

Content of the main distribution directory :

- 1) readme.pdf (this file; pdf version of readme/help file)
- 2) readme.html (html version of readme/help file)
- 3) install_sxtEEFModule.csh (installation shell script for CSHELL users)
- 4) install_sxtEEFModule.sh (installation shell script for Bourne SHELL users)
- 5) sxt_ARFModule_v02.py (python script called for sxtARFModule tool)
- 6) make_sxt_EEF_v02_py2.py (python script called for sxtEEFmaker tool)

*****Readme file for ****sxtARFModule **** is written in python*****

* The ARF v04_20190608 is most recent version of SXT ARF which with combination of a command line tool sxtARFModule can be used for spectral study of any astronomical object observed by AstroSat-SXT. This tool takes the recent versions of ARFs as input and scales this as per the observation specific parameters like the area of the source extraction region and the distance of source centroid from the bore-sight/optical axis (causing Vignetting), and write a new ARF file which can directly be used in XSPEC.

* This version of ARF is applicable for most of the non-piled SXT targets. However one may expect 1-3% uncertainties in flux estimations after using the recent ARF + sxtARFModule (tool).

* This version of ARF is tested for thermal SNR 1E0102, CasA, and blazars 1ES1959+650 +, and results are within the acceptable limits.

* If using this ARF and Module for your scientific investigations please refer the url of TIFR-POC (www.tifr.res.in/~astrosat_sxt) in data analysis section and add following texts in acknowledge section of publications

"This work has been performed utilizing the calibration data-bases and auxiliary analysis tools developed, maintained and distributed by AstroSat-SXT team with members from various institutions in India and abroad."

* According to most recent studies of SXT point spread function (PSF) of point sources, a circle of 15' radii centered at source position is sufficient to include > 95% of the total photons from the target. Therefore users can choose any circular radius between 13-17 arcmin for lowest possible systematic. Note that the sxtARFModule tool is optimized for any kind source region selections (circle, box, annulus). The circular regions are preferred extraction for sources, however, in cases it is not possible the sxtARFModule will anyway generate appropriate ARF by approximating the area to an area of circle equivalent. One will have noticeable problem if e.g., for an elliptical source regions the ratio major-axis/minor-axis [a/b] is more than 2. For a source with piled-up observations one may need to remove central a few arcmins and hence annulus/elliptical annulus will be a most suitable source extraction region. The sxtARFModule is suggested to use with spectrum file + events file + other input options, which in principle, will generate appropriate ARF for this kind of source extraction.

* The second component of PSF (broad component of double King's profile) is still found to contribute mildly to the regions/outer rings upto 19 arcmin from source centroid and hence SXT team discourage users to extract background regions from same frame as it will always be an overestimation of back-ground. The users are suggested to use background spectrum distributed by TIFR-POC (www.tifr.res.in/~astrosat_sxt) for spectral studies.

One may estimate constant background count rates for a source region and subtract it to get background subtracted lightcurve. This subtraction will be affected by a systematic due to faint modulations of daily and satellite orbit time scale variations. These may be crucial for the sources with count rate ~ 0.5 or less.

* As stated earlier the recent ARF and the modified ARFs using this tool are readily usable for spectral analysis. Users are free to use XSPEC (<https://heasarc.gsfc.nasa.gov/xanadu/xspec/>) or SPEX (<https://www.sron.nl/astrophysics-spx>) or any other similar tools. Please refer to respective manuals for the basics of the spectral fittings using XSPEC or SPEX module. One may also use the python supported version of XSPEC or

* The users are encouraged to contact nilima@tifr.res.in and/or sunil.chandra355@gmail.com in case of any bug-reports or suggestions or technical difficulties.

* Please install/upgrade the pre-requisite python libraries (see the end of this document for a list of these libraries) before use of sxtARFModule for hassle free use. The libraries like astropy, lmfit, scipy, numpy and matplotlib are a few important libraries which you should upgrade to recent versions.

