Event reconstruction software
for Athena X-IFU

Maite Ceballos, Bea Cobo
+ e2e team @ IRAP, Bamberg, SRON, GSFC
SIMULATION & RECONSTRUCTION

**SIMULATION**

Physical Parameters:
- $T$, $C_e$, $G_b$, $n$, $\alpha$, $\beta$, $R_0$, $I_0$, $T_b$, ...

**SIXTE**

**RECONSTRUCTION**

Reconstruction Parameters:
- Method, record length, Filter/noise files,...

http://www.sternwarte.uni-erlangen.de/research/sixte/index.php
Adjusted Derivative method:
- Take derivative of TES stream (+ low pass filter if noise requests)
- Remove already detected pulses to discover missing ones
Event Detection (triggering) → Event Grading → Energy Calculation

Energy Resolution FWHM [eV] – SPA – (1keV)

Energy biased by previous pulse
Energy resolution degraded by the loss of available data due to next pulse

Read-out signal [AU]

Time [ms]

T₂ (samples)

T₁ (samples)

High Res
Mid Res
Low Res
Rejected Secondaries

15.1
11.9
9.3
7.3
7.3
5.7
4.5
2.8
2.2
1.7

1000
2000
1000
1000
• Pulses are scaled versions of a single shape: Response of detector is linear (or energy-dependent filter interpolation)
• Noise is stationary

(Szymkowiak 1993, Boyce et al. 1999)

\[
\text{Data} \quad D(f) = H \times S(f)
\]

\[
\text{Minimize} \quad \chi^2 = \sum \frac{[D(f) - H \times S(f)]^2}{\text{NOISE}^2(f)}
\]

\[
\text{Energy} \sim \sum D(t) \text{OptFil}(t)
\]

\[
\text{OptFil}(f) = \frac{S^*(f)}{\langle |N(f)|^2 \rangle}
\]

Fig. 1 Plots showing (top to bottom) average pulse shape, power spectrum of the noise, and the optimal filtering template.
Accounts for noise non-stationarity & detector non-linearity

- Model template ($T$) + covariance matrix (deviations from model) + weight matrix ($W$) (inverse of covariance matrix)

Minimize $\chi^2 = (\text{Data} - T)W(\text{Data} - T)$

Energy = $f(E_\alpha, E_\beta, U, M_\alpha, M_\beta, W_\alpha, W_\beta)$

$\alpha, \beta$: calibration points that straddle the unknown signal $U$ (densely spaced calibration energies)
Event Detection (triggering) → Event Grading → Energy Calculation

Optimal Filter after transforming signal $I_{\text{TES}}$ to $R_{\text{TES}}$:
Removes non-linearity due to the bias circuit + more uniform noise (?)

( Bandler et al, 2004, 2006)
Event Detection (triggering) → Event Grading → Energy Calculation

Covariance Matrix (noise)

Covariance Matrix (data)

PCA

(Left to Right: Busch et al, 2016, Yan et al, 2016)

(Busch et al, 2016)
Event Detection (triggering) → Event Grading → Energy Calculation

PCAs

Covariance Matrix (noise)

Covariance Matrix (data)

(Busch et al, 2016) Yan et al 2016

(Busch et al, 2016) Yan et al, 2016

$K_{\alpha}$ $K_{\beta}$
Event Detection (triggering)

Energy Calculation

Event Grading

PCA

(Busch et al, 2016 Yan et al 2016)

Covariance Matrix (noise)

(Kα, Kβ)

Center: 1589.5; 5.9 keV
FWHM: 50.0; 0.2 keV

Center: 1747.2; 6.5 keV
FWHM: 51.6; 0.2 keV

(Busch et al, 2016)
Reconstruction: Optimal Filter

File Edit Tools Help

<table>
<thead>
<tr>
<th>Select</th>
<th>ID</th>
<th>10s</th>
<th>10 keV</th>
<th>GRADE1</th>
<th>GRADE2</th>
</tr>
</thead>
<tbody>
<tr>
<td>All</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Invert</th>
<th>Modify</th>
<th>Modify</th>
<th>Modify</th>
<th>Modify</th>
</tr>
</thead>
<tbody>
<tr>
<td>Invert</td>
<td>Modify</td>
<td>Modify</td>
<td>Modify</td>
<td>Modify</td>
</tr>
</tbody>
</table>

