Lab session 2:

Coronal X-ray emission: general spectroscopic parameters and light curves

Javier López-Santiago & **Raquel M. Martínez Arnáiz**

Dpto Astrofísica,
Universidad Complutense de Madrid, Spain
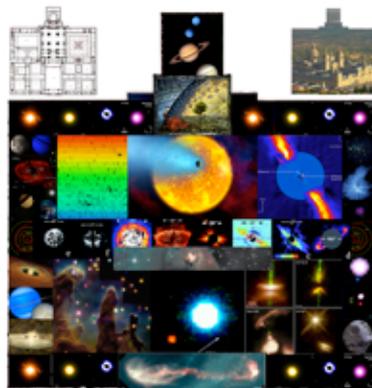
Coronal X-ray emission: general parameters and light curves

Introduction

The final products of the reduction packages of the X-ray missions XMM-Newton (SAS) and Chandra (CIAO) are binary tables (in 'fits' format) containing the 'events' registered in each one of the instrument on-board the satellite (one table for each instrument). Each line of the 'table-of-events' corresponds to a single event and it offers some information, such as the time at which the event occurred, its position in the camera in which it was registered, and its energy (in electron-volts: eV). Generally, not all the events listed in the tables are 'good' ones. Here, we are not going to discuss on how to reject 'bad' events. We encourage the reader to get the user manual of XMM and Chandra to see a detailed discussion on this issue. We already selected the good events from the 'table-of-events' in your home directory.

The reduction packages also allow the user to extract spectra and light curves of the sources detected in the observation. However, no other study can be carried out. SAS and CIAO have no specific tasks to, for instance, study an extracted spectrum. This needs to be done using specific (existing) software, such as XSPEC, or our own (specifically created) software. The aim of this exercise is: **1.** to learn how to study some parameters of the corona of young stars by fitting hot plasma models to low-resolution X-ray spectra with XSPEC, and **2.** to obtain the light curve of, at least, a source detected in an XMM-Newton observation in the proximities of a young cluster, using specific tasks in SAS.

 The following manual was created to be used as a tutorial. Instead of reading everything before executing any action in the computer, we suggest the student to read each point and perform the action one by one.

XSPEC: fitting models

1. Start XSPEC by typing xspec11 in the terminal prompt (you will need to load Heasoft (of HEASARC) first by typing 'head' in the terminal if it was not done first):

> head

This is an alias created by us in your shell environment that invokes Heasoft. If you have any problem just try:

> source $HEADAS/headas-init.csh   (for cshell environments)

Immediately after, type:

> xspec11

Now, the terminal prompt should be changed to 'XSPEC>'.

2. Load the EPIC-PN spectrum data:

XSPEC> data n 'PN_binned_spectrum'

where 'n' is the spectrum number. By default, n = 1. So, if 'n' is not given, the spectrum will take the value n = 1. 'PN_binned_spectrum' is the name of the PN spectrum prepared with grppha (see appendix). Here, you could use one of the spectra in your working directory with the extension 'pha' prepared by us first (e.g., 'src_spec.pn.pha50').

3. Load the EPIC-PN redistribution matrix (omit this step if PN.rmf was fixed during the use of grppha with 'chkey RESPFILE PN.rmf'):

XSPEC> response n PN.rmf

Here, 'n' must take the same value than in step 2. Also in the next step (step 4), 'n' has to be the same number. The calibration files 'redistribution matrix' and 'ancillary file' are different for each spectrum and chip (MOS1, MOS2 or PN).

4. Load the EPIC-PN ancillary file (omit this step if the pn.arf was fixed during the use of grppha with 'chkey ANCRFILE PN.arf'):

XSPEC> arf n PN.arf

5. Load the EPIC-PN background spectrum data (omit this step if the background was fixed during the use of grppha with 'chkey BACKFILE background.fits'):

XSPEC> backgrnd n 'background.fits'

Where 'background.fits' is the name of the background spectrum. This file is in the working directory with the name 'bkg_spec.pn.fits'.

6. Repeat steps 1 to 5 with the EPIC-MOS1 and EPIC-MOS2 data. Remember to change the parameter 'n' for each spectrum. If you forgot to change it, the first spectrum will be overwritten each time you use the command 'data'. You could use, for instance, n = 1 and n = 2 for the MOS1 and MOS2 data, respectively and n = 3 for the PN data.

7. Create the output graphic plotting window:

XSPEC> cpd /xs

This will open a PGPLOT graphic window where the data and models will be plotted.

8. Change to energy coordinates for plotting:

XSPEC> setplot energy

9. Ignore bad channels and energy ranges where the matrix are not currently calibrated:

XSPEC> ignore 0.0-0.3   (for XMM-Newton data)

XSPEC> ignore bad

N.B: if the energy range is given in 'float' format, then XSPEC understand that this is given in energy units. Contrarily, if the numbers are given in 'integer format' (without points, e.g. 'ignore 0-1') then XSPEC understand that the range is given in channel units. We strongly recommend using always energy units.

