{\rm evt} - \texttt{0.5})\,,$
   - t is the `TIME` of the event,
   - $\texttt{TIMEDEL}_{\rm evt}$ and $\texttt{TIMEPIXR}_{\rm evt}$ are the names of keywords in the `infile`,
   - t'_k and t'_{k+1} satisfy the condition $t'_k \le t' < t'_{k+1}$,[9]
   - $t'_k = \texttt{TIME}_k + \texttt{TIMEDEL}_{\rm mtl}\,(\texttt{TIMEPIXR}_{\rm mtl} - \texttt{0.5})\,,$
   - $t'_{k+1} = \texttt{TIME}_{k+1} + \texttt{TIMEDEL}_{\rm mtl}\,(\texttt{TIMEPIXR}_{\rm mtl} - \texttt{0.5})\,,$
   - $\texttt{TIME}_k$ and $\texttt{TIME}_{k+1}$ are elements of the column `TIME` in the `mtlfile`,
   - $\texttt{TIMEDEL}_{\rm mtl}$ and $\texttt{TIMEPIXR}_{\rm mtl}$ are the names of keywords in the `mtlfile`,
   - $\texttt{FP\_TEMP}_k$ and $\texttt{FP\_TEMP}_{k+1}$ are the elements of the column `FP_TEMP` in the `mtlfile` that are associated with the times $\texttt{TIME}_k$ and $\texttt{TIME}_{k+1}$, respectively,
   - $\rho_{x,ij}$ is the position-dependent serial trap density at the location $(\texttt{CHIPX}+i-2, \texttt{CHIPY}+j-2)$ in the map associated with the `CCD_ID` of the event (see sec. 1.5 and Fig. 1),[7]
   - the pulse-height dependent "volume" occupied by a packet of charge

$$V_{x,ij} = \left(\frac{\texttt{PHAS}_{ij} + \Delta'_{x,ij} + \Delta'_{y,ij} - \texttt{PHA}_l}{\texttt{PHA}_{l+1} - \texttt{PHA}_l} \right) (\texttt{VOLUME\_X}_{l+1} - \texttt{VOLUME\_X}_l) + \texttt{VOLUME\_X}_l, {}^{7} \tag{9}$$

[6] If the `ctifile` does not contain a serial CTI trap-density map for the CCD on which the event occurred, then set $\Delta_x = 0$ and skip item 3b.

[7] The use of the indices $0j$, $1j$, and $2j$ is appropriate for events on `NODE_ID`s 0 and 2. If the `NODE_ID` is 1 or 3, then replace the indices $0j$, $1j$, and $2j$ with the indices $2j$, $1j$, and $0j$, respectively.

[8] If `mtlfile = NONE` or `none`, then set $s_x = 1$.

[9] If $t' < t'_0$, then $T = \texttt{FP\_TEMP}_0$. If $t'_{N-1} < t'$, where N is the number of elements in the vector `FP_TEMP`, then $T = \texttt{FP\_TEMP}_{N-1}$.

- $\mathtt{PHA}_l$ and $\mathtt{PHA}_{l+1}$, which depend on the CCD_ID of the event, are the elements of the column PHA in the ctifile that satisfy the condition $\mathtt{PHA}_l \leq \mathtt{PHAS}_{ij} + \Delta'_{x,ij} + \Delta'_{y,ij} < \mathtt{PHA}_{l+1}$,[10]
- $\mathtt{VOLUME\_X}_l$ and $\mathtt{VOLUME\_X}_{l+1}$, which depend on the CCD_ID of the event, are the elements of the column VOLUME_X in the ctifile that are associated with $\mathtt{PHA}_l$ and $\mathtt{PHA}_{l+1}$, respectively,
- The pulse-height dependent constant $c_{x,0j}$ is set as follows:[7]

$c_{x,0j}$	Condition[11]
0	$(\mathtt{PHAS}_{0j} + \Delta'_{x,0j} + \Delta'_{y,0j}) < \mathtt{spthresh}$
1	$\mathtt{spthresh} \leq (\mathtt{PHAS}_{0j} + \Delta'_{x,0j} + \Delta'_{y,0j})$

- spthresh, a parameter of the tool acis_process_events, is the split threshold,
- The pulse-height dependent constants $c'_{x,0j}$ and $c_{x,1j}$ are set as follows:[7]