| 1      | 2.615200000000000-02 | 9.95953174243-01 | 238 | 1024 |
| 2      | 3.7684000000000000-02 | 1.0004350266200+00 | 1024 | 0    |
| 3      | 4.5766000000000000-02 | 9.35922463220-01 | 1024 | 246  |
| 4      | 6.5760000000000000-02 | 1.0001887056370+00 | 238 | 1024 |
| 5      | 6.7283000000000000-02 | 1.0003667521550+00 | 1024 | 0    |
| 6      | 7.5411200000000000-02 | 9.8687152897940-01 | 1024 | 246  |
| 7      | 9.5404800000000000-02 | 1.0000562786110+00 | 238 | 1024 |
| 8      | 9.6321600000000000-02 | 1.0015063507230+00 | 1024 | 0    |
| 9      | 1.0505600000000000-01 | 9.957556593590-01 | 1024 | 246  |
| 10     | 1.2504360000000000-01 | 9.514948007630-01 | 238 | 1024 |
| 11     | 1.2652200000000000-01 | 9.528294164550-01 | 1024 | 0    |
| 12     | 1.3470080000000000-01 | 1.0003645213860+00 | 1024 | 246  |
| 13     | 1.5663440000000000-01 | 1.0007679353210+00 | 238 | 1024 |
| 14     | 1.5621760000000000-01 | 1.0000361019910+00 | 1024 | 0    |
| 15     | 1.6430560000000000-01 | 1.0000761120320+00 | 1024 | 246  |
| 16     | 1.6493200000000000-01 | 1.00073681502240+00 | 238 | 1024 |
| 17     | 1.8586420000000000-01 | 1.0004172054540+00 | 1024 | 0    |
| 18     | 1.9399040000000000-01 | 9.537220532120-01 | 1024 | 246  |
| 19     | 2.1398400000000000-01 | 1.0003116315440+00 | 238 | 1024 |
| 20     | 2.1550720000000000-01 | 1.0007056357620+00 | 1024 | 0    |
SIXTE - Simulation of X-ray Telescopes

SIXTE is a software package for X-Ray telescope observation simulations developed at ECAF under the leadership of Christian Schmid. It allows to undertake instrument performance analyses and to produce simulated event files for mission- and analysis studies.

The software strives to find a compromise between exactness of the simulation and speed. For many cases, by using calibration files such as the PSF, RMF and ARF, efficient simulations are possible at comparably high speed, even though they include nonlinear effects such as pileup. Setups for some current and future missions such as XMM-Newton or Athena are included in the package, others can be added by the user with relatively little effort through specifying the main instrument characteristics in a flexible, human-readable XML-based format.

Properties of X-ray sources to be simulated are described in a detector-independent format, i.e., the same input can be used for simulating observations with all available instruments, and the same input can also be used for simulations with the SIMX simulator. The input files can be easily generated from standard data such as XSPEC spectral models or FITS images with tools provided with the SIXT distribution. The input data scale well from single point sources up to very complicated setups. For example, for ATHENA we have simulated observations of the galactic center based on the Chandra input catalogues and images of the diffuse emission, while for eROSITA we regularly perform simulations of the whole sky using several million time-variable point sources.

A full description of the file format is given in the SIMPUT (SIMulation inPUT) FITS specification document, file format. It lists the basic parameters of the sources such as their flux in a reference band, their positions in a table and can also reference images, spectra and lightcurves.
SIRENA documentation

Contents:

- SIRENA description
  - Purpose
  - Files
    - Auxiliary Files
    - Input Files
    - Output Files
  - Reconstruction Process
    - Event Detection
    - Event Grading
    - Event Energy Determination: methods
  - Examples
- SIRENA Tools CLI
  - gennoisespec
  - tesreconstruction
  - tessim
  - tesconstpileup
  - streamtotriggers
- SIRENA functions
- References
Event Energy Determination: methods

Once the input events have been detected and graded, their energy content can be determined. Currently all the events (independently of their grade) are processed with the same reconstruction method, but in the future, a different approach could be taken, for example simplifying the reconstruction for the lowest resolution events.

The SIRENA input parameter that controls the reconstruction method applied is `EnergyMethod` that should take values of `OPTFILT` for Optimal Filtering in Current space, `WEIGHT` for Covariance Matrices, `WEIGHTN` for first order approach of Covariance matrices method and `I2R` or `I2RBIS` for Optimal Filtering implementation in (quasi)Resistance space.

Optimal Filtering

This is the baseline standard technique commonly used to process microcalorimeter data streams. It relies on two main assumptions. Firstly, the detector response is linear; that is, the pulse shapes are identical regardless of their energy and thus, the pulse amplitude is the scaling factor from one pulse to another [Boyce+99], [Szymkowski+93].

