

Fermi-LAT Background Estimator

Vlasios Vasileiou (vlasisva@gmail.com)

CNRS/IN2P3 Laboratoire Univers et Particules de Montpellier & Université Montpellier II

BKGE version v4 - September 2013

1 Introduction

The *Fermi-LAT Background Estimator* (BKGE) is a publicly available open-source tool that can estimate the expected background of the Fermi-LAT for any observational configuration and duration.¹ It produces results in the form of text files, ROOT files, *gtlike* source-model files (for LAT maximum likelihood analyses), and PHA I/II FITS files (for RMFit/XSpec spectral fitting analyses). It currently works with Pass7 Transient-class events. However, as new LAT data classes become available (e.g., Pass8), its functionality will be extended.

Its core is written in C++ and its user interface in Python. The tool's internal workings are fully documented in Ref. [1]. Please cite this paper to acknowledge the tool's use. Please see the associated publication for caveats, scope, and the expected accuracy of its predictions.

2 Release Notes

This software is provided as is. If you have any questions on scope, degree of applicability, or if you have any spurious results please do not hesitate to email me at vlasisva@gmail.com. If you have any problems compiling it, I can provide limited support based on my experience compiling it on Ubuntu Linux.

The C++ code of the BKGE was written by Vlasios Vasileiou while at NASA, Goddard Space Flight Center and CNRS/IN2P3 Laboratoire Univers et Particules de Montpellier. The python interface was written by Vlasios Vasileiou, with debugging help and support from Giacomo Vianello and Nicola Omodei (of SLAC and Stanford University). This tool was produced within the Gamma Ray Bursts group of the Fermi-LAT Collaboration.

You are free to use, modify, and distribute the BKGE (here BKGE refers to the contents of the *BKGE* folder). In case of modification, you are not allowed to remove its branding (i.e., name, version, and author). In case of distribution, you have to also include this document with the distributed package.

If you like this tool and are performing Fermi-LAT GRB analyses, you are encouraged to also check out *gtBurst*², a GUI for GRB analysis developed for by G. Vianello.

3 Compilation

These instructions apply to Linux (with gcc) and have been tested on an Ubuntu 13.04 64bit box. The tool should also compile on Macintosh with small modifications (by you). It is not expected to work on OSX Tiger, OSX Leopard, and any Mac OSX PPC distribution because the *Fermi Science Tools* (ST), a required package by the BKGE, does not run on these systems. Finally, the BKGE does not work on Windows.

3.1 Software requirements

The BKGE depends on the following additional pieces of software:

¹ Because of its 15% systematic error, the BKGE is mostly applicable to short duration <1ks observations.

² <http://sourceforge.net/projects/gtburst/>

- Standard compilation tools, such as gcc, make, ld, etc.
- *Fermi Science Tools*³. The BKGE runs some of these tools (e.g., gtselect, gtexpcube2, gtltcube) as part of estimating the background.
- ROOT⁴. The BKGE compilation makefile needs to have access to the ROOT header files and compiled binaries, and the Python interface to PyROOT. Depending on your ST configuration you may or may not have ROOT installed. In the latter case, you need to setup your environment before compiling or running the BKGE. For csh:

```
setenv ROOTSYS <the path where ROOT is installed>
setenv PATH ${PATH}:${ROOTSYS}/bin
setenv PYTHONPATH ${ROOTSYS}/lib
setenv LD_LIBRARY_PATH ${LD_LIBRARY_PATH}:${ROOTSYS}/lib
```

- Python, for the interface. Take note that the ST already come with an installation of Python along with several needed packages (e.g., pyfits).
- Optionally, if you need output in the form of PHA I/II FITS files:
 - A working installation of the *HEASARC FTOOLS*⁵, and especially the tool *ascii2pha*⁶.
 - pyfits⁷. Hint: if your system does not already have pyfits installed, you can switch to the Python installation of the ST which includes it.

3.2 Compilation Steps

- First of all, make sure that the tool *root-config* of ROOT can be executed, since it is needed by the makefile.
- Then make sure that your Science Tools are properly setup. This entails in setting your FERML_DIR environmental variable (with the full path of your ST distribution) and executing the fermi-init.csh (.sh) script of the ST.
- Unzip the BKGE tarball and enter the BKGE directory. Then check the makefile and make sure that the line starting with LINK_TARGET points to the correct versions of the CLHEP and cfitsio libraries. The versions included by the ST at the time of the BKGE's release were 3.29 and 2.1.0.1 respectively, so we link against *libcfitsio-3.29.so* and *libCLHEP-2.1.0.1.so*. Check your \${FERML_DIR}/lib directory and see if you have these two files. If not, then you have some newer version and you have to update the LINK_TARGET line accordingly.
- After you are done, just

```
make
```

- If all goes well you will have a a ROOT library named *libBKGE.so* in the main folder of the BKGE.

