Swarm Tools

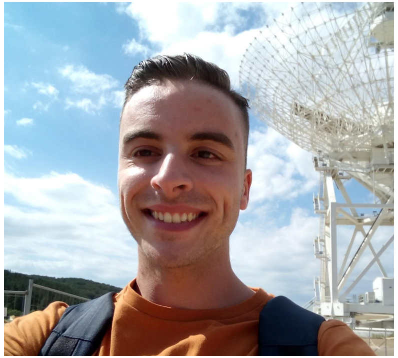
Tools for data simulation and analysis for HERMES mission

Alessandro Riggio, Andrea Sanna

The team (in addition to the usual suspects)

Alessio Anitra
(testing)

Arianna Manca
(software development,
testing)



Carlo Cabras
(computer science)

Nicola Deiosso
(simulations)

GTools Library (provisory name)

In-house C library

Provides an high level, mission-agnostic interface to FITSIO event files, specially tailored for rapid timing analysis.

Capable of:

- data selection by time intervals, energy
- timing analysis (epoch folding, epoch folding search, barycentering, FFT-like analysis)
- Lightcurve creation and analysis

Swarm Tools

A set of software tools to simulate and analyze datasets from a swarm of detectors (like HERMES-SP)

Simulator:
swarm_sim

Analysis:
swarm_pos
swarm_to_xspec

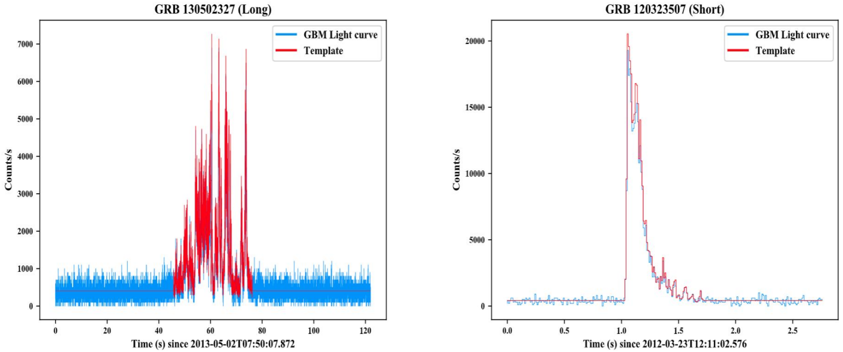
Simulator: general characteristics

What can we simulate?

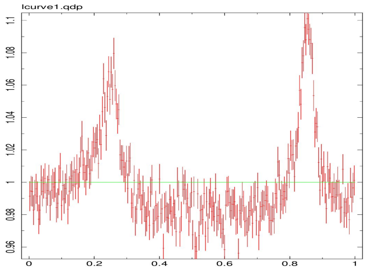
Periodic and aperiodic sources, f.e. a signal from a GRB or a pulsar, seen by an arbitrary number of detectors arbitrarily placed and oriented in space.

Takes into account both temporal and spectral characteristics of the source

GRBs

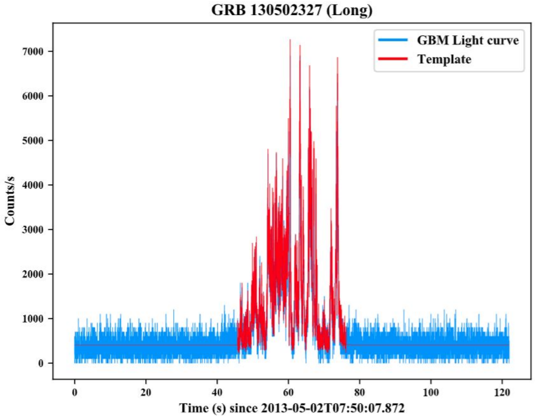


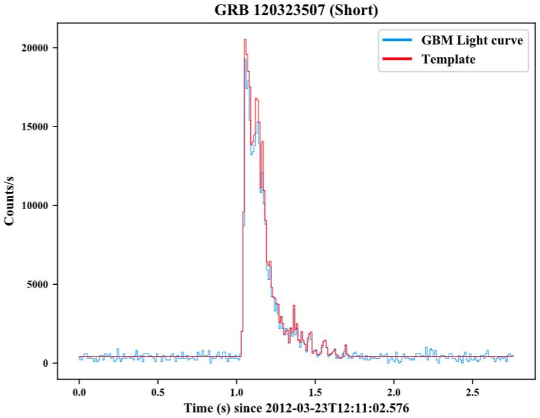
Pulsars



Simulator: timing characteristics

Lightcurve is simulated using a template. Input can be a synthetic sum of FRSD shots, event data or a binned lightcurve