$c'_{x,0j}$	$c_{x,1j}$	Condition[11]
0	0	$[(\mathtt{PHAS}_{1j} + \Delta'_{x,1j} + \Delta'_{y,1j}) < \mathtt{spthresh}]$
0	1	$[(\mathtt{PHAS}_{0j} + \Delta'_{x,0j} + \Delta'_{y,0j}) < \mathtt{spthresh} \leq (\mathtt{PHAS}_{1j} + \Delta'_{x,1j} + \Delta'_{y,1j})]$ or $[\mathtt{CHIPX} = 1$ and $\mathtt{spthresh} \leq (\mathtt{PHAS}_{1j} + \Delta'_{x,1j} + \Delta'_{y,1j})]$[12]
1	1	$\mathtt{spthresh} \leq (\mathtt{PHAS}_{0j} + \Delta'_{x,0j} + \Delta'_{y,0j}) \leq (\mathtt{PHAS}_{1j} + \Delta'_{x,1j} + \Delta'_{y,1j})$ and $[\mathtt{CHIPX} \neq 1]$[12]
FRCTRLX	FRCTRLX	$\mathtt{spthresh} \leq (\mathtt{PHAS}_{1j} + \Delta'_{x,1j} + \Delta'_{y,1j}) < (\mathtt{PHAS}_{0j} + \Delta'_{x,0j} + \Delta'_{y,0j})$ and $[\mathtt{CHIPX} \neq 1]$[12]

- FRCTRLX, which depends on the CCD_ID of the event, is the fraction of the charge that is "trailed" one pixel in the serial read-out direction and is obtained from the column named FRCTRLX in the ctifile,
- The pulse-height dependent constants $c'_{x,1j}$ and $c_{x,2j}$ are set as follows:[7]

$c'_{x,1j}$	$c_{x,2j}$	Condition[11]
0	0	$[(\mathtt{PHAS}_{2j} + \Delta'_{x,2j} + \Delta'_{y,2j}) < \mathtt{spthresh}]$
0	1	$[(\mathtt{PHAS}_{1j} + \Delta'_{x,1j} + \Delta'_{y,1j}) < \mathtt{spthresh} \leq (\mathtt{PHAS}_{2j} + \Delta'_{x,2j} + \Delta'_{y,2j})]$ or $[\mathtt{CHIPX} = 256$ and $\mathtt{spthresh} \leq (\mathtt{PHAS}_{2j} + \Delta'_{x,2j} + \Delta'_{y,2j})]$[13]
1	1	$[\mathtt{spthresh} \leq (\mathtt{PHAS}_{1j} + \Delta'_{x,1j} + \Delta'_{y,1j}) \leq (\mathtt{PHAS}_{2j} + \Delta'_{x,2j} + \Delta'_{y,2j})]$ and $[\mathtt{CHIPX} \neq 256]$[13]
FRCTRLX	FRCTRLX	$[\mathtt{spthresh} \leq (\mathtt{PHAS}_{2j} + \Delta'_{x,2j} + \Delta'_{y,2j}) < (\mathtt{PHAS}_{1j} + \Delta'_{x,1j} + \Delta'_{y,1j})]$ and $[\mathtt{CHIPX} \neq 256]$[13]

(c) Set the parallel CTI adjustment[14]

$$\Delta_{y,0j} = c_{y,0j} s_y \rho_{y,0j} V_{y,0j}, \tag{10}$$
$$\Delta_{y,1j} = c_{y,1j} s_y \rho_{y,1j} V_{y,1j} - c'_{y,0j} s_y \rho_{y,0j} V_{y,0j}, \text{ and} \tag{11}$$
$$\Delta_{y,2j} = c_{y,2j} s_y \rho_{y,2j} V_{y,2j} - c'_{y,1j} s_y \rho_{y,1j} V_{y,1j}, \tag{12}$$

for every element j where

- $\Delta_{y,ij}$ is an estimate of the amount of charge that should be added to pixel (i, j) to compensate for the effects of parallel CTI,
- the temperature-dependent scaling factor[15]

$$s_y = 1 + \mathtt{TCTIY}\,(T - 153.45\text{ K}), \tag{13}$$

- TCTIY, which depends on the CCD_ID of the event, is the fractional change in the parallel trap density per degree C and is obtained from the column named TCTIY in the ctifile,

[10]If $\mathtt{PHAS}_{ij} + \Delta'_{x,ij} + \Delta'_{y,ij} < \mathtt{PHA}_0$, then $l = 0$. If $\mathtt{PHA}_{N-1} < \mathtt{PHAS}_{ij} + \Delta'_{x,ij} + \Delta'_{y,ij}$, where $N = \mathtt{NPOINTS}$, the number of elements in the vector PHA, then $l = N - 2$.