4 How to use the BKGE

You can use the BKGE through the provided Python script *BKGE_interface.py*. This script prepares the data, performs some cross-checks, and calls (through pyROOT) the C++ functions of the BKGE. In principle, with some programming from your part to recreate the python-code functionality, you can use the C++ code directly either by linking the BKGE library in your code or by talking to the BKGE directly from inside ROOT. This option is not suggested. However, if you decide to pursue it, please consult the source codes of the Python interface and of the BKGE. Before starting you need to add the *external* folder to your LD_LIBRARY_PATH as for csh:

³ <http://fermi.gsfc.nasa.gov/ssc/data/analysis/software/>

⁴ <http://root.cern.ch>

⁵ <http://heasarc.gsfc.nasa.gov/ftools/>

⁶ <http://heasarc.gsfc.nasa.gov/lheasoft/ftools/fhelp/ascii2pha.txt>

⁷ http://www.stsci.edu/institute/software_hardware/pyfits

```
setenv LD_LIBRARY_PATH ${LD_LIBRARY_PATH}:<full path of external
folder>
```

The Python interface provides functions that can produce background estimates in various formats, as shown below. The common arguments among these functions are explained in Table 1. The remaining, special to each function, arguments are explained below.

- **CalculateBackground** (start, stop, grb_trigger_time, RA, DEC, FT1, FT2, OUTPUT_DIR="output", emin=-1, emax=-1, ebins=-1, chatter=1, overwrite=False, EvaluateMaps=True, CalcResiduals=True, ROI_Calculate=1, ROI_Containment=0.95, ROI_Localization_Error=0, ROI_Radius=12, ROI_Max_Radius=12, GRB_NAME="", ROI_RadiusFile="")
 - Estimates the background for a simple observation. The output is in the form of a text and a ROOT file, and of some pretty graphics (see e.g., Fig. 1)
 - The arguments *EvaluateMaps* and *CalcResiduals* are to be used internally by the code. Please ignore them.
- **Make_BKG_PHA**(start, stop, grb_trigger_time, RA, DEC, FT1, FT2, emin, emax, ebins, ROI_Calculate, OUTPUT_DIR="output", chatter=1, overwrite=False, ROI_Containment=0.95, ROI_Localization_Error=0, ROI_Max_Radius=12, ROI_Radius=12, GRB_NAME="", ROI_RadiusFile="")
 - Estimates the background for a single observation and produces a PHA FITS file that can be used for spectral fitting with RMFit or XSpec.
- **Make_BKG_PHA2**(grb_trigger_time, RA, DEC, FT1, FT2, emin, emax, ebins, ROI_Calculate, ROI_Localization_Error=0, ROI_Max_Radius=12, ROI_Radius=12, ROI_Containment=0.95, ROI_RadiusFile="", OUTPUT_DIR="output", GRB_NAME="", chatter=1, overwrite=False, Time_bins_def_file="", start=-1, stop=-1, dt=-1)
 - Estimates the background for a series of consecutive or not observations to produce a PHA II FITS file that can be used for spectral fitting with RMFit or XSpec.
 - There are two ways to specify the time intervals. Either by providing a triplet of (*start*, *stop*, *dt*) to process a series of consecutive intervals with duration *dt* each, or by providing a file containing the time-bin definitions through the *Time_bins_def_file* argument. This file can be either a text file containing one space-separated start/stop pair of time offsets (i.e., times relatives to the grb_trigger_time) per line or a FITS bindef file produced using the ST *gtbindef*.⁸
- **MakeGtLikeTemplate**(start, stop, grb_trigger_time, RA, DEC, FT1, FT2, OUTPUT_DIR="output", chatter=1, ROI_Radius=15, GRB_NAME="")
 - Estimates the background for a simple observation and saves it in the form of a text file appropriate to be used by the maximum-likelihood analysis tool *gtlike*.