GRB 130502327 (Long)


GRB 120323507 (Short)

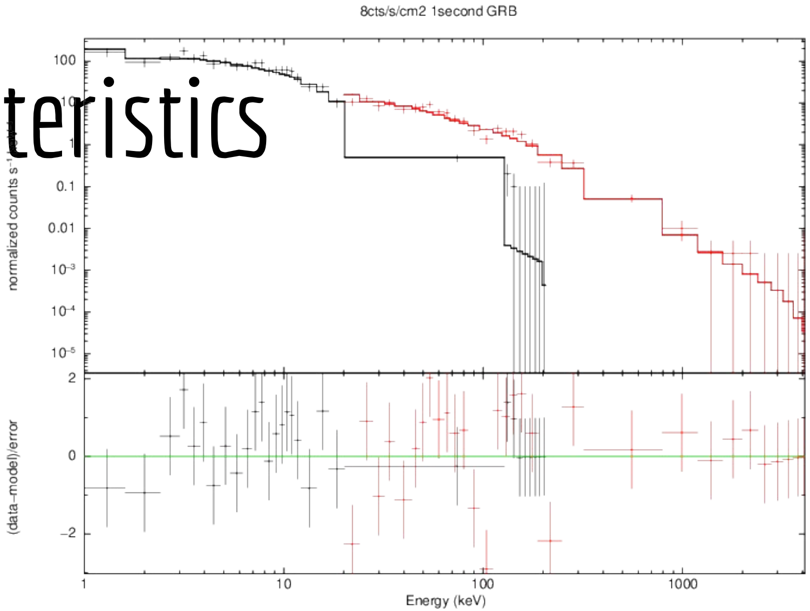
If the template is noisy on short timescales, it is smoothed to prevent artifacts due to statistical fluctuations

Simulator: timing characteristics (II)

The detectors can be placed anywhere in space and with any pointing. Position and pointing is specified (at the moment) via an input file

- Arbitrary number of detectors
- Arbitrary time resolution
- Arbitrary length of the simulation (RAM limited)

The simulator: spectral characteristics



Events generated using the Band function for the GRB (Band+93) and the CXB function for the background (Gruber+99).

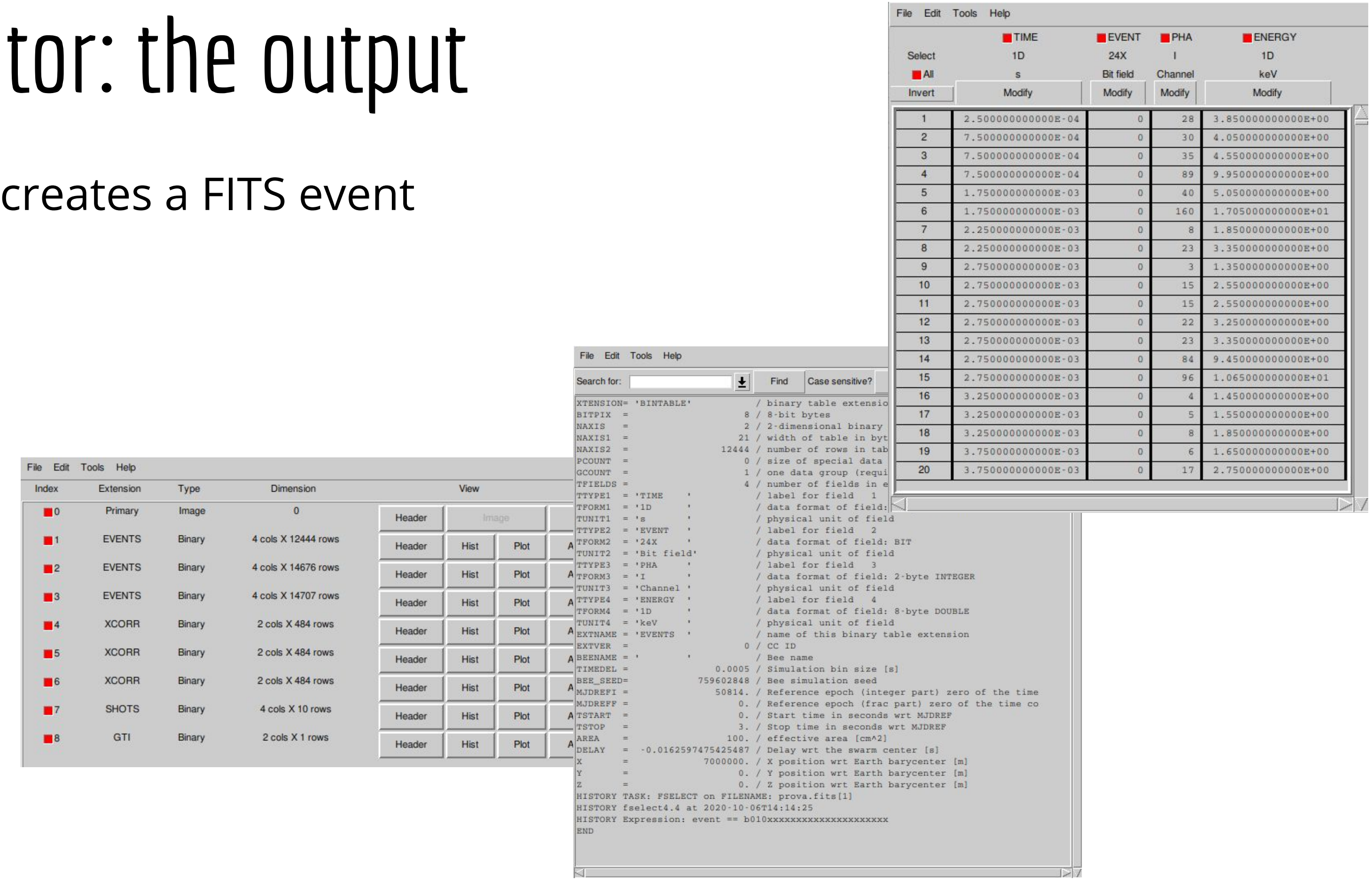
The addition of other models is very simple.