N.B.2: Sometimes, the calibration of the X-ray spectrum at high energies is not very good: for instance, when the signal of the source is not very high or when if was a high background event during the observation that cannot be discarded. In those occasions, it could be a good idea to 'ignore' also the data at hard energies. For this exercise, a good choice is 'ignore 5.0-**', that will ignore all the data with energy over 5.0 keV.

10. Plot the data (with X and Y axis in logarithm):

XSPEC> plot ldata

At this stage, you should see your data (three spectra) plotted in the PGPLOT window. White colour is used for the first spectrum (n = 1 in step 2), red for the second one (n = 2), and green for the third one (n = 3).

XSPEC: fitting data

XSPEC fits automatically the model to each spectrum, so all the commands given in the following are typed only once, instead of three times. For this exercise, a hot plasma model with absorption will be used. Here, the absorption model is a

multiplicative model that represents both the interstellar and circumstellar absorption.

1. Load the model:

XSPEC> model wabs*apec

Where WABS is the absorption model (which has only one parameter: the column density N_H) and APEC is the hot plasma model (with four different parameters: temperature, abundance, redshift, and normalization constant). After loading the model, you will be asked to give initial values to each variable. You can use the default values given by XSPEC or change them. Obviously, the redshift should be set to zero in our stellar case.

2. Fitting the model:

XSPEC> fit 300   (300 is the maximum number of iterations)

This will fit the model to our data showing, at the end, the results for each fitted parameter. By default, the abundance remains fixed in XSPEC when using APEC. To 'liberate' it, you should write:

XSPEC> thaw n

Where 'n' is the number of the parameter in the list of parameters given by XSPEC. To know it, type:

XSPEC> show par

One could also fix other parameters, such as the temperature or the column density. This is made by typing:

XSPEC> freeze n

Where n is the number of the parameter to be fixed to a value.

N.B.: after fixing or liberating a new parameter, nothing is changed in the fit until you type again 'fit' in the terminal.

3. Finally, to plot the results (both model and residuals):

XSPEC> plot ldata del

XSPEC: changing model parameters

A useful thing in XSPEC is the possibility of changing the values of any parameter at any time you want. This is made with the command 'newpar'. Contrarily to the case explained above, when the value of a parameter is changed with 'newpar' it immediately affects the model

1. Change the value of the temperature (kT) in the model:

XSPEC> newpar 2 3.0

2. Now, plot the data with the model again:

XSPEC> plot ldata del

Now, in the graphic window you should see the effects of changing the temperature in the model. The fit should not be good now. If you type 'fit' again in the terminal, a new fit is performed and the new value of kT is given.

XSPEC: adding new models

In general, stellar coronae are not well defined by a unique thermal model. As in the case of the sun, they present a gradient in temperatures. In some cases,

generally with cool coronae, one can obtain a good result when fitting a model with only one temperature (1-T model) but, usually, multi-temperature models give better results. Only when the total number of counts in the X-ray spectrum is small, 1-T models produce good results.

1. To add a second model type:

XSPEC> addco 2 apec

This will add a second plasma model to the model defined at the beginning. You will be asked to give initial values to the parameters of the new model. Obviously, the abundance should be the same than in the other thermal model. To fix it, when XSPEC ask you to give the initial value for the abundance, type:

3:apec:Abundanc> = 7

N.B: parameter 'n' can be fixed equal to parameter 'm' at any time during the fitting process by typing 'newpar n = m'

2. Fit the spectra:

XSPEC> fit

Here we recommend making proofs with different initial values for each parameter and letting XSPEC fit the model to the spectra. If the results are not good (for instance, if the second temperature has no sense) you can delete the model by typing:

XSPEC> delco 2 apec

3. Plot the results:

XSPEC> plot ldata del

Then, the new multi-temperature model is over-plotted to the spectra. XSPEC plots the whole model [wabs*(apec+apec)], including both thermal components. If you want to see the two thermal components separately, you could type 'plot model'. To turn back to the fit, type again 'plot ldata del'.

XSPEC: inspecting the results (goodness of the fit)

Sometimes happens that small variations in the initial values produce different results when fitting the model with similar goodness of fit. Here, the goodness of fit is measured by the Chi-Square statistics, although other statistics could be used inside XSPEC. The value of the chi-squared is given after the fitting process by XSPEC. Good values of the reduced chi-squared are around 1. XSPEC permits to study the parameter space. Thus, one could see if two parameters are correlated or if some parameters are not well defined

1. Use 'steppa' to create stepped parameters from a given value (see example bellow):

XSPEC> steppa par1 low1 high1 num1 par2 low2 high2 num2

Here, par1 and par2 are the list numbers of two different parameters of the model (say abundance and column density), low and high are the minimum and the maximum values of this parameter to be checked, and num is the number of steps. For example, imagine you are using a model where the first parameter is the column density and the third one is the abundance. Then, the sequence:

XSPEC> steppa 1 0.0 0.1 10 3 0.01 1.0 10

will perform fits to the data with the column density ranging from 0.0 to 0.1 (10 steps) and the abundance ranging from 0.01 to 1 (10 steps). The result is a matrix with the chi-squared of each fit, which can be plotted in the graphic device.

