



epiclcorr

June 2, 2019

Abstract

The aim of this task is to correct EPIC source time series.

1 Instruments/Modes

Instrument	Mode
------------	------

2 Use

pipeline processing	no
interactive analysis	yes

3 Description

The aim of this task (**epiclcorr**) is to correct *EPIC* source TS produced by **eveselect** for the following effects:

- **Absolute Corrections:** Corrections due to events lost through inefficiencies of various sorts in the mirror-detector system. We will consider these corrections as energy dependent:
 - Vignetting
 - Bad pixels
 - Chip gaps
 - PSF
 - Quantum Efficiency

This will give the Count Rate which would have been detected by this camera with this filter, if it had been observed on-axis with an infinity extraction radius. As such the final Count Rates may be converted to fluxes using tools such as PIMMS.

- **Relative Corrections:** Corrections which vary with time:



- Dead Time.
- GTIs.
- Exposure.
- background counts.

- **Background Substraction:**

Optionally, an *EPIC* background TS can be supplied and in this case the initial source TS will be also background corrected (**backscale** correction included). The first version of **epicccorr** will only accept background TS with exactly the same start and stop time and bin size as in the source TS. The background TS will be corrected for relative effects related to the geometrical area of the source TS. The results of the background corrected rates and errors are stored in columns **BACKV** and **BACKE** in the output file. Bear in mind that a big background area will slow down **epicccorr** run.

epicccorr also needs as input the event file used to create both source and background time series.

epicccorr will perform the **Absolute corrections** making use of the **arfgen** and **evselect** SAS tasks. A broad energy spectrum (50 eV energy bin) will be created using the Data SubSpace (DSS) expressions and it will be passed to **arfgen** SAS task. This task takes calibration information provided by the CAL, performs the necessary corrections for instrumental factors, such us, Vignetting, Bad Pixels, Chip Gaps, PSF, filter transmission, Quantum Efficiency, to get the corresponding **Absolute Correction** for the source TS. If the extraction region is too complex, **arfgen** can fail trying to calculate the source position from the DSS. In this case, the **withsourcepos** parameter must be used, and the user must introduce the source coordinates.

epicccorr output also contains the following extensions:

- **SRC_GTIS**. This extension contains GTIs which cover all the non-null **RATE** values.
- **BKG_GTIS**. This extension (only present if **withbkgset='yes'**) contains GTIs which cover all the non-null **BACKV** values.

epicccorr is also compatible with the output of **barycen** task.

The following corrections will not be included in the first version:

- source spectrum time variability.
- Out of Time Events.
- pileup.

The **relative corrections** will be performed by designing and coding specific algorithms for each instrument *EPICMOS* and *EPICPN* and each mode. A detailed description can be found in section ??.

3.1 Extraction region over more than one CCD

In cases where users select an extraction region over more than one CCD, the algorithm to correct TS changes completely. We have to deal with light curves coming from different detectors and therefore, we



have to take into account all possible detector problems for each CCD. The key point in these cases is to calculate the effective area per CCD that it is being used in the extraction region.

We make a per CCD correction assuming that the counts come from a point source. If one of the CCDs contain a significant background contribution, then this algorithm wont be right. Because of this, we encourage users to apply background subtraction in order to perform all corrections properly.

epiclcorr corrects the TS using the following algorithm.

- Create a TS for each CCD.
- Correct background TS.
- Subtract background.
- Calculate an effective time per bin taking into account the effective area of the source region for each CCD.
- Calculate corrected rate and errors.

If the user does not apply background subtraction, the source TS correction could overestimate the background contribution in cases where background counts are higher or similar to the source counts.