* The present version of SXT pipeline does not correct for the gain shifts. The users are requested to check for gain shift for a particular observations. One can load spectrum files and models and then run

> gain fit

One should use fixed gain slope = 1. Only gain offset should be kept free.
Once you have decided appropriate gain offset value (best fit) then these values can be forzen in the model so that the model fitting doen't get affected.

After removing your gain fit component one can use

> gain 1 1 gain-offset-value

*PYTHON libraries HELP is given at the end of file.

* For Linux machines, please install it using this method...

#-----Installation Part

A) for BASH shell users:

1) Download the attached tarfile

2) Untar it using

-> tar xzvf SXTEEFModule.tar.gz

-> cd SXTEEFModule

-> chmod u+x install_sxtEEFModule.sh

-> ./install_sxtEEFModule.sh

-> source ~/.bashrc [or ~/.profile based on text outputs on terminal, while running above command]

B) For CSHELL users:

1) Download the attached tarfile

2) Untar it using

-> tar xzvf SXTEEFModule.tar.gz

-> cd SXTEEFModule

-> chmod u+x install_sxtEEFModule.csh

-> ./install_sxtEEFModule.csh

-> source ~/.cshrc [or ~/.profile based on text outputs on terminal, while running above command]

These will enable users to utilyze two tools (sxtEEFmaker and sxtARFModule) available from terminal window. Please run following to test whether these are linked properly of not...

sxtARFModule -h [OR sxtARFModule --help]

and

sxtEEFmaker -h [OR sxtEEFmaker --help]

#-----

Following lists details of the input options...

You can generate this by running following in a terminal window

-> python sxt_ARFModule_v02.py -h (if you directly use this python file)

or

sxtARFModule -h (If you install it as per guidelines)

=====

Running Task: sxt_ARFModule_v02.py

Version: 0.01 Release Date: 2019-06-16

Developed By : Sunil Chandra,

CSR, North-West University, 10, June, 2019

+++++

The estimations provided by this script is based on actual SXT observations. The profiles used here are corrected for background counts. This script is best suited for point sources.

We may expect ~5 % uncertainty in flux estimations when extracting ARFs for sources with fluxes ~ 1 Crab or brighter...

The parametric way of estimating the factor uses kings's profile as functional form + best fit archived parameters

+++++

=====

Usage: sxt_ARFModule_v02.py [options]

Options:

-h, --help show this help message and exit

-m MODE, --mode=MODE Mode for extraction; Two options a) using table and b) using fit parameters Default is b) Input options are ['a', 'tbl', 'tab', 't', 'table'] for a) and one of ['b', 'par', 'param', 'fit', 'fitpar'] for b)

-r RADIUS, --radius=RADIUS Source Region Size in arcmin.. Always use for cicular regions... Multiple radii (r1, r2, r3) can be entered in format or 'r1nr2nr3' or 'r1ANDr2ANDr3' or 'r10Rr20Rr3' Default is 15 arcmin

--sxtarf=SXTARF The Full Path of Base ARF provided by SXT Team... For Example 'sxt_pc_excl00_v04.arf'

-e EVTFILE, --evtfile=EVTFILE The Events file in case you are entering source postion in sky pixels instead of detector coordinates.....Default = None

--sxtpha=SXTPHA The Name of the input SXT spectrum [needed to fetch source region related information]... For Example 'sxt_pc_mytarget_src.pha' Default is None..[This means source coordinates should be entered manually]

-x XPIX, --xpix=XPIX X coordinate (RA) of the source position [sky/RAW pixels] ... Default is None [If None you should provide spectrum as input file]

-y YPIX, --ypix=YPIX Y coordinate (DEC) of the source position [sky/RAW pixels] ... Default is None [If None you should provide spectrum as input file]

-o OUTSTEM, --outstem=OUTSTEM Stem for output ARF File...