The effective area (ARF) is used to take into account the instrument response.

The *swarm_to_xspec* tools produces XSpec compliant files.

Simulator: the output

Simulator creates a FITS event



Index	Extension	Type	Dimension	View
🟥 0	Primary	Image	0	Header Image
🟥 1	EVENTS	Binary	4 cols X 12444 rows	Header Hist Plot
🟥 2	EVENTS	Binary	4 cols X 14676 rows	Header Hist Plot
🟥 3	EVENTS	Binary	4 cols X 14707 rows	Header Hist Plot
🟥 4	XCORR	Binary	2 cols X 484 rows	Header Hist Plot
🟥 5	XCORR	Binary	2 cols X 484 rows	Header Hist Plot
🟥 6	XCORR	Binary	2 cols X 484 rows	Header Hist Plot
🟥 7	SHOTS	Binary	4 cols X 10 rows	Header Hist Plot
🟥 8	GTI	Binary	2 cols X 1 rows	Header Hist Plot

```
XTENSION= 'BINTABLE'           / binary table extensio
BITPIX  =                    8 / 8-bit bytes
NAXIS   =                    2 / 2-dimensional binary
NAXIS1  =                   21 / width of table in byt
NAXIS2  =                12444 / number of rows in tab
PCOUNT  =                    0 / size of special data
GCOUNT  =                    1 / one data group (requi
TFIELDS =                    4 / number of fields in e
TTYPE1  = 'TIME    '           / label for field   1
TFORM1  = '1D      '           / data format of field:
TUNIT1  = 's       '           / physical unit of field
TTYPE2  = 'EVENT   '           / label for field   2
TFORM2  = '24X     '           / data format of field: BIT
TUNIT2  = 'Bit field'          / physical unit of field
TTYPE3  = 'PHA     '           / label for field   3
TFORM3  = 'I       '           / data format of field: 2-byte INTEGER
TUNIT3  = 'Channel '           / physical unit of field
TTYPE4  = 'ENERGY  '           / label for field   4
TFORM4  = '1D      '           / data format of field: 8-byte DOUBLE
TUNIT4  = 'keV     '           / physical unit of field
EXTNAME = 'EVENTS  '           / name of this binary table extension
EXTVER  =                    0 / CC ID
BEENAME = '        '           / Bee name
TIMEDEL =               0.0005 / Simulation bin size [s]
BEE_SEED=            759602848 / Bee simulation seed
MJDREFI =               50814. / Reference epoch (integer part) zero of the time
MJDREFF =                   0. / Reference epoch (frac part) zero of the time co
TSTART  =                   0. / Start time in seconds wrt MJDREF
TSTOP   =                   3. / Stop time in seconds wrt MJDREF
AREA    =                 100. / effective area [cm^2]
DELAY   =  -0.0162597475425487 / Delay wrt the swarm center [s]
X       =             7000000. / X position wrt Earth barycenter [m]
Y       =                   0. / Y position wrt Earth barycenter [m]
Z       =                   0. / Z position wrt Earth barycenter [m]
HISTORY TASK: FSELECT on FILENAME: prova.fits[1]
HISTORY fselect4.4 at 2020-10-06T14:14:25
HISTORY Expression: event == b010xxxxxxxxxxxxxxxxxxxxx
END
```

