



dpssflag

January 12, 2017

Abstract

Flags sources in EPIC combined source list according to several criteria

1 Instruments/Modes

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

2 Use

pipeline processing	yes
interactive analysis	yes

3 Description

The aim of this task is to flag sources in EPIC combined source list in an automatic way for later purification through visual screening.

The flags for each source are written back to the input source list using the column named “FLAG”. The final flag string is a list of T/F characters, T in the position where the corresponding flag is set and F otherwise. The flag string is first read from the source list and then is reset to “FFFFFFFFFFFFFF” if it does not have any previous content (otherwise it preserves initial content). If a given flag is not selected in the command line, the content of the flag string in the corresponding bit is kept unchanged.

Those sources with `SCTS=NULL` are not flagged.

The list of flags set by the task is as follows:

1. Flag 1: All those sources with a MASKFRAC column value lower than the value given by parameter `mfraclimit` will have their first flag set to true, i.e. from 'FFFFFFFFFFFFFF' to 'TFFFFFFFFFFFFFF'. Flag 1 is propagated to the summary row (`ID_INST=0`) if it is set for any of the rows (`ID_INST/ID_BAND`) of that source.
2. Flag 2: The task defines a 'circle of influence' around each point source (whose parameters - position, count rate, etc.- are taken from its summary row). Then all neighbour sources



(in every band and in every instrument) inside this circle will have flag 2 set to T, i.e. from 'FFFFFFFFFFFF' to 'FTFFFFFFFFFFFF'.

The circle radius (in arcs) is calculated using task input parameters **brcolumn**, **op1**, **op2** and **brfactor** as follows:

op1	op2	Expression
sqrt	add	$\sqrt{\text{brcolumn}(i)} + \text{brfactor}$
sqrt	mul	$\sqrt{\text{brcolumn}(i)} * \text{brfactor}$
sqrt	none	$\sqrt{\text{brcolumn}(i)}$
none	add	$\text{brcolumn}(i) + \text{brfactor}$
none	mul	$\text{brcolumn}(i) * \text{brfactor}$
none	none	$\text{brcolumn}(i)$

This 'circle of influence' will have a minimum and a maximum value given by the input parameters **minrad** and **maxrad**.

- Flag 3: Using the information given in column **EXT** and the parameter **extfactor** the task defines a 'circle of extended emission' around each extended source. Then all sources (in every band and in every instrument) inside this circle will have flag 3 set, i.e. from 'FFFFFFFFFFFF' to 'FTFFFFFFFFFFFF'. The circle radius (in pixels) is calculated as :

$$\text{radius}(i) = \text{EXT}(i) * \text{extfactor}$$

Note: This radius is internally converted to arcs taking into account the pixel size of the images used for the source detection.

This 'circle of extended emission' will have a maximum value in arcs given by the input parameter **maxextrad**.

The starting value for a source being considered an extended source is controlled by parameter **extlimit**. Therefore those sources with **EXT** value lower than **extlimit** will not be considered as 'extended' and will not define an area of extended emission.

- Flag 4: likely spurious extended source in 'circle of influence' of bright point source. This flag will be set if source is extended ($\text{EXT} \geq \text{extlimit}$), has Flag 2 set to true (i.e. is in circle of influence of a bright point source) and the causing source is brighter than counts given by parameter **brlimit**, i.e.:

$$\text{SCTS}(\text{causing_src}) > \text{brlimit}$$

Distance is calculated using information in the first **ID_BAND=0** row and then propagated to the other occurrences of that source.

- Flag 5: likely spurious extended source inside 'fixed radius circle' of brighter extended source. This flag will be set if source is extended ($\text{EXT} \geq \text{extlimit}$), is closer than a given distance (**fixext**) to another extended source and it has less than an input fraction (parameter **brexfraction**) of the causing source brightness (calculated from **brcolumn**), i.e.:

$$\text{brcolumn}(\text{src_flagged}) < \text{brexfraction} * \text{brcolumn}(\text{causing_src})$$

Distance is calculated using information in the first **ID_BAND=0** row and then propagated to the other occurrences of that source.