--coordtyp=COORDTYP Input Coordinate Type, .i.e., Detector Coordinates or Sky Coordinates...Default is Sky The input options are ['Sky', 'sky', 'SKY', 's', 'S', 1, True] and/or ['RAW', 'raw', 'Raw', 'R', 'r', 0] Default is 'SKY'

--vigcorrflag=VIGCORRFLAG The flag for Vignetting Correction for off-axis SXT observations, if needed... Accepted options are [1., True, 'yes', 'YES', 'y', 'Y'] Default is 'no'

--pltflag=PLTFLAG The flag used for making the ARF diagnostics plot to display various versions... Accepted options are [1., True, 'yes', 'YES', 'y', 'Y'] Default is 'no'

--onlyceoff=ONLYCEOFF The flag to print only the coefficients for the input radii..No ARF files will be generated..... Accepted options are [1., True, 'yes', 'YES', 'y', 'Y'] Default is 'no'

Descriptions:

This tool can be used in two ways a) Just to estimate the correction factor (f) to estimate the actual normalization after fitting some spectra extracted for a source radius (r arcmin). This is an external factor which users can utilize to correct normalization/flux [effectively over 0.3-7.0 keV band] is following way :
corrected Norm. (Flux) = (2-f) * Norm (Flux)

Here you just need to enter following important parameters...

- i) radius (R) or radii (in format of R1nR2nR3 or R1ANDR2ANDR3 or R10RR20RR3)
- ii) onlycoeff=yes ; a flag to confirm that we want only coefficients no ARF generations
- iii) Mode of estimation [optional] : you may either choose pre-calculated table (A) or pre-estimated Kings profile (B)
(Both the results agree within the uncertainties)

For this options users can ignore other options ... i.e. Default valuse if users are not defining then explicitly

Example :

```
python ~/Desktop/SXTEEFModule/sxt_ARFModule_v02.py --radius=20n15n10n5 --onlycoeff=1 --mode=b --onlycoeff=yes
```

or

```
python ~/Desktop/SXTEEFModule/sxt_ARFModule_v02.py -r 20n15n10n5 --onlycoeff=1 -m b --onlycoeff=yes
```

b) To generate the corrected ARF for your particular source extraction region.

The important input options for this step are

i) radius (R) or radii (in format of R1nR2nR3 or R1ANDR2ANDR3 or R1ORR2ORR3)

ii) onlycoeff=yes ; a flag to confirm that we want only coefficients no ARF generations

iii) Mode of estimation [optional] : you may either choose pre-calculated table (A) or pre-estimated Kings profile (B)

(Both the results agree within the uncertainties)

iv) Name of events file with full path if not in current directory [if not entered with command line, it will ask you when calling this file in module]

v) Name of spectrum file (.pha file) to extract information about source regions...

vi) Name of reference SXT ARF file.. [if not entered with command line, it will ask you when calling this file in module]

The inclusion of events file and spectrum file are kept at higher priority than manual entry of source position and source region information, therefore if users provide these files along with --srcx=SRCX --srcy=SRCY --radius=R, these inputs are not used... These are only used if events file and spectrum are not entered while running this command..

This gives more flexibility to user to opt how to go...

vii) --vigcorrflag=yes [default is no] option is optional but recommended to use for any kind of spectral study...

viii) --pltflag=yes [default is yes] option makes a png file comparing the resulting ARF with original ARF..

The resulting ARFs thus generated are readily usable in xspec for spectral fitting ... No additional correction is needed for normalization of flux estimations...

Example : Case - 1 (estimating factors with no ARF outputs)

The radius input here is for multiple radii in one run of the script...Users, however, can give a single radius as input same way (i.e., --radius=15 or -r 15 for a circular region of 15 arcm radius)

(using bash script):-->

```
sxtARFModule --radius=20n15n10n5 --onlycoeff=yes --outstem=ARFTESTS1 --mode=b
```

or

```
sxtARFModule -r 20n15n10n5 --onlycoeff=yes -o ARFTESTS1 -m b
```

OR

(using python code) :-->

```
python sxt_ARFModule_v02.py --radius=20n15n10n5 --onlycoeff=0 --outstem=ARFTESTS1 --mode=b
```

or

```
python sxt_ARFModule_v02.py -r 20n15n10n5 --onlycoeff=0 -o ARFTESTS1 -m b
```

Users can avoid giving ARF, spectra and events file as input as with --onlycoeff=yes will force program to generate the numbers for source radius inputs

(If you are using source regions other than circle please estimate the total area of that region and calculate the radius of a circle with equal area and use this as input; i.e, circle equivalent of the used source region)

Case - 2 (estimating the factors and generate ARF for input spectra and based on input ARF...)