[11]While the code should use $\mathtt{PHAS}_{ij}$ instead of $(\mathtt{PHAS}_{ij} + \Delta'_{x,ij} + \Delta'_{y,ij})$ for the comparisons to the spthresh, changing the code might require recalibration.

[12]If the NODE_ID of the event is 1, 2, or 3 instead of 0, then use 512, 513, and 1024 instead of 1 for the condition on CHIPX.

[13]If the NODE_ID of the event is 1, 2, or 3 instead of 0, then use 257, 768, and 769 instead of 256 for the condition on CHIPX.

[14]If the ctifile does not contain a parallel CTI trap-density map for the CCD on which the event occurred, then set $\Delta_y = 0$ and skip item 3c.

[15]If mtlfile = NONE or none, then set $s_y = 1$.

- $\rho_{y,ij}$ is the position-dependent parallel trap density at the location ($\texttt{CHIPX}+i-2, \texttt{CHIPY}+j-2$) in the map associated with the CCD_ID of the event (see sec. 1.5 and Fig. 1),[7]
- the pulse-height dependent "volume" occupied by a packet of charge

$$V_{y,ij} = \left(\frac{\texttt{PHAS}_{ij} + \Delta_{x,ij} + \Delta'_{y,ij} - \texttt{PHA}_m}{\texttt{PHA}_{m+1} - \texttt{PHA}_m} \right) (\texttt{VOLUME\_Y}_{m+1} - \texttt{VOLUME\_Y}_m) + \texttt{VOLUME\_Y}_m,^{7} \tag{14}$$

- $\texttt{PHA}_m$ and $\texttt{PHA}_{m+1}$, which depend on the CCD_ID of the event, are the elements of the column PHA in the ctifile that satisfy the condition $\texttt{PHA}_m \leq \texttt{PHAS}_{ij} + \Delta_{x,ij} + \Delta'_{y,ij} < \texttt{PHA}_{m+1}$,[16]
- $\texttt{VOLUME\_Y}_m$ and $\texttt{VOLUME\_Y}_{m+1}$, which depend on the CCD_ID of the event, are the elements of the column VOLUME_Y in the ctifile that are associated with $\texttt{PHA}_m$ and $\texttt{PHA}_{m+1}$, respectively,
- The pulse-height dependent constant $c_{y,0j}$ is set as follows:[7]

$c_{y,0j}$	Condition[17]
0	$(\texttt{PHAS}_{0j} + \Delta_{x,0j} + \Delta'_{y,0j}) <$ spthresh
1	spthresh $\leq (\texttt{PHAS}_{0j} + \Delta_{x,0j} + \Delta'_{y,0j})$

- The pulse-height dependent constants $c'_{y,0j}$ and $c_{y,1j}$ are set as follows:[7]

$c'_{y,0j}$	$c_{y,1j}$	Condition[18]
0	0	$(\texttt{PHAS}_{1j} + \Delta_{x,1j} + \Delta'_{y,1j}) <$ spthresh
0	1	$(\texttt{PHAS}_{0j} + \Delta_{x,0j} + \Delta'_{y,0j}) <$ spthresh $\leq (\texttt{PHAS}_{1j} + \Delta_{x,1j} + \Delta'_{y,1j})$
1	1	spthresh $\leq (\texttt{PHAS}_{0j} + \Delta_{x,0j} + \Delta'_{y,0j}) \leq (\texttt{PHAS}_{1j} + \Delta_{x,1j} + \Delta'_{y,1j})$
FRCTRLY	FRCTRLY	spthresh $\leq (\texttt{PHAS}_{1j} + \Delta_{x,1j} + \Delta'_{y,1j}) < (\texttt{PHAS}_{0j} + \Delta_{x,0j} + \Delta'_{y,0j})$

