

## Introducción a la Astronomía de Rayos-X

### Guía 03: Imágenes y detección de fuentes

#### Introducción

El objetivo del análisis de imágenes en rayos-X es similar al que este conlleva en otras longitudes de onda: identificación de fuentes hasta cierto límite por sobre el ruido, discriminación entre fuentes puntuales y extendidas, identificación de estructuras y su variación en diferentes bandas de energía, es decir, lo que generalmente se denomina estudio morfológico. En particular, a diferencia de otras longitudes de onda, en Astronomía de Rayos-X, debemos trabajar con dos serias limitaciones: un bajo número de cuentas y grandes cambios de la PSF (función de dispersión de una fuente puntual) en la superficie del detector. Es importante tener presente que el tamaño de la PSF sobre el eje óptico es de 6" en XMM-Newton y 0.5" en Chandra y que este valor es bastante constante a lo largo de toda la cámara EPIC en XMM-Newton, mientras que en ACIS, de Chandra, el tamaño de la PSF se degrada fuertemente al alejarse del eje óptico.

#### La imagen como proyección de la lista de eventos sobre las coordenadas espaciales:

##### Despliegue de listas de eventos en ds9:

```
> ds9
```

Utilizando *ds9*, podemos desplegar las listas de eventos filtradas por *GTI* en la Guía anterior (por ej. Pnclean.fits). Modificar los parámetros de bisección, escala, color, contraste y suavizado hasta obtener una imagen adecuada. De ser posible, reconocer en ella emisión extendida y/o fuentes puntuales.

A partir de la expresión de filtro en los parámetros de bisección, es posible determinar el rango de energía donde la fuente de interés brilla por sobre el ruido de fondo: usar la columna *PI*. Utilizar la impresión a archivo para guardar la imagen obtenida en formato *ps*. O la opción de respaldo o *backup* para guardar la sesión de *ds9*.

##### Imágenes usando *evselect*:

A partir de la lista de eventos de cada cámara filtrada por *GTI* en la Guía anterior (PNclean.fits, MOS1clean.fits, MOS2clean.fits) y utilizando *evselect*, generar una imagen con un bisección tal que cada píxel equivalga a 5", en el rango de energía establecido en el ejercicio anterior:

```
> evselect table=entrada xcolumn=X ycolumn=Y \
  imagebinning=binSize ximagebinsize=?? yimagebinsize=?? \
  withimageset=true imageset=salida expression="(PI in [?:?])"
```

Utilizar un marco RGB de *ds9* para cargar las 3 imágenes resultantes superpuestas. Notar que el menú de bisección ya no está disponible porque estamos desplegando imágenes *FITS* y no listas de eventos.

Notar las diferencias entre la forma de la cámara PN y las cámaras MOS. ¿En qué modo fueron usadas? ¿Son paralelas entre sí? Notar cómo la superposición de las diferentes cámaras aporta a la resolución del espacio entre los distintos CCDs y sus píxeles/columnas malas.

**La imagen de tres colores en rayos-X como indicador del índice espectral:**

Utilizando el *script* `xmmosaic.csh`, a partir de las listas de eventos filtradas por *GTI* podemos construir una imagen *FITS* para cada cámara (*PN*, *MOS1* y *MOS2*) en los rangos de energía *soft* (200-1200 eV), *medium* (1200-2500 eV) y *hard* (2500-8000 eV) o en los que considere más adecuados según lo determinado en la primera parte de esta Guía. En “attfile” escriba “none” para no corregir por mapa de exposición (*flat*). En caso de querer efectuar la corrección, deberíamos apuntar al “attitude file” de la observación (el archivo que termina en *\*AttHk.ds*) en *SAS\_ODF*.

```
> ./xmmosaic.csh PNclean.fits MOS1clean.fits MOS2clean.fits \
  "attfile" 64 300 1000 soft_
> ./xmmosaic.csh PNclean.fits MOS1clean.fits MOS2clean.fits \
  "attfile" 64 1000 2500 med_
> ./xmmosaic.csh PNclean.fits MOS1clean.fits MOS2clean.fits \
  "attfile" 64 2500 8000 hard_
```

Podemos visualizar el resultado cargando las 3 imágenes resultantes en un marco RGB de *ds9* usando el color rojo para la imagen *soft*, el verde para la *medium* y el azul para la *hard*. Ajustar el contraste e identificar las zonas más blandas/duras de la emisión extendida, o el índice espectral de las fuentes puntuales. Elegir un suavizado adecuado para resaltar la imagen y guardarla en formato *ps* usando la impresión a archivo.

