backcorr

August 16, 2005

Abstract

backcorr makes a background spectrum appropriate to the accumulation region used for a given source spectrum. The task calls **arfgen** twice; once for the source region and once for the background region. The regions are read directly from the input datasets by **arfgen** . This gives the integrated effective area as a function of energy over both regions. From these two effective area curves, a direct normalization of the input background spectrum is performed.

1 Instruments/Modes

Instrument	Mode
EPIC MOS	IMAGING
EPIC PN	IMAGING

2 Use

pipeline processing	yes
interactive analysis	yes

3 Description

backcorr makes a background spectrum appropriate to the accumulation region and position used for the source spectrum. The output of the task is an "EPIC FITS source background spectrum" as defined in the SSC data products ICD [1].

backcorr is a metatask that calls **arfgen** twice in its so-called extended source mode to compute the integrated effective area curves over the source and background regions. The extraction regions, which should be in detector co-ordinates, are read directly from the input spectra (both source and background), as the information is stored in the files using the data-subspace. This explains the presence of the source spectrum (i.e. the spectrum extracted from the source region, in channel and counts) as an input of **backcorr**. Then **backcorr** takes the input background spectrum (also in channel and counts), the arf files and makes the appropriate correction (in a task called *make_backcorr*).

Before going into the details of the task description, it is worth stressing that **backcorr** is primarily intended for use in the PPS. It could also be used in the IAS, but in many cases more sophisticated

analysis will be required to get an accurate and reliable background spectrum. **backcorr** must be robust, work for point sources and for observing modes which allow the extraction regions to be defined. Its main input is a background spectrum which has corrections made to it. The details on how this input is determined is beyond the scope of the task, and should be investigated separately.

3.1 Example

To create a corrected background spectrum, called outbkgspectrum.ds, from the source and background spectra, spectrum.ds and background.ds, the following example can be used.

```
backcorr srcspectrumset="spectrum.ds" bkgspectrumset="background.ds"
outbkgspectrumset="outbkgspectrum.ds"
```

Alternatively, to create a corrected background spectrum using the position in detector co-ordinates, the following could be used:

```
backcorr srcspectrumset="spec1.ds" bkgspectrumset="back1.ds"
outbkgspectrumset="outbkgspectrum.ds" withsourcepos=yes
sourcecoords="det" sourcex=-3917.0 sourcey=-3308.5 bkgdx=-2141.5 bkgdy=-2973.5
```

4 Parameters

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

Parameter	Mand	Type	Default	Constraints
bkgspectrumset	yes	dataset	bkgspectrum.ds	Input dataset containing the spectrum accumulated in the background region.
srcspectrumset	yes	dataset	srcspectrum.ds	Input dataset containing the spectrum accumulated in the source region.
outbkgspectrumset	yes	dataset	outbkgspectrum.ds	Output dataset containing the corrected background spectrum.

withsourceposition	no	boolean	no	yes/no

Indicate whether you wish to specify the source position

sourcecoords	no	string	eqpos	eqpos/pos/det

Indicate the coordinate system for which the source position is specified

sourcex	no	real	0	0

Indicate the x-position of source centre, in terms of the coordinate system specified in sourcecoords.

sourcey	no	real	0	0

Indicate the y-position of source centre, in terms of the coordinate system specified in sourcecoords.

bkgdx	no	real	0	0

Indicate the x-position of the background region, in terms of the coordinate system specified in sourcecoords.

bkgdy	no	real	0	0

Indicate the y-position of the background regions, in terms of the coordinate system specified in sourcecoords.

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.

Wrong instrument name *(error)*
Should be either EMOS1, EMOS2, EPN

WrongSourceCoordType *(error)*
Should be one of det, eqpos or pos

Input spectrum may be wrong *(warning)*
The input spectrum has a negative or null total number of counts
corrective action: none

Corrected spectrum may be wrong *(warning)*
The corrected background spectrum is negative everywhere
corrective action: none

Corrected spectrum may be wrong *(warning)*
The corrected background spectrum has a negative or null total number of counts
corrective action: none

6 Input Files

1. Input EPIC background spectrum [1]
2. Input EPIC source spectrum [1]

7 Output File

1. EPIC **backcorr** corrected background spectrum [1]

8 Algorithm

```
subroutine backcorr

   Arfgen computes the effective area curves for both the source
   (arf_in_src.arf) and background (arf_in_bkg.arf) regions. The
   extraction region is read directly in the input spectra through
   the data sub space.

   Calls the subroutine make_backcorr which :

      Reads in the two temporary arf files: arf_in_src.arf & arf_in_bkg.arf

      Reads in the input background spectrum (==> bkg_input_spectrum)

      Makes the appropriate correction :
         bkg_corrected_spectrum=(arf_src/arf_bkg)*bkg_input_spectrum

      Writes out bkg_corrected_spectrum into a FITS file with the
      same header of the input file + an updated history

   End of make_backcorr

   Delete temporary arf files

end subroutine  backcorr
```

9 Comments

- The various operations made on the spectra are done in units of counts. However, the task supports the input spectrum in the form of "RATES"(to comply to the OGIP/92-07 standard).

10 Future developments

So far, there are no corrections for the particle background component that may be different in the source and background regions. This is due to an insufficient knowledge of what the spatial distribution of the particle background will be. Therefore it is likely that the task will evolve after the launch of XMM, once the first in-flight calibrations are performed. These calibrations, which are critically required, should tell us about the most reliable way to estimate the particle component in a given region. Once this is known with sufficient accuracy, it will be fairly easy to modify *backcorr*.

The most straightforward way will be to decompose the background input spectrum as the sum of a "sky" component and a particle component computed in the background region. The correction will be done in such a way:

```
Bsrc(E) = (Asrc(E)/Abkg(E)) x (Bbkg(E) - Cbkg(E))  +  Csrc(E)

where we define:

Bsrc(E) is the estimated background spectrum in the source region (output)
Bbkg(E) is the background spectrum in the background region (input)
Asrc(E) is the effective area integrated over the source region
Abkg(E) is the effective area integrated over the background region
Csrc(E) is the particle background in the source region
Cbkg(E) is the particle background in the background region

E is the energy.
```

The present version assumes that Cbkg and Csrc are equal to zero at all energies.

References

[1] SSC. XMM Survey Science Centre to Science Operations ICD for SSC Products. Technical Report XMM-SOC-ICD-0006-SSC Issue 2.1, SSC, Mar 2000.