Energy binning

The BKGE predicts the background internally in 20 consecutive log-energy bins ranging from 50MeV to 150GeV. Using the *emin*, *emax*, *ebins* parameters, you can ask for an alternative binning configuration. Take note, however, that if your desired binning is considerably different than the default configuration (e.g., the edges of your desired bins are not aligned with the default ones), there may be a small accuracy penalty. You cannot ask for estimates in energies external to the 50MeV-150GeV range, and you cannot have non logarithmic-in-energy binning.

⁸ <http://fermi.gsfc.nasa.gov/ssc/data/analysis/scitools/help/gtbindef.txt>

Region of Interest

The BKGE can produce estimates for several kinds of Region of Interest (ROI) configurations, controlled by the *ROI_Calculate* argument, as:

- *ROI_Calculate*=0 -> Use a constant ROI radius equal to *ROI_Radius*.
- *ROI_Calculate*=1 -> Calculate an energy-dependent ROI equal to the $\langle \text{ROI_Containment} \rangle$ containment radius of the LAT point spread function (PSF) and the *ROI_Localization_Error* added in quadrature. Since the LAT PSF depends on the energy and the off-axis angle of the events, it is evaluated at an energy equal to the geometric mean of each energy bin and an off-axis angle equal to that of the center of the ROI (i.e., at *RA/DEC*) at the middle of the analyzed time interval (i.e., at time $(\text{start}+\text{stop})/2$)¹⁰. The resulting ROI radius is capped at a value of *ROI_Max_Radius* to prevent it from becoming too wide (hence the background contamination becoming too large) at small energies.
- *ROI_Calculate*=2 -> Read ROI radius from file given by *ROI_RadiusFile* parameter. The file should contain *ebins* entries, one per line, with each entry being a ROI radius in degrees.

Earth limb and Zenith theta cut

The Earth-limb component of the background is not fully accounted for by the BKGE. A standard procedure in LAT data analysis is to minimize the contribution from this component using a cut on the maximum acceptable Zenith theta angle of the events. The BKGE produces estimates following such a cut, and specifically a 100deg maximum value of Zenith theta. There is no way to change this value.

FT1 and FT2 files

The BKGE needs an FT1 and an FT2 file spanning the observation under consideration. The time interval spanned by the FT1 file can be as small as that of the observation, while the FT2 file needs to have some padding around the observation of at least 5 minutes. The latter is required because the FT2 files available to the public contain the pointing information in 30s intervals. Because the BKGE performs calculations on finer time scales (down to 5s), it interpolates these 30s-interval data to produce a time series with 1s precision. The 5 minutes padding time is needed to avoid the edge effects of the interpolation appearing inside our time interval.¹¹

The BKGE does not use the events inside the FT1 file for estimation purposes, but only for producing some pretty graphics comparing actual and predicted numbers of events. However, the important piece of information in the FT1 files is the Good Time Intervals (GTIs), which inform the BKGE when the LAT was off or whether you have excluded some parts of the observation from your observation. The latter can happen during a maximum likelihood analysis with *gtlike*, in which the tool *gtmktime* is first applied on the FT1 files to reject all time intervals for which the ROI radius intersects your Zenith theta cut. In such a case, parts of your original time interval under consideration might be excluded, as reflected in the GTIs saved in the produced by *gtmktime* FT1 file. By providing this FT1 file to the BKGE, you can be sure that the BKGE will produce estimates for an observation described by the exact GTIs of your analysis.

4.1 Output in ROOT and text file forms

The simplest application is to ask for the estimated background of a single observation. For example let's try to estimate the background during the prompt emission phase (say 0-10s) of the very bright GRB090510.

We can download these files from the *Fermi Science Support Center* (FSSC), and specifically from the LAT data server¹². We enter the name of the object ("GRB090510"), a 15deg ROI radius, a 50MeV-150GeV energy range (the energy range we want to produce background estimates for), and a time

¹⁰ Exception: if your time interval starts before the trigger time, the PSF is evaluated at an off-axis angle at trigger time instead.

¹¹ The LAT collaboration has also internally access to 1s sampling-rate FT2 files, which can be used by the BKGE without the need of interpolation and extra padding. Such files may or may not become publicly available in the future.