Select	🟥 TIME	🟥 EVENT	🟥 PHA	🟥 ENERGY
	1D	24X	I	1D
🟥 All	s	Bit field	Channel	keV
Invert	Modify	Modify	Modify	Modify
1	2.500000000000E-04	0	28	3.850000000000E+00
2	7.500000000000E-04	0	30	4.050000000000E+00
3	7.500000000000E-04	0	35	4.550000000000E+00
4	7.500000000000E-04	0	89	9.950000000000E+00
5	1.750000000000E-03	0	40	5.050000000000E+00
6	1.750000000000E-03	0	160	1.705000000000E+01
7	2.250000000000E-03	0	8	1.850000000000E+00
8	2.250000000000E-03	0	23	3.350000000000E+00
9	2.750000000000E-03	0	3	1.350000000000E+00
10	2.750000000000E-03	0	15	2.550000000000E+00
11	2.750000000000E-03	0	15	2.550000000000E+00
12	2.750000000000E-03	0	22	3.250000000000E+00
13	2.750000000000E-03	0	23	3.350000000000E+00
14	2.750000000000E-03	0	84	9.450000000000E+00
15	2.750000000000E-03	0	96	1.065000000000E+01
16	3.250000000000E-03	0	4	1.450000000000E+00
17	3.250000000000E-03	0	5	1.550000000000E+00
18	3.250000000000E-03	0	8	1.850000000000E+00
19	3.750000000000E-03	0	6	1.650000000000E+00
20	3.750000000000E-03	0	17	2.750000000000E+00

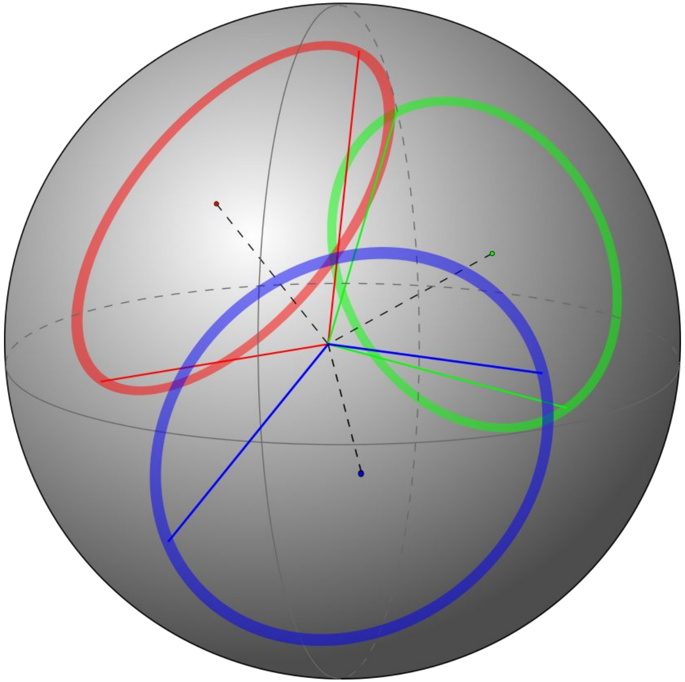
Positioning the source

Two methods implemented so far:

- Triangulation via measuring delays (see e.g. Pal’shin+13)
- Kevin Hurley method (Hurley 2020)