- FRCTRLY, which depends on the CCD_ID of the event, is the fraction of the charge that is "trailed" one pixel in the parallel read-out direction and is obtained from the column named FRCTRLY in the ctifile,
- The pulse-height dependent constants $c'_{y,1j}$ and $c_{y,2j}$ are set as follows:[7]

$c'_{y,1j}$	$c_{y,2j}$	Condition[17]
0	0	$(\texttt{PHAS}_{2j} + \Delta_{x,2j} + \Delta'_{y,2j}) <$ spthresh
0	1	$(\texttt{PHAS}_{1j} + \Delta_{x,1j} + \Delta'_{y,1j}) <$ spthresh $\leq (\texttt{PHAS}_{2j} + \Delta_{x,2j} + \Delta'_{y,2j})$
1	1	spthresh $\leq (\texttt{PHAS}_{1j} + \Delta_{x,1j} + \Delta'_{y,1j}) \leq (\texttt{PHAS}_{2j} + \Delta_{x,2j} + \Delta'_{y,2j})$
FRCTRLY	FRCTRLY	spthresh $\leq (\texttt{PHAS}_{2j} + \Delta_{x,2j} + \Delta'_{y,2j}) < (\texttt{PHAS}_{1j} + \Delta_{x,1j} + \Delta'_{y,1j})$

(d) Set the CTI-adjusted pulse heights

$$\texttt{PHAS\_ADJ} = \texttt{PHAS} + \Delta_x + \Delta_y. \tag{15}$$

(e) If

$$N_{\text{iter}} < \texttt{max\_cti\_iter} \text{ and} \tag{16}$$
$$\left|\texttt{PHAS\_ADJ}_{ij} - \texttt{PHAS\_ADJ}'_{ij}\right| \geq \texttt{cti\_converge} \tag{17}$$

for one or more of the nine pixels, where N_{iter} is the number of iterations for the event, then perform another iteration by repeating steps 3a through 3d.

If

$$N_{\text{iter}} \leq \texttt{max\_cti\_iter} \text{ and} \tag{18}$$
$$\left|\texttt{PHAS\_ADJ}_{ij} - \texttt{PHAS\_ADJ}'_{ij}\right| < \texttt{cti\_converge} \tag{19}$$

for all nine pixels, then stop iterating for the event. The computation of the CTI adjustment is done. Based on the conditions shown in Tables 1 and 2, set STATUS bit 20 (of 0–31) equal to zero to indicate that the adjustment converged for the event. Set the values of the array PHAS_ADJ equal the values from the last iteration.

[16] If $\texttt{PHAS}_{ij} + \Delta_{x,ij} + \Delta'_{y,ij} < \texttt{PHA}_0$, then $m = 0$. If $\texttt{PHA}_{N-1} < \texttt{PHAS}_{ij} + \Delta_{x,ij} + \Delta'_{y,ij}$, where $N = \texttt{NPOINTS}$, the number of elements in the vector PHA, then $m = N - 2$.

[17] While the code should use $\texttt{PHAS}_{ij}$ instead of $(\texttt{PHAS}_{ij} + \Delta_{x,ij} + \Delta'_{y,ij})$ for the comparisons to the spthresh, changing the code might require recalibration.

Table 1. Input Conditions

Case	Parameter `apply_cti`	Parameter `doevtgrade`	Parameter `calculate_pi`	Keyword `CTI_CORR`	Comment
1	yes	yes	yes	F	default in pipeline
2	yes	yes	yes	T	default for reprocessing
3	yes	yes	no	F	
4	yes	yes	no	T	
5	yes	no	yes	F	
6	yes	no	yes	T	
7	yes	no	no	F	
8	yes	no	no	T	
9	no	yes	yes	F	
10	no	yes	yes	T	remove adjustment
11	no	yes	no	F	
12	no	yes	no	T	
13	no	no	yes	F	
14	no	no	yes	T	
15	no	no	no	F	
16	no	no	no	T	

If

$$N_{\rm iter} \geq \texttt{max\_cti\_iter} \text{ and} \tag{20}$$

$$\left|\texttt{PHAS\_ADJ}_{ij} - \texttt{PHAS\_ADJ}'_{ij}\right| \geq \texttt{cti\_converge} \tag{21}$$

for one or more of the nine pixels, then stop iterating. The computation of the CTI adjustment has not converged. Based on the conditions shown in Tables 1 and 2, set `STATUS` bit 20 (of 0–31) equal to one to indicate that the adjustment did not converge for the event. Set the values of the array `PHAS_ADJ` equal the values from the last iteration.

