Astronomy & Astrophysics manuscript no. gammapy-paper ©ESO 2020
August 13, 2020

Gammapy: A Python package for gamma-ray astronomy

Christoph Deil[1], Axel Donath[1], Régis Terrier[2], Johannes King[1], Léa Jouvin[2], Manuel Paz Arribas[5], Ellis Owen[6], Julien Lefaucheur[3], Olga Vorokh[8], Jonathan Harris[13], Brigitta Sipocz[11, 12], Dirk Lennarz[10], Helen Poon[1], Nachiketa Chakraborty[1], Domenico Tiziani[9], Luigi Tibaldo[1], Victor Zabalza[14], Rolf Bühler[4], Stefan Klepser[4], and Ignasi Reichardt[7]

[1] Max-Planck-Institut für Kernphysik, Heidelberg, Germany
[2] APC, University of Paris 7, France
[3] LUTH, Observatoire de Paris, Meudon, France
[4] DESY, Zeuthen, Germany
[5] Humboldt University, Berlin, Germany
[6] UCL-MSSL, Dorking, United Kingdom
[7] INFN, Padova, Italy
[8] Belarusian State University, Belarus, Minsk
[9] FAU, Erlangen, Germany
[10] School of Physics and Center for Relativistic Astrophysics, Georgia Institute of Technology, Atlanta, GA, USA
[11] Centre for Astrophysics Research, University of Hertfordshire, College Lane, Hatfield, AL10 9AB, UK
[12] Institute of Astronomy, University of Cambridge, Madingley Road, Cambridge, CB3 0HA, UK
[13] Department of Physics, Durham University, South Road, Durham, DH1 3LE, UK
[14] TODO: Victor address

August 13, 2020

ABSTRACT

In this paper, we plan to describe Gammapy version 0.9, and briefly outline plans for 1.0. We plan to submit it in December 2018.
Gammapy context: HESS, CTA, Fermi, open data, open-source, Python, community
Gammapy aims
Gammapy methods
Gammapy results
Gammapy conclusions

Key words. Gamma rays: general - Astronomical instrumentation, methods and techniques - Methods: data analysis

1. Introduction

Gamma-ray astronomy is a rather young field of research, driven by experiments with proprietary software often based on ROOT, because of the particle physics background. Such as HESS, Veritas or Magic.

The Cherenkov Telescope Array will be operated as an open observatory for the first time. Thus there is a need for open analysis software as well.

In recent years Python has established as one of the standard programming languages for data science as well as astronomy. After a phase of many packages the ecosystem was unified with `Astropy` (Astropy Collaboration et al. 2013).

Gammapy is a Python package for gamma-ray astronomy.

TODO: Figure 1: Data -> Gammapy -> Spectra etc with some details

Basic idea: build on Numpy and Astropy, use Python stack

TODO: Figure 2: Gammapy software stack

Here's a list of references I'd like to cite ... to be incorporated into the main text somewhere:

- The Python programming language[1]
- Gammapy webpage[2]
- PyFACT (Raue & Deil 2012)
- FITS (Pence et al. 2010)
- Gamma-astro data formats tbd (all contributors to the spec)
- Sherpa (Refsdal et al. 2011, 2009)
- Naima[3] (Zabalza 2015)
- Gammapy use in science publications: (Owen et al. 2015), SNR shell, HGPS

* Gammapy – A Python package for gamma-ray astronomy * Gammapy – A prototype for the CTA science tools * Astropy: A community Python package for astronomy * THE ASTROPY PROJECT: BUILDING AN INCLUSIVE, OPEN-SCIENCE PROJECT AND STATUS OF THE V2.0 CORE PACKAGE * GammaLib and ctools * Fermipy proceedings * SunPy: Python for Solar Physics. An implementation for local correlation tracking *

2. Gammapy package

The `Gammapy` package is structured into multiple sub-packages which roughly follow the stages in the data reduction workflow.