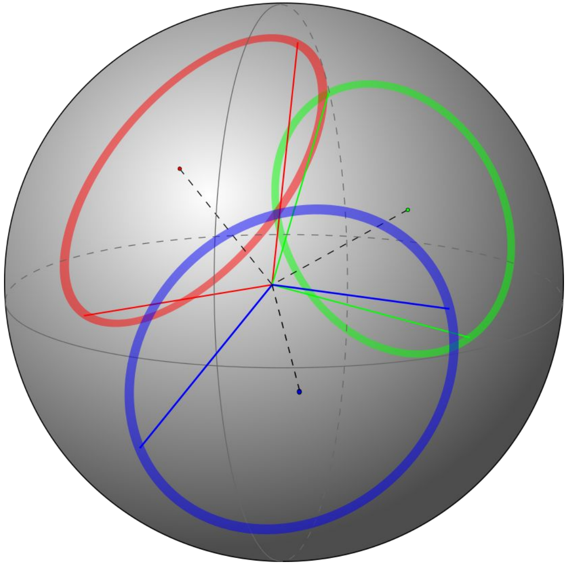
Possible future implementations

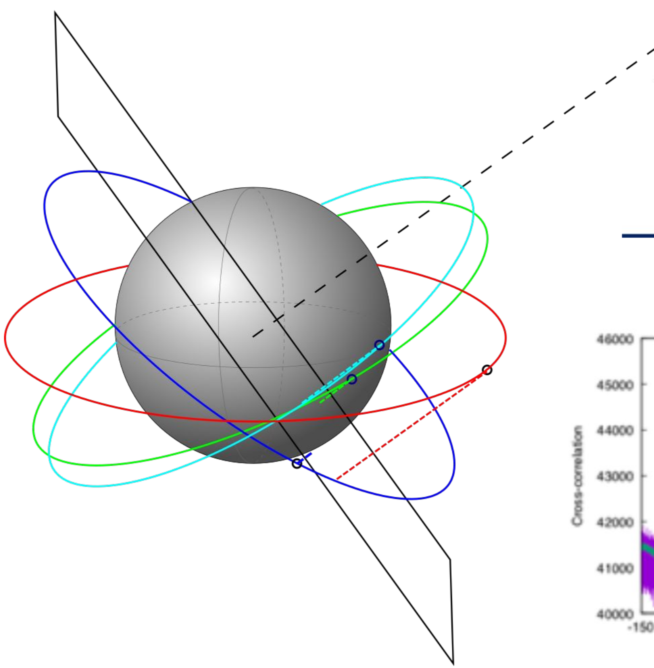
- Nazgul (Burgess+20)

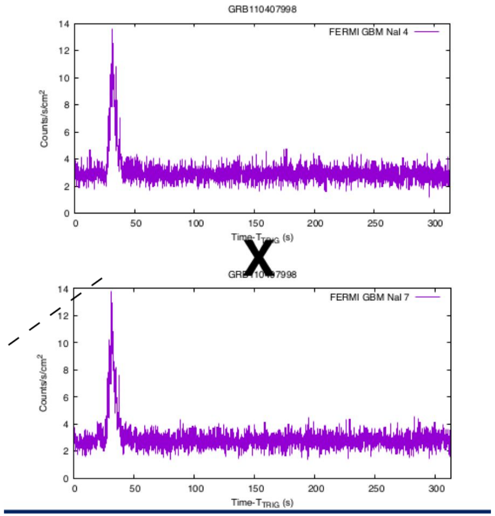


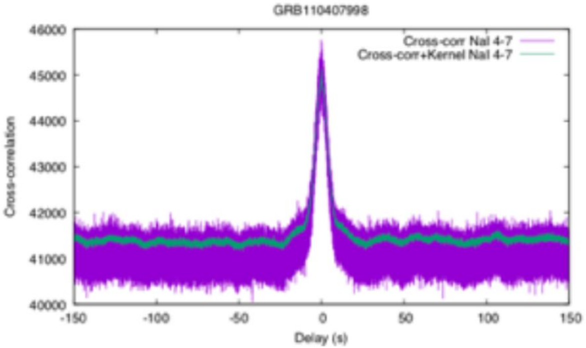
Positioning the source

$$\chi^2(\hat{\boldsymbol{d}}) = \sum_{i=0}^{n-2} \sum_{j=i+1}^{n-1} \frac{(\Delta\tau_{ij} - \Delta t_{ij}(\hat{\boldsymbol{d}}))^2}{\delta\Delta\tau_{ij}^2 + \delta\Delta t_{ij}^2},$$