¹² <http://fermi.gsfc.nasa.gov/cgi-bin/ssc/LAT/LATDataQuery.cgi>

interval extending by 1ks after the trigger time, which is at FERMI Mission Elapsed Time (MET) of 263607781.9710s. We select “Extended” for the data type since the “Photon” type does not include Transient-class events. We check the Spacecraft data box to also ask for an FT2 file. The data server prepares the file and also gives us the direction of the GRB, which we’ll use: RA/DEC=(333.553, -26.5975)deg (J2000).

We start by running Python in interactive mode and executing the BKGE interface as:

```
python -i BKGE_interface.py
```

Let’s first ask for a background estimate at an interval far after the end of the GRB, say at 500-700s post-trigger:

```
CalculateBackground(start=500, stop=700, RA=333.553, DEC=-
26.5975, grb_trigger_time=263607781.9710, FT1="/tmp/FT1.fits", FT2="/tmp/FT2.fits")
```

After less than a minute, the code finishes and produces the following output:

```
*-----*
| Background Estimator (public) |
| v4 September 2013            |
|                               |
| contact: Vlasios Vasileiou   |
| vlassisva@gmail.com         |
| http://arxiv.org/abs/1307.4284 |
*-----*
BackgroundEstimator: Using P7TRANSIENT_V6 data class.
Data Files Energy: (50.0-150000.0)MeV - 20 bins
User is using the default (shown above) energy configuration
Make_Background_Map: Calculating exposure
[*****] *****
Out[1]: (37.0, 40.23, -0.57, -0.4)
```

The function returns a list of values containing the number of detected events, the number of estimated by the BKGE events, the significance, and the significance if a 15% systematic error on the background estimate is taken into account. In addition, it produces the following files in the folder *output/GRB_MET_263607781.97/500.00_700.00*¹³:

- A text file containing a range of information for each of the energy bins: the edges, the actual and estimated number of events (both per bin and cumulative), the exposure, and the ROI radius used.
- A png image (see top panel of Fig.1) comparing the actual and estimated numbers of events, showing the ROI radius used, and giving some configurational details. This canvas is also saved in file with name “...Results....root”.
- A ROOT file, *Plots.root*, containing histograms with the LAT pointing information (i.e., data from the FT2 file).
- Various other ROOT and FITS files containing information produced in each step of the background-estimation process. You can check them out, but they should not be of direct interest to the typical user.

Repeating the above exercise, and to appreciate the brightness of GRB090510, we ask for a background estimate from 0-10s post-trigger. The BKGE predicts 2.4 events, while 266 are actually detected (that’s bright!). The result is in the right panel of Fig. 1.