3.2 Absolute Corrections

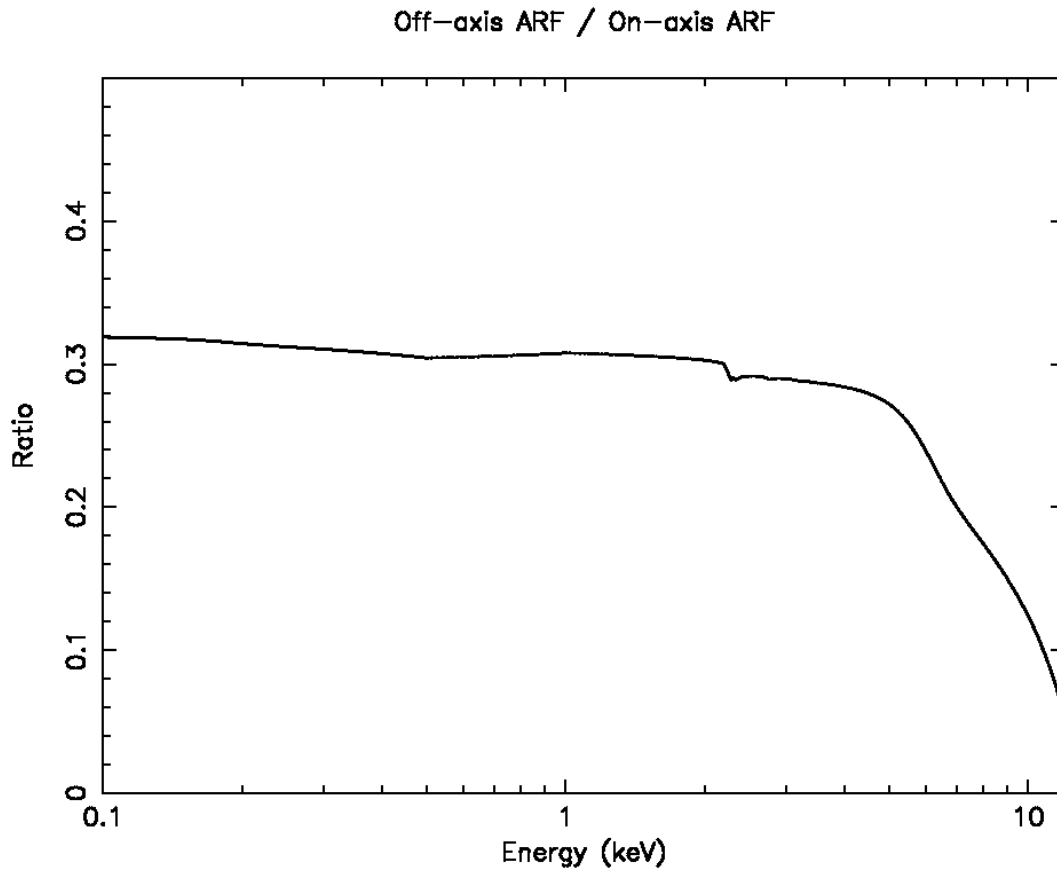
The first version of **epiclcorr** will be optimised to correct time series extracted from **point sources**. The **Absolute Correction** will be finally applied to the corrected source TS background subtracted.

A coarse spectrum (50 eV bins) will be produced by **evselect** from the source region, over the energy range used in the constructions of the TS. This will be used with **arfgn** to find the effective area array for the source as a function of photon energy.

arfgn will be executed twice; The fist call with the spectrum in the real position (off-axis) and with the standard default parameters.

A second call to **arfgn** using the parameters (*withsourcepos=yes sourcecoords="tel" sourcecx=0 sourcecy=0 filterdss=no withdetbounds=yes withbadpixcorr=N modelee=N detmaptype=flat detxbins=1 detybins=1*) will calculate the ideal effective area of this camera/filter combination with no losses.

By dividing the effective area, returned by **arfgn** for an infinite on-axis extraction region by the effective area returned from the actual extraction region, we will obtain an array of absolute corrected factors (Fig.see figure 3.2:). These will be weighted by the source spectrum to find a single absolute correction factor.



we show the ratio of an off-axis (aprox. 10 arcmin) vs. on-axis ARF values with respect to the energy. The ratio remains almost constant up to 4 keV.

3.3 Relative Corrections

epiccorr will treat all these corrections individually, developing and designing specific algorithms to correct the **Relative Corrections**.

GTIs correction will be calculated using the GTI extensions located in the time series file, if any. Otherwise, the STDGTI will be used to calculate the GTIs corrections. Collection regions overlapping more than one ccd will be treated using the GTI of each CCD.