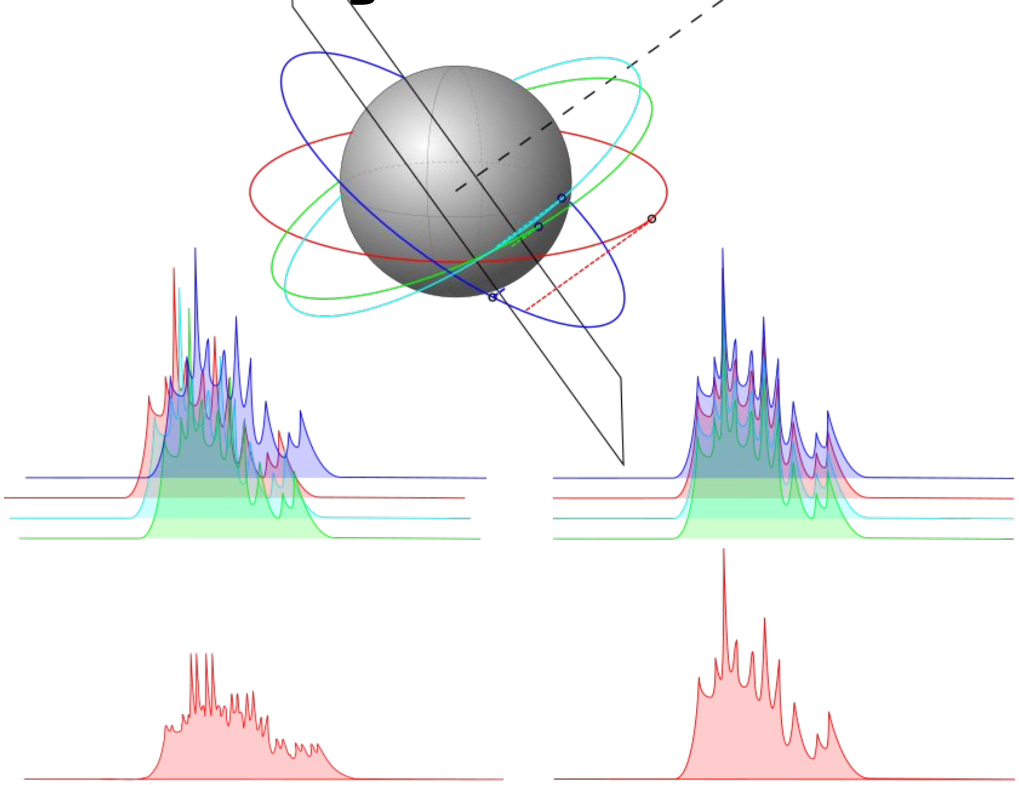


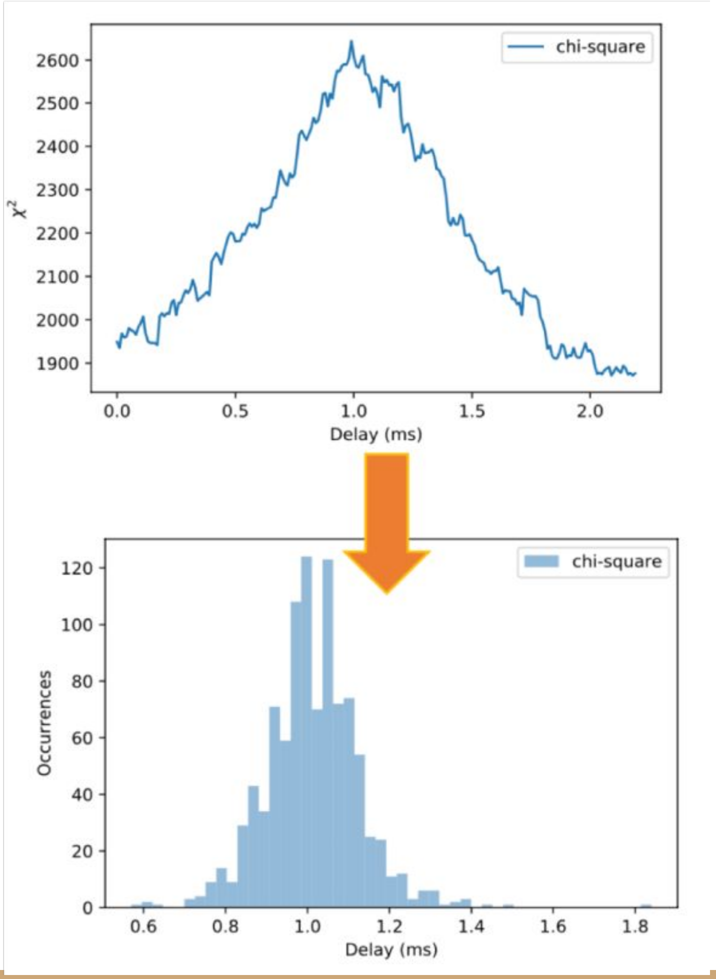






Positioning the source





Current status of the software

The software is in an advanced stage and nearly ready for a beta release.