¹³ The output folder is decided as *<OUTPUT_DIR>/<GRB_NAME>/<START>_<STOP>*

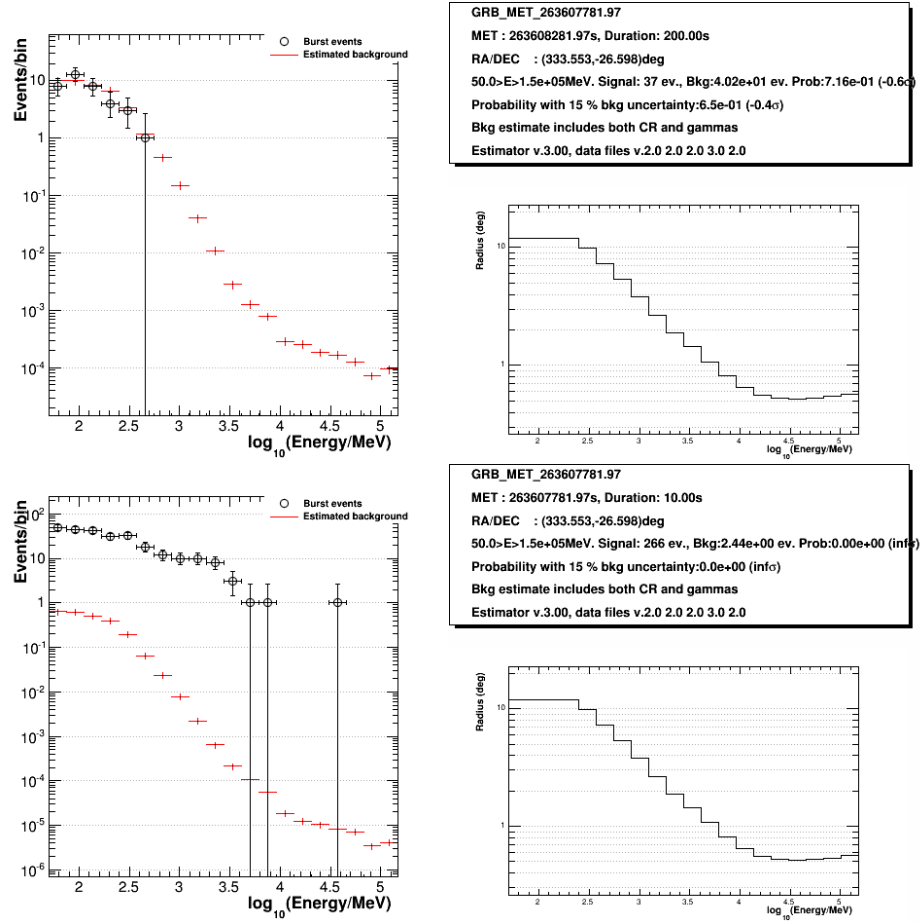


Fig. 1: Output samples from the BKGE and GRB090510. Top: 500-700s post-trigger (emission has subsided), bottom: 0-10s post-trigger (strong emission).

4.2 Maximum-likelihood analysis template

The *gtlike* tool, part of the ST, can be used to perform maximum likelihood analyses on LAT data. Part of these analyses is the description of a source model for the expected background in the data. The user has multiple ways for describing the source model¹⁴, with one of them called *FileFunction*. This option requires an input ASCII file with columns of energy and differential flux values, which the BKGE provides using the *MakeGtLikeTemplate* command, as:

```
MakeGtLikeTemplate(start=0, stop=10, RA=333.553, DEC=-26.5975,
grb_trigger_time=263607781.9710, FT1="/tmp/FT1.fits",
FT2="/tmp/FT2.fits", ROI_Radius=15)
```

This command produces three such ASCII text files with names ending in: "CR_EGAL.txt", "GALGAMMAS.txt", and "TOTAL.txt", describing the predicted background flux per energy bin of the Cosmic Ray and extra-galactic components of the background (the "isotropic background"), the point sources and diffuse galactic component of the background, and their sum, respectively.

You have freedom to model the LAT background in various ways with *gtlike*. You can describe it using a single total-background component or instead use its two constituent components (GALGAMMAS and CR_EGAL) separately. In the latter case, you can choose to model the CR_EGAL using the the BKGE, the Galactic diffuse component using the templates provided by the LAT collaboration¹⁵, and add any bright sources manually. This is the approach adopted by the Fermi-LAT collaboration for creating the first LAT GRB catalog[2], selected based on the fact that the diffuse-emission background templates

¹⁴ See http://fermi.gsfc.nasa.gov/ssc/data/analysis/scitools/source_models.html

¹⁵ <http://fermi.gsfc.nasa.gov/ssc/data/access/lat/BackgroundModels.html>

provide both spatial and spectral information of the background, while the BKGE Python interface produces only a single text file (not a skymap) containing only the spectral info¹⁶. For analyses near the Galactic plane, where the flux gradient of the diffuse component is large, the official LAT template provides more accurate modeling of the Galactic diffuse background than the BKGE template.

The ROI radius used by this function is always constant and equal to “ROI_Radius”, since *gtlike* analysis is performed on fixed-ROI-radius data sets.

Earth's Limb and Zenith theta cut

gtlike and the BKGE try to minimize the Earth-limb component of the background differently. The BKGE treats a Zenith theta cut as a cut to be applied *directly on the events*. Thus, if your ROI intersects the Earth's limb, the background estimates will correspond to *only the unocculted part of the ROI*. On the other hand, *gtlike* applies a Zenith theta cut *on the data as a whole* and specifically on the *time intervals* (using *gtmktime*), rejecting *all* events produced in intervals during which any part of the ROI intersected your Zenith theta cut. This different application of the Zenith theta cut can invalidate certain analyses, if care is not exercised. Specifically, you need to make sure to provide the BKGE with the FT1 file produced by *gtmktime*, instead of the one given to *gtmktime*. This way, the BKGE will know which parts of the observation were excluded by *gtmktime* and will produce results proper for your specific *gtlike* analysis. Finally, because you cannot change the Zenith theta value of the BKGE (it is fixed at 100deg), you must use the same value in your *gtlike* analysis.