Since you are providing spectra and events file for the reference of source region information and hence the radius input will not be used (so can be omitted to simply the command line arguments). In this step --onlycoeff=no [--onlycoeff=0] is mandatory.

```
sxtARFModule --onlycoeff=0 --outstem=ARFTESTS1 --mode=b --evtfile=RX1856/RX1856_or15427_15444_merged_cl.evt
```

```
--sxtpha=RX1856/src10arcm_g0.pha --
sxtarf=/home/chandra/Dropbox/SXTCalibrationUpdates/RMFnARF/sxt_pc_excl00_v04_20190608.arf --vigcorrflag=yes
or
sxtARFModule --onlyceoff=0 -o ARFTESTS1 -m b -e RX1856/RX1856_or15427_15444_merged_cl.evt --
sxtpha=RX1856/src10arcm_g0.pha --
sxtarf=/home/chandra/Dropbox/SXTCalibrationUpdates/RMFnARF/sxt_pc_excl00_v04_20190608.arf --vigcorrflag=yes
```

OR

```
(using python code) :-->
python sxt_ARFModule_v02.py --radius=20n15n10n5 --onlyceoff=0 --outstem=ARFTESTS1 --mode=b --
evtfile=RX1856/RX1856_or15427_15444_merged_cl.evt --sxtpha=RX1856/src10arcm_g0.pha --
sxtarf=/home/chandra/Dropbox/SXTCalibrationUpdates/RMFnARF/sxt_pc_excl00_v04_20190608.arf --vigcorrflag=yes
or
python sxt_ARFModule_v02.py -r 20n15n10n5 --onlyceoff=0 -o ARFTESTS1 -m b -e
RX1856/RX1856_or15427_15444_merged_cl.evt --sxtpha=RX1856/src10arcm_g0.pha --
sxtarf=/home/chandra/Dropbox/SXTCalibrationUpdates/RMFnARF/sxt_pc_excl00_v04_20190608.arf --vigcorrflag=yes
```

If you have set --vigcorrflag=yes then two output ARFs will be generated one with Vignetting correction and other without it. Users are encouraged to use with proper Vignetting correction. This option is not default because of some other regions. Anyway this option is essential for any observations with SXT as not a primary instrument.

-----Requisites Modules or python libraries -----

* Important python libraries [Users should check whether these are installed or not]

- 1) astropy
- 2) numpy
- 3) scipy
- 4) lmfit
- 5) shutil
- 6) glob
- 7) matplotlib
- 8) pip [optional for Installation of python packages]
- 9) Kapteyn [optional; needed only if users want to use sxtEEFmaker for modelling the profile]

*please upgrade astropy, numpy and scipy if using old Installations...

```
astropy version : 2.0.12 or later
numpy version: 1.16.3 or later
For upgrading users can take follow instructions as below :
For pip users :
Download get-pip.py from https://bootstrap.pypa.io/get-pip.py
sudo python get-pip.py
sudo pip install pip --upgrade
sudo pip install --upgrade astropy
sudo pip install --upgrade numpy
In case you use both python-2 and python-3 (2:default)
sudo python3 get-pip.py
sudo pip install pip --upgrade
sudo pip install --upgrade astropy
sudo pip install --upgrade numpy
*check the version of your working astropy and numpy
python (or python3)
> import astropy
> print (astropy.version.version)
> print (numpy.version.version)
```

+++++SXTEEFMAKER Module help +++++

This distribution of modules include another python module which can be used to generate encircled energy profile (EEF) for any particular SXT observations. This module is basically of importance for POC related work, however, is for users interest in case one wants to study non-standard sources.