In the frequency domain (as noise can be frequency dependent), the raw data can be expressed as $P(f) = E \cdot S(f) + N(f)$, where $S(f)$ is the normalized model pulse shape, $N(f)$ is the noise and $E$ is the scalar amplitude for the photon energy.

The second assumption is that the noise is stationary, i.e. it does not vary with time. The amplitude of each pulse can then be estimated by minimizing (weighted least-squares sense) the difference between the noisy data and the model pulse shape, being the $\chi^2$ condition to be minimized:

$$\chi^2 = \int \frac{(P(f) - E \cdot S(f))^2}{|N(f)|^2} df$$

In the time domain, the amplitude is the best weighted (optimally filtered) sum of the values in the pulse.
The `tesreconstruction` tool is a wrapper to perform the energy reconstruction of the photon events by means of two different implementations: `Rmethod=PP` runs the branch developed by Philippe Peille and `Rmethod=SIRENA` runs the SIRENA code in this documentation.

SIRENA code takes a FITS input file of data, optionally performs the detection of the events, then grades them and finally reconstructs their energy following the algorithm selected by the user in the input command line of `tesreconstruction`.

The input data should be a FITS file with the data splitted into records.

To run SIRENA implementation, the user must supply the following input parameters (see Event Energy Determination: methods for a detailed description in the context of the reconstruction methods to which they apply):

```
Rmethod=< PP | SIRENA>

Reconstruction method (PP or SIRENA)

RecordFile=<str>

Input record FITS file

TesEventFile=<str>

Output event list file

PulseLength=<samples>

Pulse length in samples

EventListSize=<str>
```
SIRENA functions

Search functions by name at Index.

int addFirstRow
(int reconstructInitSIRENA*, reconstruct_init; fitsfile** inLibObject, double samprate, int runForB0val, gsl_vector* E, gsl_vector* PHEIGHT, gsl_matrix* PULSE, gsl_matrix* PULSE0, gsl_matrix* MF, gsl_matrix* MFB0, gsl_matrix* COVAR, gsl_matrix* WEIGHT)

Located in file: tasksSIRENA.cpp

This function writes the first row of the library (without intermediate AB-related values, because it would be necessary to have at least two rows=energies in the library)

Members/Variables

ReconstructInitSIRENA** reconstruct_init

Member of ReconstructInitSIRENA structure to initialize the reconstruction parameters (pointer and values). In particular, this function uses mode and noise_spectrum in order to run calculus_optimalFilter().

fitsfile** inLibObject

FITS Object containing information of the library FITS file

double samprate
Looking ahead...

End of Phase A: Q1 2017 (TBC)
Looking ahead...

- Work in progress... (launch 2028)
  Code optimization
- A lot of technical issues needs implementation/correction: gain drift, DRE effects, multi-pixel issues (crosstalk)
  Code optimization
- Currently testing reconstruction algorithms (performance & HW costs) + DRE interfaces
  Code optimization
- To be implemented in real (flight) hardware
THANKS
<table>
<thead>
<tr>
<th>Parameter</th>
<th>Requirements</th>
</tr>
</thead>
<tbody>
<tr>
<td>Energy range</td>
<td>0.3-12 keV</td>
</tr>
<tr>
<td>Energy resolution: E &lt; 7 keV</td>
<td>2.5 eV</td>
</tr>
<tr>
<td>Energy resolution: E &gt; 7 keV</td>
<td>( E/\Delta E = 2800 )</td>
</tr>
<tr>
<td>Field of View</td>
<td>5’ (diameter)</td>
</tr>
<tr>
<td>Detector quantum efficiency @ 1 keV</td>
<td>&gt;60%</td>
</tr>
<tr>
<td>Detector quantum efficiency @ 7 keV</td>
<td>&gt;70%</td>
</tr>
<tr>
<td>Gain error (RMS)</td>
<td>0.4 eV</td>
</tr>
<tr>
<td>Count rate capability – faint source</td>
<td>1 mCrab (&gt;80% high-resolution events)</td>
</tr>
<tr>
<td>Count rate capability – bright source</td>
<td>1 Crab (&gt;30% low-resolution events)</td>
</tr>
<tr>
<td>Time resolution</td>
<td>10 ( \mu ) s</td>
</tr>
<tr>
<td>Non X-ray background</td>
<td>&lt; 5 ( 10^{-3} ) counts/s/cm(^2)/keV</td>
</tr>
</tbody>
</table>

Table 2: Key performance requirements for the Athena+ X-ray Integral Field Unit