4. If the parameter `eventdef` includes `"s:phas"`, then write the unadjusted values of `PHAS` to the `outfile`.

5. If the parameter `eventdef` includes `"f:phas_adj"`, then write the CTI adjusted values of `PHAS_ADJ` to the `outfile`.[5]

6. Based on the conditions shown in Table 1, create or update the keywords `CTI_CORR` and `CTIFILE` in the `outfile` as shown in Table 2.

7. Create or update the keyword `MTLFILE` in the `outfile`.

8. Create or update the keyword `CTI_APP` in the `outfile`. `CTI_APP` is a ten character string with one character for each CCD. If no CTI adjustment is performed for a CCD, then the character for the CCD is `N`. If only a parallel CTI adjustment is performed, then the character is `P`. If both serial and a parallel CTI adjustments are performed, then the character is `B`. For example, the default at present is `CTI_APP = PPPPPBPBPP`.

9. Based on the conditions shown in Table 1, compute the values of `PHA`, `ENERGY`, `PI`, `FLTGRADE`, and `GRADE` as shown in Table 2.

1.5 CTI ARD file

The CTI ARD file in the CALDB has the following structure.

Table 2. Output

Case	Column PHA[a]	Column ENERGY	Column PI	Column FLTGRADE[a]	Column GRADE	STATUS bit 20[b]	Keyword CTI_CORR	Keyword CTIFILE[c]	Keyword GAINFILE[d]	Keyword MTLFILE[e]
1	compute	compute	compute	compute	compute	set	T	ctifile	gainfile	mtlfile
2	compute	compute	compute	compute	compute	set	T	ctifile	gainfile	mtlfile
3	compute	copy[f]	copy	compute	compute	set	T	ctifile	copy	mtlfile
4	compute	copy	copy	compute	compute	set	T	ctifile	copy	mtlfile
5	copy	compute	compute	copy	copy	unset	F	NONE	gainfile	NONE
6	copy	compute	compute	copy	copy	copy	T	copy	gainfile	copy
7	copy	copy	copy	copy	copy	unset	F	NONE	copy	NONE
8	copy	copy	copy	copy	copy	copy	T	copy	copy	copy
9	compute	compute	compute	compute	compute	unset	F	NONE	gainfile	NONE
10	compute	compute	compute	compute	compute	unset	F	NONE	gainfile	NONE
11	compute	copy	copy	compute	compute	unset	F	NONE	copy	NONE
12	compute	copy	copy	compute	compute	unset	F	NONE	copy	NONE
13	copy	compute	compute	copy	copy	unset	F	NONE	gainfile	NONE
14	copy	compute	compute	copy	copy	copy	T	copy	gainfile	copy
15	copy	copy	copy	copy	copy	unset	F	NONE	copy	NONE
16	copy	copy	copy	copy	copy	copy	T	copy	copy	copy

[a] If `apply_cti = yes`, then PHA and FLTGRADE are computed using PHAS_ADJ. If `apply_cti = no`, then PHA and FLTGRADE are computed using PHAS.
[b] If `apply_cti = yes`, then STATUS bit 20 (of 0–31) is set to one for an event only if the CTI adjustment for the event did not converge.
[c] The name of the CTI ARD file used to perform the CTI adjustments.
[d] The name of the gain ARD file used to compute ENERGY from PHA.
[e] The name of the mission time-line file used to compute the focal-plane temperature.
[f] Copied from the `infile` to the `outfile`.

1.5.1 Binary table

The first HDU after the primary HDU includes a binary table with the columns

- CCD_ID,
- CHIPX_LO,
- CHIPX_HI,
- CHIPY_LO,
- CHIPY_HI,
- NPOINTS,
- PHA,
- VOLUME_X,
- VOLUME_Y,
- FRCTRLX,
- FRCTRLY,
- VFTRLX,
- VFTRLY,
- TCTIX, and
- TCTIY.