This takes events file and source position (centroid in Sky Pixels or detector pixels) Following summarizes the input details of this module :

This tool is also installed itself (added in your .bashrc file as

```
chandra@blazars:~$ sxteefmaker -h
```

```
=====
Running 'SXT EEF Maker' Tool
  Task: /home/chandra/auxpyscript/make_sxt_EEF_v02_py2.py
  Version: 02 Release Date: 2018-11-14
  Originally Developed By :

  Sunil Chandra,
  TIFR, Mumbai, 15 January, 2017
  Updated on 03 July, 2019
+++++
```

```
Description: This tool makes Encircled Energy Fraction (EEF) profile for sources by using merged clean events
files as input. The profile is shown to have radii
containing 90% and 60% of total incoming photons. This version provides an option to correct profiles for
background [Default].
+++++
=====
```

```
Usage: make_sxt_EEF_v02_py2.py [options]
```

```
Options:
```

```
-h, --help          show this help message and exit
-e EVTFILE, --evtfile=EVTFILE
                    Input Events file Name; The most important input of
                    script give with full path if not in same directory
-r RA, --ra=RA      Input RA in Sky pixels Note that open your events
                    file with ds9 and note the position of centriod for
                    the source in pixels
-d DEC, --dec=DEC   Input DEC in Sky pixels
-o OUTFILE, --outfile=OUTFILE
                    Stem for output files
-x MINR, --min=MINR Radius of inner circle for annulii (in Sky pixel);
                    Default value is 8 pix (~0.5 arcmin)
-y MAXR, --max=MAXR Radius of outer circle for annulii (in Sky pixel);
                    Default value is 328 pix (~22.5 arcmin)
-s STEPS, --steps=STEPS
                    Steps size to decide number of data points in outpur
                    profile [N = int((r_out - r_in)/stepsize)]; default
                    is 8 pixel
-p PROCESS, --process=PROCESS
                    Plot type: 'plot' or 'both'; Whether you wish to make
                    plot only or make tables as well; You may use any of
                    ["BOTH", "AUTO", "B", "A",
                    "PROCESS"] for default way
                    [making profile and plot] and any of ["PLOT", "PL",
                    "P"] for plot-only option if you already have tables
                    with you
--estring=ESTRING   String for energy selection; the acceptable input
                    format is 0p3to7p0 (for 0.3-7.0 keV) Default is
                    0.3-7.0 keV
-g GRADE, --grade=GRADE
                    String for grade selections in the format 0-12
                    Default is all grades 0-12
-t SIMEEF, --simeef=SIMEEF
                    Filename with full path for EEF from simulation;
                    Needed only for POC related bussiness, you make ignore
                    it, Default is None, which mean no
-l IPACFILE, --ipacfile=IPACFILE
                    IPAC file name with data; needed for 'plot' option for
                    process, Default = None
--coordtyp=COORDTYP
                    Input Coordinate Type, .i.e., Detector Coordinates or
                    Sky Coordinates...Default is Sky The input options
                    are ['Sky', 'sky', 'SKY', 's', 'S', 1, True] and/or
                    ['RAW', 'raw', 'Raw', 'R', 'r', 0] Default is 'RAW'
--bkgcorr=BKGCORR   The flag for back-ground correction, Default options
                    is 'yes'
--NrefStyl=NREFSTYL The flag for using method to normalize the profile,
                    options are either use total counts from largest safe
                    radius of source extraction or the counts from a
                    source region of 20 arcmin [extra-polated, in cases
                    20arcmin is not possible] to use. Input options =
                    ['extrapolate', 'ff', 'full', '20arcmin', 'auto', 1] for
                    using 20arcmin anything else except ['both', 'BOTH',
```

'debug', 2] will result in using largest possible safe radius for normalization of profiles; You may also chose for any of ['kings', 'fit', 'model', 0]

--normradius=NORMRADIUS

The radius [in arcm] what script should use in case user is confident with source PSF

--logfile=LOGFILE

The Name of logfile ..Default is

'LogFile_DateTimeFormat.log'

--usepoly3flag=USEPOLY3FLAG

The Flag whether polynomial of order 3 should be used for fitting or not.. Default = 'no'