- Flag 6: likely spurious extended source: detection in one band. This flag will be set if source is extended ($\text{EXT} \geq \text{extlimit}$) and has been detected in just one instrument, just one band, i.e. in just one of the 15 images:

$$\text{DET_ML}(i, \text{one_of_15_images}) > \text{detmlfraction} * \text{det_ml_sum}(1:15)(i)$$



7. Flag 7: likely spurious extended source: it has Flag4, Flag5 or Flag6 set to “T”.
8. Flag 8: The PN band 1 sources on a bad PN band 1 column (currently hardcoded, M.Freyberg communication) or those from MOS1 bands 1,2,3 on CCD 1 bad areas (J.Ballet communication) will have flag 8 set to T, i.e. from 'FFFFFFFFFFFF' to 'FTFFFFFFTTTT'.
9. Flag 9: Taking into account information from column given by parameter **badcolumn** (default CUTRAD) modified by parameter **badfactor** the task defines a 'distance to bad area' for the PN band 1 or the MOS1 CCD 1 bands 1,2,3. Then all sources closer than this distance to these bad areas (not **on** the bad areas) will have flag 9 set to T, i.e. from 'FFFFFFFFFFFF' to 'FTFFFFFFTTTT'. This 'critical' distance (in image pixels) is calculated as:

`distance(i) = badcolumn(i) * badfactor`

The activation of Flags 8 and 9 in PN band 1 or MOS1 CCD 1 bands 1,2,3 sources also sets these flags to 'T' for summary rows (ID_INST=0 and ID_BAND=0).

10. Flag 10: sources in RGA spike or in OOT events area. The task reads an input lists of image masks and the corresponding list of instrument numbers for which these masks should be applied ('0' for EPIC, '1' for PN, '2' for M1 and '3' for M2). Then it sets flag 10 to “T” for all sources in which the image mask has “0” valued pixels. When the mask applied is an EPIC-type mask, the rows which are flagged with flag10 are the summary rows (ID_INST=0). When using instrument related masks (PN, M1, or M2 masks) the flag10 is set for the instrument rows (ID_INST=1, ID_INST=2 and ID_INST=3 respectively).
11. Flag 11: sources in the regions set manually with DS9. Given an input image mask which has “0” or “2” in the pixels where a DS9 region is (manually) defined and “1” otherwise, the task sets flag 11 to “T” for all sources in any of such “0” or “2” pixels.
12. Flag12: sources causing spurious detections in their surroundings. Typically, these are bright point sources inside a manual DS9 (exclusion) region, that due to their brightness deserve to be considered for the catalogue. When the DS9 region where they are located is read by the task **ds9tomask**, this task puts a value of “2” in the mask pixels so that it can be distinguishable from “good” pixels. Once **dpssflag** reads this input mask, the sources located in these “2” pixels are flagged with Flag12=”T”. This flag is set only if Flag11 setting is active.

4 Parameters

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

Parameter	Mand	Type	Default	Constraints
-----------	------	------	---------	-------------

srclisttable	yes	table	srclist.ds:SRCLIST	
---------------------	-----	-------	--------------------	--

Name of the EPIC (combined) ML source list table

mfraclimit	no	real	0.75	> 0.
-------------------	----	------	------	------

Threshold value for MASKFRAC column. All sources with MASKFRAC<mfraclimit will have flag 1 set to T (see description).

brcolumn	no	string	RATE	
-----------------	----	--------	------	--

Name of column to be used to define a circle of influence of a bright source (used to set flags 2,4 and 5)



op1	no	string	sqrt	sqrt—none
------------	----	--------	------	-----------

Operation to apply to 'brcolumn' to build the total expression to calculate the radius of the circle of influence to set flag 2 (and 4).

op2	no	string	mul	add—mul—none
------------	----	--------	-----	--------------

Operation to apply to 'brfactor' to build the total expression to calculate the radius of the circle of influence to set flag 2 (and 4).

brfactor	no	real	100.	> 0.
-----------------	----	------	------	------

Number by which the 'brcolumn' is multiplied to obtain the 'circle of influence' radius in arcs (for flags 2,4)

minrad	no	real	10.	> 0.
---------------	----	------	-----	------

Minimum value (in arcs) of the 'circle of influence' of a source (for flag 2,4)

maxrad	no	real	400.	> 0.
---------------	----	------	------	------

Maximum value (in arcs) of the 'circle of influence' of a source (for flag 2,4)

extfactor	no	real	1.	> 0.
------------------	----	------	----	------

Number by which the column EXT is multiplied to obtain the 'circle of extended emission' radius in pixels (internally converted to arcs) (for flag 3 and 5)

extlimit	no	real	1.	> 0.
-----------------	----	------	----	------

Threshold value in pixels for a source to be considered extended.