At the moment we are concentrated in

- testing procedures
- Hurley method statistical properties
- Developing a GUI (too much input parameters!)

A lot of simulations!

To Do List (Not in a specific order)

- Find a robust algorithm for automatic cross-correlation fitting and source position determination (specially confidence regions).
- Response matrix-aware simulations (not important at the expected HERMES SP count-rates.)
- Satellite position and pointing dependent background, Earth albedo (see f.e. Campana+13)
- Dead time pruning of generated event lists.
- Moving satellites (not important for short GRBs, but relevant for long GRBs and mostly pulsar). We need to know the format of the orbit files.
- Any suggestions?

Technical stuff

Developed using standard Posix C99. Working on Linux (tested on Debian-like distributions) and Mac OS X

...

The code in numbers

Lines of Code: 7443

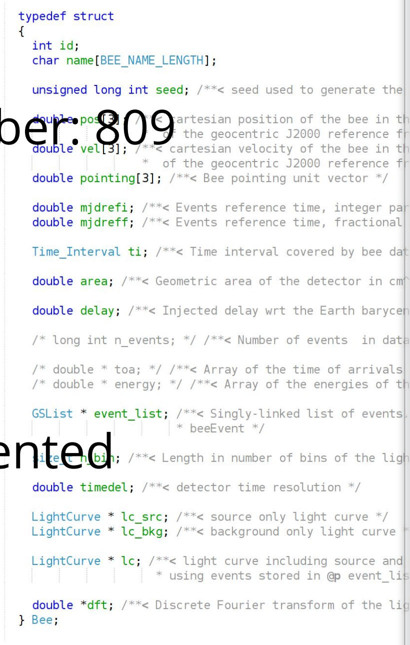
Lines of Comment: 2821

McCabe's Cyclomatic Number: 809

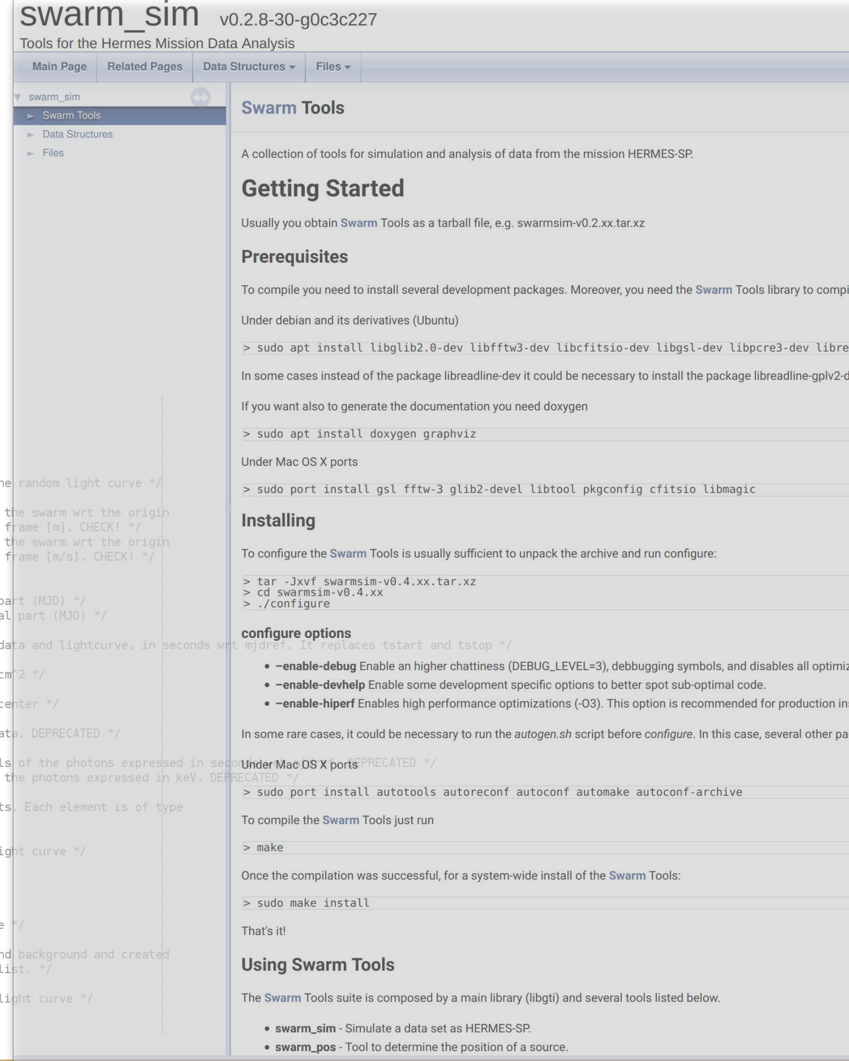
LOC/COM: 2.6

MVG/COM: 0.29

The source code is documented using Doxygen



```
typedef struct
{
  int id;
  char name[BEE_NAME_LENGTH];

  unsigned long int seed; /**< seed used to generate the random light curve */

  double pos[3]; /**< cartesian position of the bee in the swarm wrt the origin
                  *  of the geocentric J2000 reference frame [m]. CHECK! */
  double vel[3]; /**< cartesian velocity of the bee in the swarm wrt the origin
                  *  of the geocentric J2000 reference frame [m/s]. CHECK! */
  double pointing[3]; /**< Bee pointing unit vector */

  double mjdrefi; /**< Events reference time, integer part (MJD) */
  double mjdreff; /**< Events reference time, fractional part (MJD) */

  Time_Interval ti; /**< Time interval covered by bee data and lightcurve. in seconds wrt mjdref. It replaces tstart and tstop */

  double area; /**< Geometric area of the detector in cm^2 */

  double delay; /**< Injected delay wrt the Earth barycenter */

  /* long int n_events; */ /**< Number of events  in data. DEPRECATED */

  /* double * toa; */ /**< Array of the time of arrivals of the photons expressed in seconds wrt mjdref. DEPRECATED */
  /* double * energy; */ /**< Array of the energies of the photons expressed in keV. DEPRECATED */

  GSList * event_list; /**< Singly-linked list of events. Each element is of type
                        * beeEvent */