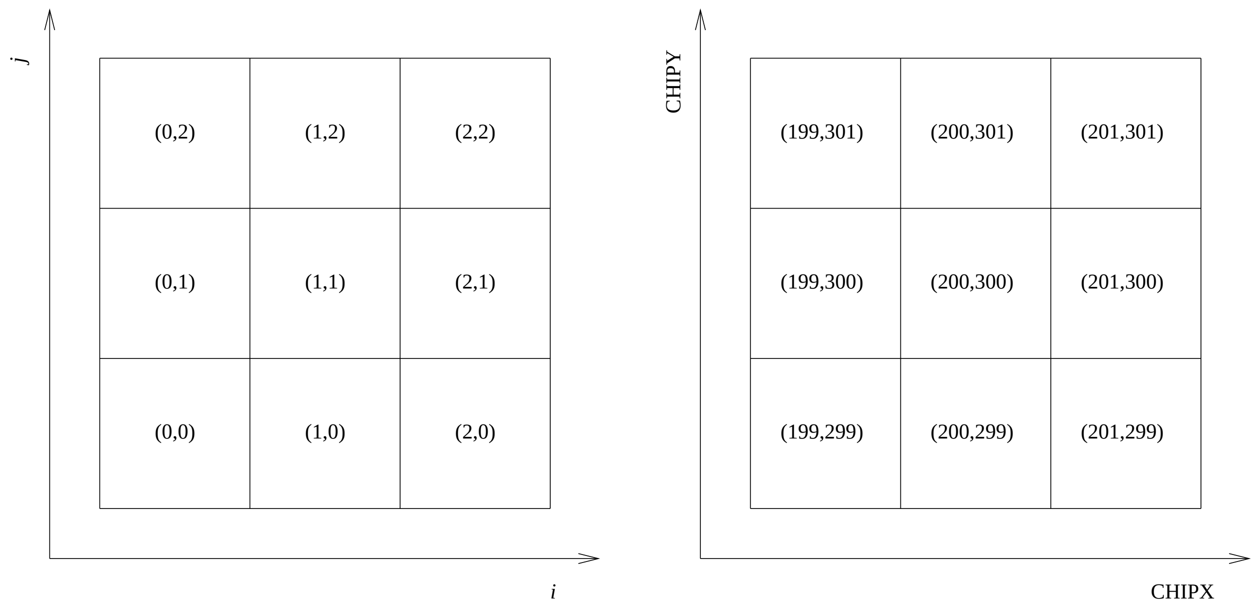

Figure 1: *Left:* The elements (i, j) of a 3×3 pulse-height array. *Right:* The associated chip coordinates for an event that occurred at $(\mathtt{CHIPX}, \mathtt{CHIPY}) = (200, 300)$. Since the 1024×1024 element trap-density maps have indices that run from 0 to 1023, the trap density associated with element (i, j) of the pulse-height array for an event that occurred at $(\mathtt{CHIPX}, \mathtt{CHIPY})$ is the density at the location $(\mathtt{CHIPX} + i - 2, \mathtt{CHIPY} + \mathtt{j} - 2)$.

The columns `CCD_ID`, `CHIPX_LO`, `CHIPX_HI`, `CHIPY_LO` and `CHIPY_HI` define a complete set of spatially-separate regions for the ten CCDs. At present, each row of the table corresponds to one CCD and includes the vectors `PHA`, `VOLUME_X`, and `VOLUME_Y`. Each vector has `NPOINTS` elements. The use of these vectors, and the scalars `FRCTRLX`, `FRCTRLY`, `TCTIX`, and `TCTIY` are described in section 1.4. The scalars `VFTRLX` and `VFTRLY` are used by the algorithm associated with the parameter `check_vf_pha`, which is described elsewhere.

1.5.2 Trap-density maps

There are several HDUs following the binary table. Each one contains a parallel or serial trap-density map for a particular CCD. The keywords `CCD_ID` and `TRAN_DIR` specify the CCD (0–9) and the clocking direction (`PARALLEL` or `SERIAL`), respectively. The maps have indices i and j that each range from 0 to 1023. The value at (i, j) represents the number of parallel or serial traps across which an event at $(\mathtt{CHIPX}, \mathtt{CHIPY}) = (i+1, j+1)$ is clocked as it is read out. To save disk space, the values are stored as two-byte integers. The real-valued trap density $\rho_{ij} = \mathtt{BZERO} + \mathtt{BSCALE} \times M_{ij}$, where `BZERO` and `BSCALE` are keywords and M_{ij} is the integer value in the map for the element (i, j).