

# gtobsSim Results

Review by *Tom Stephens*

Unfortunately, due to preparation at the GSSC for the upcoming GLAST Ground Readiness Test #1, I didn't have as much time to work on this as I would have liked. I was able to run the code in a bunch of different scenarios with different combinations of sources. I haven't really had the time to look at the details of the results and so mainly have comments on the interface and documentation.

## Timing Results

I tried out most of the models I knew about (the ones in the ObsSim documentation and on Seth's Confluence page and a few gleaned from example XML files). I haven't had the time to really look into PulsarSpectrum yet and I didn't run the GRB sources.

The table below lists the execution times for runs of gtobssim with various source types, numbers and combinations. All of these runs were for 1 day of simulated time, without any diffuse model and using the TEST response.

Source Type	N_sources	CPU secs	N_sources	CPU sec	Ratio
Power Law	1	1.16	5	4.73	4.08
broken Power Law	1	1.79	5	1.56	0.87
TransientSource	1	3.21	5	31.18	9.71
TransientTemplate	1	9.99	5	90.06	9.02
Periodic	1	27.83	5	375.49	13.49
Pulsar	1	15.91	5	84.52	5.31
Gaussian	1	1.12	5	3.16	2.82
SpectralTransient	1	1.18	5	3.5	2.97
all of the above	1 each	160.82	5 each	735.08	4.57

While not strictly an apples-to-apples comparison as all the sources were randomly generated and unique to each run, the above table give a qualitative feel for the relative timings for the various sources and how they scale and interact. There are a few (TransientSource, TransientTemplate and Periodic) that don't seem to scale well at all and adding multiple sources of the same type seems to really bog down the system.

Also of note is that it takes much longer (by a factor of 2-3) to simulate a bunch of different source together than it does to generate them individually.

I also ran all the above models with the Galactic diffuse and extragalactic diffuse emission as well. I haven't had time to look at those runs in any detail but the basic effect was to add ~200 seconds to the execution times which was how long it took to run just the diffuse models by themselves.

## MapSource and MapSpectrum

There were two sets models of the diffuse galactic background in the default model XML files using two different map sources, MapSource and MapSpectrum. I made 1 day runs of both source types and found the MapSource source to be about 2x faster than the MapSpectrum source.

## Output Problems

In looking at the output files (and trying to ingest the ones Seth made into the database system) I noticed a few problems. Classification in to major or minor issues is purely my opinion.

### Major Issues

1. TSTART, TSTOP, DATE-OBS and DATE-END do not correspond to the data in the file. Rather they are just the start and stop time of the entire simulation. While this may not be an issue at the moment for any of the science tools, it did have an effect on ingesting the data into the data server. One of the things the data server does is divide it's internal files (and some of the files returned to the user) by time to manage the file size. To do this it compares the TSTART value of the existing file to the TSTART value of the new data. When the difference reaches a certain size, the current file is closed out and a new file is started. If the TSTART keyword is the same in all the files, it will never create the new files. This effect can be seen in the data in the data server for the checkout. If you do a time only search you will be returned a link to a single file containing the entire data set regardless of your time search. The correct effect should have been to break that single file up into files of ~1 week of data but since TSTART was equal to 0 in all the files ingested, the break up never occurred.
2. The entry in the GTI extension was the entire data interval instead of just the data interval covered by the file. This has effects on calculating the exposure that have been discussed in with respect to other tools as well.

The interesting thing is that both of these are new errors. The version of obsSim used in checkout 1 had the correct functionality.

### Minor Issues

1. ONTIME & TELAPSE keywords are in all three headers. According to the FITS file definition they should only be in the GTI extension
2. TSTART & TSTOP keywords are in the primary header. These should not be here according to the file definitions.

## Documentation

The observationSim documentation was generally good and easy to use. The documentation of the sources in the celestialSources packages was not always as useful. There were a couple of specific things I noticed, however:

1. Not all the source models are documented. A few are documented in the User's Guide found in the observationSim doxygen pages. A few were documented in Seth's confluence page but some of the basic ones like point sources with a PowerLaw or BrokenPowerLaw spectrum didn't seem to be documented anywhere. I found them by looking at the sample source XML files in the package. Admittedly, I only looked in the

observationSim, celestialSources and flux packages. There should be a single page documenting each type of source available, its parameters and an example of the XML entry to create that source.

2. PulsarSpectrum documentation was a little spread out and incomplete. There is some information about the source in the Doxygen but then the tutorial is on a completely different server (The link to which did work but just came back to the page the link was on).
3. For the PulsarSpectrum model, the last 4 parameters in the XML model definition are never described or explained. The Doxygen says to refer to the tutorial for how to create a PulsarSpectrum source and the tutorial says to look at the Doxygen for the description of those parameters.

## Interface

The program was relatively easy to use but took a little work to get the necessary input files set up. A few comments on the interface:

1. The gui keyword is not yet in the parameter file and the functionality to use the st\_app gui interface is not set up. I was trying to play around with the number of events per file and other hidden parameters and the gui makes seeing and editing those parameters very easy.
2. Creating a PulsarSource object is different than all the other sources in that you not only have to put an entry in the XML file, but you also have to edit a 'PulsarDataList.txt' file which is in the source tree, not the local directory as well. I think this should be changed. You should only have to specify parameters in a single location, preferably the XML file like all the other sources. Additionally, in the examples and documentation, the source position is defined twice, once in the PulsarDataList.txt file and once in the XML file. What happens if they are different in the two locations, does one trump the other and if so which one.

## Functionality

One bit of functionality that needs to be implemented at some point is SAA passages. Right now, as far as I can tell there is no way to tell the program to simulate the SAA.