Exposure correction is the time inside each frame where the CCD is taking data correctly (no counting or MIP problems). This correction will be calculated using the values of the *FRACEXP* column in the *EXPOSUnn* extension. Being *FRACEXP* the effective time fraction of each frame.

Dead Time correction is the time where the CCD is not active (CCD read out time). This correction will be calculated using the *Live Time* and *Frame Time* values of each mode.

The *EXPOSUnn* extension, together with the keywords *FRMTIME*, *TIMEDEL*, etc... will be used to calculate the **Dead Time** correction.



4 Parameters

This section documents the parameters recognized by this task (if any).

Parameter	Mand	Type	Default	Constraints
srctslst	yes	table		
Name of the source time series file				
eventlst	yes	table		
Event list used to extract the original time series				
withbkgset	no	boolean	false	
Boolean to enable/disable the background time serie				
bkgtslst	no	table		
Name of the background time series				
outset	no	string		
Name of the output file				
applyabsolutecorr	no	boolean	false	
Absolute correction applied?				
withsourcepos	no	boolean	false	
Source position specified				
sourcecoords	no	string		
Coordinate system for which source position is specified				
sourcecx	no	real		
X position of source				
sourcecy	no	real		
Y position of source				
detxbins	no	int		
define the detector map grid (X).				
detybins	no	int		
define the detector map grid (Y).				

5 Errors

This section documents warnings and errors generated by this task (if any). Note that warnings and errors can also be generated in the SAS infrastructure libraries, in which case they would not be documented here. Refer to the index of all errors and warnings available in the HTML version of the SAS documentation.



Binning (*error*)

Source and Background Lightcurve not aligned.

ZeroBackscale (*error*)

Source or Background backscale 0.

NoCCDID (*error*)

No CCDID keyword found in GTI table. The user has to recreate the eventlist, srctlist and bkgtslist files using the latest SAS version.

EmptyGTIList (*error*)

No GTI found to the corresponding CCD.

ZeroAreaCorrection (*error*)

Sum of all CCD areas is empty. Please check your source and background expression.

EmptyGTI (*error*)

GTI extension empty.

BinSizeTooSmall (*error*)

Light curve bin size smaller than frame time.

CCDs Count Difference (*warning*)

A difference in the sum of the counts of single CCDs, with respect to the counts of the original light curve, has been found.

corrective action:

EventsNotRandomized (*warning*)

Events were not randomized.

corrective action: Run epproc or emproc with randomize option enabled.

ZeroArfgenArea (*warning*)

Zero Effective area. Please check your source and background expression.

corrective action:

wrongArfgenResults (*warning*)

The effective area calculated for a particular CCD is too small that arfgen gives wrong results.

corrective action:

PNTimingRand (*warning*)

Randomization for PN Timing or Burst not implemented

corrective action:

NoPIInfoInDss (*warning*)

No PI information in the DSS

corrective action:

6 Algorithm

```
\begin{itemize}
\item[1.] Get source time serie.
\item[2.] Check time serie integrity.
\item[3.] Get original event list.
\item[4.] Check events list integrity.
```



```
\item[5.] Create time binning.
\item[6.] Apply relative corrections to source time serie.
\item[7.] Calculate errors.
\item[8.] Create corrected light curve.
\end{itemize}
Extensions:
\begin{itemize}
\item[6a:] Absolute corrections.
\begin{itemize}
\item [.1] Calculate spectrum.
\item [.1] Call arfgen.
\item [.2] Apply absolute corrections.
\end{itemize}
\item[6b:] Introduce background time serie.
\begin{itemize}
\item [.1] Check background binning.
\item [.2] Apply background relative corrections.
\item [.3] Normalise background to source collection area.
\item [.4] Substract background time serie.
\end{itemize}
\item[7a.] Check for more initial time series.
\begin{itemize}
\item [.1] Repeat points 4, 5 and 6.
\end{itemize}
\item[7b:] Copy intermediate results to a file.
\end{itemize}
```

7 Comments

•

References