maxextrad	no	real	200.	> 0.
------------------	----	------	------	------

Maximum value (in arcs) of the 'circle of extended emission' of an extended source (for flag 3)

brlimit	no	real	10000.	> 0.
----------------	----	------	--------	------

Number of counts from SCTS for a point source to be considered causing spurious extended sources around it as set by flag 4

brexfraction	no	real	0.20	> 0.
---------------------	----	------	------	------

Likely spurious extended sources flagged with flag 5 should have less than this fraction of the brighter extended source brightness.

fixext	no	real	200.	> 0.
---------------	----	------	------	------

Likely spurious extended sources flagged with flag 5 should be closer than this fixed distance to a brighter extended source.

detmlfraction	no	real	0.9	> 0.
----------------------	----	------	-----	------

Likely spurious extended sources flagged with flag 6 should have (for just one of the 15 images) more than this fraction of the summed detection likelihood

badcolumn	no	string	CUTRAD	
------------------	----	--------	--------	--

Name of column to be used to define the critical distance to a bad columns in PN/MOS band 1 images (used to set flags 8,9)

badfactor	no	real	1.	> 0.
------------------	----	------	----	------

Number by which the 'badcolumn' is multiplied to obtain the critical distance in image pixels (for flags 8,9)

ootrgamasklist	no	dataset list	ootrga.mask	
-----------------------	----	--------------	-------------	--

List of input masks with position of OOT and RGA spikes as "0" valued pixels

minstlist	no	int list	0	
------------------	----	----------	---	--



List of instrument numbers for input masks ('0' for EPIC, '1' for PN, '2' for 'M1' and '3' for 'M2').

ds9maskset	no	dataset	manual.mask	
-------------------	----	---------	-------------	--

Input mask image with manual exclusion areas as "0" valued pixels

withflag1	no	bool	yes	yes—no
------------------	----	------	-----	--------

Switch to activate flagging with flag 1. The required parameters are: **mfractionlimit**

withflag2	no	bool	yes	yes—no
------------------	----	------	-----	--------

Switch to activate flagging with flag 2. The required parameters are: **brcolumn,op1,op2,brfactor,minrad,maxrad,extlimit**

withflag3	no	bool	yes	yes—no
------------------	----	------	-----	--------

Switch to activate flagging with flag 3. The required parameters are: **extfactor,extlimit,maxextrad**

withflag4	no	bool	yes	yes—no
------------------	----	------	-----	--------

Switch to activate flagging with flag 4. The required parameters are: **brlimit,brcolumn,op1,op2,brfactor,minrad,maxrad**

withflag5	no	bool	yes	yes—no
------------------	----	------	-----	--------

Switch to activate flagging with flag 5. The required parameters are: **brcolumn,brfraction,fixext,extfactor**

withflag6	no	bool	yes	yes—no
------------------	----	------	-----	--------

Switch to activate flagging with flag 6. The required parameters are: **dtmfraction**

withflag7	no	bool	yes	yes—no
------------------	----	------	-----	--------

Switch to activate flagging with flag 7.

withflag8	no	bool	yes	yes—no
------------------	----	------	-----	--------

Switch to activate flagging with flag 8. The required parameters are: **badcolumn,badfactor**

withflag9	no	bool	no	yes—no
------------------	----	------	----	--------

Switch to activate flagging with flag 9. The required parameters are: **badcolumn,badfactor**

withflag10	no	bool	no	yes—no
-------------------	----	------	----	--------

Switch to activate flagging with flag 10. The required parameters are: **ootrgamasklist,minstlist**

withflag11	no	bool	no	yes—no
-------------------	----	------	----	--------

Switch to activate flagging with flag 11. The required parameters are: **ds9maskset**

withflag12	no	bool	no	yes—no
-------------------	----	------	----	--------

Switch to activate flagging with flag 12. The required parameters are:

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.