Las tres imágenes pueden combinarse también usando la tarea *emosaic* de *SAS*:

```
> emosaic imagesets="soft... med... hard..." \
  mosaicedset="total.fits" withexposure=no sampling=point
```

Nuevamente, podemos utilizar *ds9* para desplegar la imagen, elegir un suavizado y contraste adecuado y guardar. Tener en cuenta que esta imagen “total.fits” ya no tendrá información de la energía de los eventos como en la imagen a 3 colores o bandas.

**Detección de fuentes:**

El problema de la detección de fuentes se reduce a la utilización de una serie de algoritmos estadísticos diseñados para encontrar una región de la imagen que presente un exceso de cuentas respecto del promedio de cuentas del fondo. Para ello se utilizan tareas como “*eboxdetect*” que se basa en una celda deslizante en forma de caja para calcular esta estadística o “*wavdetect*” (*Chandra*) que utiliza pequeñas onditas con tamaños variables para simular la *PSF* del instrumento. La caracterización del tipo de fuente (ya sea puntual o extendida) se realiza a partir de la comparación del perfil radial de la fuente detectada con la *PSF* del instrumento en su posición sobre el detector, utilizando por ejemplo un método de máxima verosimilitud “*emldetect*”. En el caso de *XMM-Newton*, generalmente se utiliza la cadena de tareas “*edetect\_chain*” que realiza todos estos pasos concatenadamente para dar una lista de fuentes detectadas y sus propiedades.

**Paso a paso:**

Como primer ejemplo, con la idea de comprender los pasos involucrados en la detección de fuentes, realizaremos la búsqueda de fuentes en la imagen de la cámara *MOS*, a través de la siguiente serie de pasos:

a) A partir de la lista de eventos filtrada por *GTI* de MOS, usando *evselect*, generamos una imagen en el rango 0.3-10 keV con un bincado de 80 y llamarla “mos\_image\_full.fits”.

b) Obtenemos el mapa de exposición:

```
> eexpmap attitudeset="attfile" eventset=MOSclean.fits \  
  imageset=mos_image_full.fits pimin=300 pimax=10000 \  
  expimageset=MOS_expmap.ds
```

c) Creamos un mapa de detección que anule las regiones donde el tiempo de exposición sea bajo:

```
> emask expimageset=MOS_expmap.ds threshold1=0.25 \  
  detmaskset=MOS_mask.ds
```

d) Realizamos una detección de caja deslizante, usando un fondo estimado localmente:

```
> eboxdetect usemap=no likemin=8 withdetmask=yes \  
  detmasksets=MOS_mask.ds imagesets=mos_image_full.fits \  
  expimagesets=MOS_expmap.ds pimin=300 pimax=10000 \  
  boxlistset=eboxlist_local.fits
```

e) Generamos un mapa del fondo, tapando las fuentes detectadas en el paso anterior:

```
> esplinemap bkgimageset=MOS_bkg.ds scut=0.005 \  
  imageset=mos_image_full.fits nsplinenodes=16 withdetmask=yes \  
  detmaskset=MOS_mask.ds withexpimage=yes \  
  expimageset=MOS_expmap.ds boxlistset=eboxlist_local.fits
```

f) Hacemos una segunda detección de fuentes utilizando el mapa de fondo obtenido:

```
> eboxdetect usemap=yes bkgimagesets=MOS_bkg.ds likemin=8 \  
  withdetmask=yes detmasksets=MOS_mask.ds \  
  imagesets=mos_image_full.fits expimagesets=MOS_expmap.ds \  
  pimin=300 pimax=10000 boxlistset=eboxlist_map.fits
```