2. Plot results of 'steppa':

XSPEC> plot contour

This will plot the results of 'steppa' in the graphic window. To turn back to the model plot, type 'plot ldata del'.

For this exercise, we recommend to use 'steppa' first with the column density and the abundance and then with the two temperatures of the 2-T model, to see if one of these four parameters is less defined than the others.

APENDIX: working with 'grppha'

If you want to prepare the spectra by yourself, you will need to use the task GRPPHA, available in the Ftools sub-package of HEASOFT. GRPPHA is an interactive command driven task to define (or redefine) and/or display the grouping (binning) and quality flags. Eventually, you can append some information to the 'header' of the resultant file as, for instance, the calibration files to be used. The task can be used interactively or in a single command. In the following we show two examples of using GRPPHA with a PN spectrum (the same should be done for MOS spectra):

1) Interactive use:

After typing grppha, you are requested to give the name of the original spectrum (e.g. 'src_spec.pn.fits'). Then, the output name is requested (in this example, we will assume that the binning will be 6, so we will use the name extension 'pha6').

       input filename: src_spec.pn.fits

       output filename: src_spec.pn.pha6

The name extension is optional. You could write another one, but it is better to use extensions that will permit you to recognise the binning parameters easily. Then you could (optionally) add some information to the output file, such as the name of the background file and the calibration files to be used by XSPEC, simply typing:

       chkey BACKFILE bkg_spec.pn.fits

       chkey RESPFILE PN.rmf

       chkey ANCRFILE PN.arf

Finally, set the binning and exit:

    group min 6

    exit

2) GRPPHA in a single line:

>grppha infile=src_spec.pn.fits outfile=src_spec.pn.pha6 comm='chkey BACKFILE bkg_spec.pn.fits & chkey RESPFILE pn.rmf & chkey ANCRFILE pn.arf & group min 6 & exit'

SAS: generate light curves

1. Start SAS by typing sasrun in your home directory:

> sasrun

SASRUN is an alias created in your '.cshrc' in your home directory. It first loads the HEASOFT package and then run SAS (the XMM-Newton reduction software).

2. Display the PN image in your home directory (to look for X-ray sources):

> ds9 pn.ds &

3. Inspect the PN image:

To inspect the image, you will need to use appropriate binning, zoom, and scale. In the DS9 top menu, go to 'bin' and select block 32 (this is a good choice for our image). Now, go to 'zoom' and select 'zoom 1/2'. The whole field of view should be now visible in the DS9 window. Finally, change the scale to 'logarithmic' in the scale menu.

4. Change colour map, contrast, and brightness (optional):

The colour map, the contrast, and the brightness of the image can be also changed in DS9. The colour map is changed in the top menu. Go to 'color' and choose one of the maps (we strongly recommend 'blackbody': bb). The contrast and brightness are changed simply moving the mouse while pressing the right button.

5. Select the source to extract the light curve:

After step 4 you should be able to see several sources in the image. Select one of them. To extract the light curve of this source with SAS you will need to know its position in the image and its extension (or better, the extension of its PSF in the image). For this exercise, we will use only physical coordinates (given by DS9 in

the top panel). For the extension of the PSF, as an approximation, you could use 330 pixels (~ 17 arcsec).

N.B.: You may need to change the zoom to better establish the centre of the source.

6. Create the light curve with EVSELECT:

```
> evselect table=0682_0148300101_EPN_S003_ImagingEvts.ds withrateset=Y \

   rateset=curve.fits timebinsize=200 maketimecolumn=Y makeratecolumn=Y \

   expression='#XMMEA_EP && (PI in [300:5000]) && (PATTERN<=4) && ((X,Y) IN
circle(xpoint,ypoint,330)) '
```

This will extract the light curve of a source situated in physical coordinates: xpoint and ypoint (remember to change xpoint and ypoint by the values obtained in DS9) in a radius of 330 pixels from the centre. For this example, we have chosen an energy (PI) range of 0.3 to 5.0 KeV. You can also change it and extract different curves in different energy ranges. The time bin-size is 200 seconds.

7. Inspect the light curve

The output file (curve.fits) can be opened with the task 'FV' of HEASOFT. Just type:

```
> fv curve.fits
```

Two windows are opened: the first one with the FV menu and the second one with some columns. In the line with 'Extension' name 'RATE', mark 'plot' in the 'view' column. Another window is opened, with some lines in the left panel and the plotting parameters in the right panel. Now, you only need to write 'TIME' in the X box and 'RATE' in the Y one and mark 'GO' in the bottom panel. The light curve should now be plotted in a different graphic window.