label (error)



explanation

label (*warning*)

explanation

corrective action: this is the corrective action

6 Input Files

1. EPIC combined source list

7 Output Files

1. EPIC combined source list with column FLAG filled

8 Algorithm

Setting flag 1: MASKFRAC value \leq mfraclimit

```
Foreach source in list{
  If MASKFRAC (value  $\leq$  mfraclimit) then FLAG[1:1](i)=T
}
# flag also summary row
if(ID_INST(i) == 0 and FLAG[1:1](this source)=T) then FLAG[1:1](i)=T
```

Setting flags 2&4: source close to another bright point source

```
Foreach point source i:
  calculate radius of 'circle_of_influence'= brcolumn(i)*brfactor
  if(distance_of_nearest_neighbour < radius(i))then
    Foreach source j/=i :
      !set Flags 2 (and 4):
      if(source_j = source_j-1) then
        FLAG[2:2](j)=FLAG[2:2](j-1)
        FLAG[4:4](j)=FLAG[4:4](j-1)
      else
        calculate distance(i,j)
        if(distance(i,j) < radius(i)) then
          FLAG[2:2](j)=T
          if(EXT(j) $\geq$ extlimit .AND. SCTS(i) $\geq$ brlimit) FLAG[4:4](j)='T'
        endif
      endif
    }
  endif
}
```

Setting flags 3&5: source in extended emission area of an extended source



```
Foreach source i:
  calculate radius of 'circle_of_extension'= ext(i)*extfactor
  if(distance_of_nearest_neighbour < radius(i))then
    Foreach source j/=i :
      !set flags 3 (and 5):
      if(source_j = source_j-1) then
        FLAG[3:3](j)=FLAG[3:3](j-1)
      else
        calculate distance(i,j)
        if(distance(i,j) < radius(i)) then
          FLAG[3:3](j)=T
          if(EXT(j)>=extlimit .AND. brcolumn(j)<brexfraction*brcolumn(i) FLAG[5:5](j)='T'
        endif
      endif
    }
  endif
}
```

Setting flag 6 : extended source detected in one only band

```
Foreach source i:
  Calculate detmlsum(1-15 images) and detmltot(IDINST=0,IDBAND=0)
  if(EXT(i) >= extlimit)then
    if(IDBAND(i)>0 .AND. IDBAND(i)<=5 .AND. &
      detml(i)>detmlfraction*detmlsum(ML_ID_SRC(i)))then
      flag6(ML_ID_SRC(i))="T"
    endif
  endif
}
```

Setting flag 7: likely spurious extended source

```
Foreach source i:
  if(FLAG[4:4]="T" .or. FLAG[5:5]="T" .or. FLAG[6:6]="T") then
FLAG[7:7]="T"
  }
}
```

Setting flag 8/9: source ON/CLOSETO bad PN(MOS) bands 1(1,2,3) column(s)

```
Foreach source i:
  calculate 'critical' distance to a bad column=oper1(badcolumn(i))oper2(badfactor)
  if ((PN and BAND=1) or (M1 and band 1,2,3)) then
    Foreach bad column:
      if(src ON bad colum):
        FLAG[8:8](i)='T'
      elseif(distance_to_bad_column < critical_dist)then
        FLAG[9:9](i)='T' (if PN, 'pnclosetobadon'=yes)
      endif
    }
  endif
}
if FLAG[8:8](src)=T then FLAG[8:8](summary_row)='T'
if FLAG[9:9](src)=T then FLAG[9:9](summary_row)='T'
```



Setting flag 10: source ON exclusion area (OOT/RGA) given by input mask

```
Foreach source i:  
  if(src on ''0'' pixel mask) FLAG[10:10]=''T''
```

Setting flag 11: source ON exclusion area (DS9 region) given by input mask

```
Foreach source i:  
  if(src on ''0'' pixel mask) FLAG[11:11]=''T''
```

Setting flag 12: source ON inclusion area inside exclusion are (bright causing sources)

```
  if(src on ''2'' pixel mask) FLAG[12:12]=''T''
```

9 Comments

From version 3.3.3, dpssflag uses variable “applyAttcalfudge = .true.” for ssclib-3.27 coordinate conversion routines, to take into account problem with attcalc task.

•

References