g) Realizamos un ajuste de máxima verosimilitud a las fuentes detectadas en el último paso para optimizar el centrado de las fuentes (usando todas las cámaras y rangos de energía) y determinamos si se trata de fuentes puntuales o extendidas al ajustar la PSF local. Usar el factor de conversión de energía *ecf* adecuado según el filtro usado en la observación:

```
> emldetect imagesets=mos_image_full.fits \  
  expimagesets=MOS_expmap.ds bkgimagesets=MOS_bkg.ds \  
  boxlistset=eboxlist_map.fits ecf=2.0 \  
  mllistset=emllist.fits mlmin=10 determineerrors=yes
```

h) Creamos mapas de sensibilidad pixel por pixel:

```
> esensmap expimagesets=MOS_expmap.ds bkgimagesets=MOS_bkg.ds \
detmasksets=MOS_mask.ds mlmin=10 sensimageset=MOS_sens_map.fits
```

i) Desplegamos la lista de fuentes sobre la imagen MOS original usando *ds9*:

```
> srcdisplay boxlistset=emllist.fits \
  imageset=mos_image_full.fits sourceradius=0.01
```

Ayuda/Guía:

[http://xmm.esac.esa.int/sas/current/documentation/threads/src\\_find\\_thread\\_stepbystep.shtml](http://xmm.esac.esa.int/sas/current/documentation/threads/src_find_thread_stepbystep.shtml)

### Detección de fuentes en forma sintética usando *edetect\_chain*:

A partir de la lista de eventos filtrada por *GTI* de cada cámara, usando *evselect* crear una serie de imágenes con un bisección de 80 en los rangos 300-1200, 300-500, 500-1000, 1000-2000, 2000-4500, 4500-12000 eV, llamándolas 'pn\_image\_full.fits', 'pn\_image\_b1.fits', ..., 'pn\_image\_b5.fits' (repetir para mos1 y mos2). Correr la cadena de tareas *edetect\_chain* para cada cámara, utilizando los factores de conversión de energía *ecf* adecuados según el filtro usado en la observación.

```
> edetect_chain imagesets='"pn_image_b1.fits" "pn_image_b2.fits" \
  "pn_image_b3.fits" "pn_image_b4.fits" "pn_image_b5.fits"' \
  eventsets=PN.evt attitudeset=AttHk.ds pimin='300 500 1000 2000 \
  4500' pimax='500 1000 2000 4500 12000' ecf='8.970 6.596 \
  1.953 0.941 0.240' eboxl_list='pn_eboxlist_l.fits' \
  eboxm_list='pn_eboxlist_m.fits' esp_nsplinenodes=16 \
  esen_mlmin=15 eml_list='pn_emllist.fits'
```

Podemos finalmente utilizar la tarea *srcdisplay* para desplegar en *ds9* la imagen completa con la lista de fuentes detectadas superpuesta y guardarla en formato *ps*.

```
> srcdisplay boxlistset=pn_emllist.fits \
  imageset=pn_image_full.fits sourceradius=0.01
```

Ayuda/Guía: [http://xmm.esac.esa.int/sas/current/documentation/threads/src\\_find\\_thread.shtml](http://xmm.esac.esa.int/sas/current/documentation/threads/src_find_thread.shtml)

### Opcional:

Generar una imagen combinada usando el *script images*:

[http://xmm.esac.esa.int/external/xmm\\_science/gallery/utills/images.tar.gz](http://xmm.esac.esa.int/external/xmm_science/gallery/utills/images.tar.gz)

Este *script* realiza correcciones por períodos de fondo intenso (*GTI*), remueve píxeles y columnas malas, realiza un suavizado espacial, corrige por mapa de exposición y combina las imágenes de PN y MOS en diferentes bandas de energía. Su aplicación es directa. Sólo hay que apuntar correctamente el directorio ODF y elegir los rangos de energía en los que trabajar. Para más detalles, pueden remitirse al manual o consultar:

[http://xmm.esac.esa.int/external/xmm\\_science/gallery/utills/README.pdf](http://xmm.esac.esa.int/external/xmm_science/gallery/utills/README.pdf)