4.2.1 PHA and PHAII files for RMFit and XSpec

We can produce a PHA I FITS file containing the background of a single observation as:

```
Make_BKG_PHA(start=0, stop=10, RA=333.553, DEC=-26.5975,
grb.trigger_time=263607781.9710, FT1="/tmp/FT1.fits",
FT2="/tmp/FT2.fits", emin=50, emax=150000, ebins=20)
```

and for two consecutive observations each of 5s duration as:

```
Make_BKG_PHA2(start=0, stop=10, dt=5, RA=333.553, DEC=-26.5975,
grb.trigger_time=263607781.9710, FT1="/tmp/FT1.fits",
FT2="/tmp/FT2.fits", emin=50, emax=150000, ebins=20)
```

The resulting files will be in the same output folder as above, and will have *.pha* extensions. Some comments:

- Some parameters in these two commands, specifically *emin*, *emax*, *ebins*, and *ROLCalculate*, do not have a default value. The user has to manually specify them just to be sure that the values meant to be chosen by the user are actually chosen.
- The produced FITS files have an additional table named *ROLRADIUS* showing the used radius in the background estimate. It is suggested that the user checks that the used ROI radius (as reported in that table) is what was requested, especially if *ROLCalculate*==1. It is also a good idea to make sure that the ROI_Radius used in the BKGE is exactly the one used in the code used for producing the actual events files (to be used together with the BKGE results for spectral fitting).
- There is a SYST_ERR (systematic error) of 15% and a STAT_ERR (statistical error) of 1% reported in the produced files.

References

- [1] V. Vasileiou, “A tool to estimate the Fermi Large Area Telescope background for short-duration observations”, *Astroparticle Physics*, Volume 48, August 2013, Pages 61–74 <http://www.sciencedirect.com/science/article/pii/S0927650513001151>
<http://arxiv.org/abs/1307.4284>

¹⁶ The BKGE does estimate the background from each direction in the sky, saved in the secondary ROOT files in the output directory. Thus, in principle, the functionality of creating predicted background template skymaps for the Galactic diffuse plus point source components is available. However, the official templates are accurate enough so this functionality was not developed.

-
- [2] Fermi-LAT collaboration, “*The First Fermi LAT Gamma-Ray Burst Catalog*”, accepted by Astrophysical Journal supplements, <http://arxiv.org/abs/1303.2908>

Argument name	Units	Default	Comments
start, stop	sec	–	Edges of the observation under consideration, relative to grb_trigger_time
grb_trigger_time	sec	–	GRB trigger time in FERMI Mission Elapsed Time (MET) ⁹
RA, DEC	deg	–	Center of your Region of Interest (ROI)
FT1	–	–	Full or relative path of an FT1 (LAT events file) spanning the observation under consideration. The code needs these files to know the Good Time Intervals (GTIs). It does not use the events contained in them.
FT2	–	–	Full or relative path of an FT2 (LAT spacecraft file) spanning the observation under consideration.
OUTPUT_DIR	–	“output”	Full or relative path of directory with BKGE’s results
(emin,emax,ebins)	(MeV, MeV, #)	-1,-1,-1	Limits of the energy range of the produced background estimates and numbers of energy bins. For best accuracy do not use these parameter to use the default configuration present in the BKGE data files. For the P7 Transient class, the default is (50MeV - 150GeV, 20 bins).
chatter	–	1	Verbosity parameter. Takes values 1–5
overwrite	Boolean	False	overwrite results?
ROLCalculate	0,1,2	0	Should the code calculate an energy-dependent ROI radius? See text for explanation.
ROLRadius	deg	12	User-specified ROI radius (for ROLCalculate==0)
ROLContainment	–	0.95	Containment fraction of the P7TRANSIENT_V6 PSF used for calculating the ROI Radius (for ROLCalculate==1)
ROLMax_Radius	deg	12	Maximum value of the ROI radius (for ROLCalculate==1). Use 0 to have an uncapped max radius. A value of 12 deg is typically used in Fermi-LAT publications.
ROLLocalization_Error	deg	0	Error of the GRB localization (for ROLCalculate==1). This is added in quadrature to the containment fraction of the LAT PSF to produce the final ROI radius.
ROLRadiusFile	–	“”	This is the full path of a file containing ROI radius data (for ROLCalculate==2)
GRB_NAME	–	“”	Name of the GRB. It is only used for naming output folders. Leave empty for the code to decide a name based on the trigger MET.

Tab. 1: Arguments common among the functions comprising the python interface.