  size_t n_bin; /**< Length in number of bins of the light curve */

  double timedel; /**< detector time resolution */

  LightCurve * lc_src; /**< source only light curve */
  LightCurve * lc_bkg; /**< background only light curve */

  LightCurve * lc; /**< light curve including source and background and created
                    * using events stored in @p event_list. */

  double *dft; /**< Discrete Fourier transform of the light curve */
} Bee;
```



Swarm Tools

A collection of tools for simulation and analysis of data from the mission HERMES-SP.

Getting Started

Usually you obtain Swarm Tools as a tarball file, e.g. swarmsim-v0.2.xx.tar.xz

Prerequisites

To compile you need to install several development packages. Moreover, you need the Swarm Tools library to compi

Under debian and its derivatives (Ubuntu)

```
> sudo apt install libglib2.0-dev libfftw3-dev libcfitsio-dev libgsl-dev libpcre3-dev libre
```

In some cases instead of the package libreadline-dev it could be necessary to install the package libreadline-gplv2-d

If you want also to generate the documentation you need doxygen

```
> sudo apt install doxygen graphviz
```

Under Mac OS X ports

```
> sudo port install gsl fftw-3 glib2-devel libtool pkgconfig cfitsio libmagic
```

Installing

To configure the Swarm Tools is usually sufficient to unpack the archive and run configure:

```
> tar -Jxvf swarmsim-v0.4.xx.tar.xz
> cd swarmsim-v0.4.xx
> ./configure
```

configure options

- **−enable-debug** Enable an higher chattiness (DEBUG_LEVEL=3), debbugging symbols, and disables all optimiz
- **−enable-devhelp** Enable some development specific options to better spot sub-optimal code.
- **−enable-hiperf** Enables high performance optimizations (-O3). This option is recommended for production in

In some rare cases, it could be necessary to run the *autogen.sh* script before *configure*. In this case, several other pa

Under Mac OS X ports

```
> sudo port install autotools autoreconf autoconf automake autoconf-archive
```

To compile the Swarm Tools just run

```
> make
```

Once the compilation was successful, for a system-wide install of the Swarm Tools:

```
> sudo make install
```

That's it!

Using Swarm Tools

The Swarm Tools suite is composed by a main library (libgti) and several tools listed below.

- **swarm_sim** - Simulate a data set as HERMES-SP.
- **swarm_pos** - Tool to determine the position of a source.