epnoise

June 2, 2019

Abstract

Algorithm to reject soft X-ray noise in the EPIC-pn camera

1   Instruments/Modes

Instrument	Mode
EPIC PN	IMAGING

2   Use

pipeline processing	yes
interactive analysis	yes

3   Description

3.1   General

The task **epnoise** removes soft X-ray noisy frames from EPIC-pn event lists.

The task calculates the number of events per frame between 20 to 30 adu and removes those frames above a certain threshold defined by the `noisecut` parameter. Once the noisy frames have been removed, the exposure time is updated accordingly.

To perform this filtering, the **epnoise** task logic has been divided into two different steps.

During the first step, **epnoise** is run using as input the output files of **epframes** and **badpixfind** tasks. Then, **epnoise** identifies the noisy frames, creates or updates the column `NEVT_FRM`, containing the number of events per frame and CCD with PHA values below a certain threshold, and writes keywords containing suggestions for subsequent filtering. These keywords are:

- `LAMBDA`: lambda of Poissonian fit

- `NORM`: normalization of Poissonian fit

- `NEVT_CUT`: suggested cut value ("10%")

- `NEVT_ALT`: alternative cut value (1.0)

Pixels which are affected by bright celestial sources in this energy range are removed through a mask generation. To create this mask for removing bright sources, **epnoise** calculates the median of the full image and applies a cut using the `sigmacut` parameter. Then a mask for the bad pixels is created and added to the previous mask. The `savemasks` parameter writes the masks of all CCDs to disk.

After this first step of **epnoise**, the rest of the EPIC-pn processing chain is executed, propagating the new column (`NEVT_FRM`) and the new keywords ({knLAMBDA, `NORM`, `NEVT_CUT`, `NEVT_ALT`).

During the second step, **epnoise** filters the final (merged) event list using the `NEVT_CUT` threshold for each CCD, with the following expression per CCD: `NEVT_FRM` $\geq$ `NEVT_CUT`. It then updates the `STDGTI` extension adding the gaps corresponding to the frames that have been removed. Finally, it updates the `ONTIME` and `LIVETIME` keywords.

4   Parameters

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

Parameter	Mand	Type	Default	Constraints

set	yes	filename		

Name of the epframes output file

eventSet	no	filename		

Name of the calibrated event file

identifynoisyframes	no	boolean	yes	yes—no

Identify Noisy Frames?

applyfilter	no	boolean	no	yes—no

Keep output of filtering process?

sigmacut	no	real	3.0	

Sigma cut for bright sources

noisecut	no	int	2	>0

Noise cut (maximum allowed number of soft events in frame)

savemasks	no	boolean	no	yes—no

Save CCD masks to a file

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.

MissingParameter *(error)*
> Missing input file name

TooManyFrames *(error)*
> Too many frames while computing/counting the events per frames.

RawEventFileEmpty *(error)*
> epframes output file is empty

NoisyEventsEmpty *(warning)*
> No noisy events filtered.
> *corrective action:* Check the `NEVT_CUT` value.

6 Input Files

1. The output file of **epframes** + **badpixfind** tasks (step 1).

2. The previous files (one for each CCD) + filtered event list (step 2).

7 Output Files

1. Event file with soft X-ray noisy filtered.

8 Algorithm

```
do i = 1,nfiles
  call identifynoisyframes [step1]
    open file
    call framecounter
    call ftpois
      call dpoiss
        call factrl
          call gammln
```

```
     call epnoisemask
        call createNoisyMask
     call framecounter
     call ftpois
        call dpoiss
          call factrl
             call gammln
     close file
     call writeInfo
   call removenoisyframes [step2]
enddo
call filterEventfile [step2]
```

9 Comments

-

References