[1] `http://fits.gsfc.nasa.gov/`

[2] `http://gammapy.org`

[3] `https://github.com/zblz/naima`

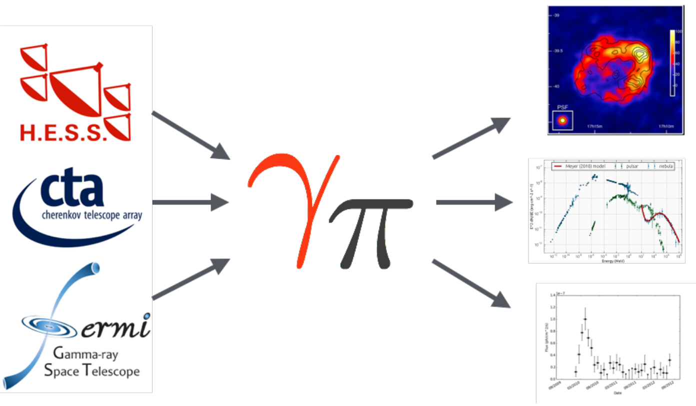

Fig. 1. Gammapy is a Python package for high-level gamma-ray data analysis. Using event lists, exposures and point spread functions as input you can use it to generate science results such as images, spectra, light curves or source catalogs. So far it has been used to simulate and analyse H.E.S.S., CTA and *Fermi*-LAT data, hopefully it will also be applied to e.g. VERITAS, MAGIC or HAWC data in the future.

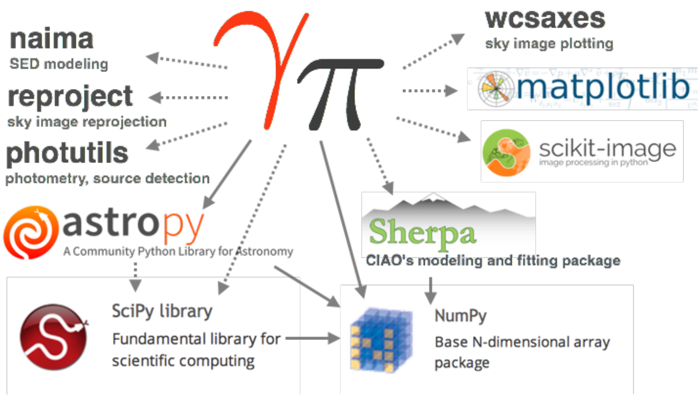
naima
SED modeling
reproject
sky image reprojection
photutils
photometry, source detection
γπ
wcsaxes
sky image plotting
matplotlib
scikit-image
image processing in python
Sherpa
CIAO's modeling and fitting package
astropy
A Community Python Library for Astronomy
SciPy library
Fundamental library for scientific computing
NumPy
Base N-dimensional array package

Fig. 2. The Gammapy stack. Required dependencies Numpy and Astropy are illustrated with solid arrows, optional dependencies (the rest) with dashed arrows.

```python
from gammapy.data import DataStore

data_store = DataStore.from_dir("$GAMMAPY_DATA")
obs_ids = [1, 2, 3]
observations = data_store.get_observations(obs_ids)
```

Listing 1: Using gammapy.data to access DL3 level data with a DataStore

2.1. *gammapy.data*

The `gammapy.data` sub-package provides access to DL3 level data and observation handling.

2.2. *gammapy.makers*

Data reduction

```python
from gammapy.makers import MapDatasetMaker

maker = MapDatasetMaker()
dataset = maker.run(dataset, observation)
```

Listing 2: Using gammapy.data to access DL3 level data

```python
from gammapy.modeling.models import (
        SkyModel,
        PowerLawSpectralModel,
        PointSpatialModel
)

pwl = PowerLawSpectralModel()
point = PointSpatialModel()

model = SkyModel(
        spectral_model=pwl,
        spatial_model=point
        name="my-model"
)
print(model)
```

Listing 3: Using gammapy.data to access DL3 level data

2.3. *gammapy.datasets*

DL4 level data

2.4. *gammapy.modelling*

Models and fitting

2.5. *gammapy.estimators*

Estimators

2.6. *gammapy.visualisation*

Plotters etc.

2.7. *gammapy.analysis*

High level analysis API

2.8. *gammapy.astro*

Dark matter models, source population modelling

2.9. *gammapy.catalog*

Gamma-ray catalog access

2.10. gammapy.maps

Map data structures WCS and HPX

2.11. gammapy.irf

IRF classes

2.12. gammapy.utils

Utility functions...

Outline: * List typical analysis use cases * Can use from Python and Jupyter -> show Figure with Jupyter notebook here. * Gammapy code structure * How Numpy and Astropy is used

Figures: * Add a Figure showing dataflow in a typical application DL3 at the top, spectrum, map, lightcurve, fit results at the bottom. Mention major classes in between (DataStore, EventList, Map, MapMaker, MapFit, . . .) * Probably not: Figure showing sub-packages and how they relate (gammapy.data and gammapy.irf at the base, then gammapy.maps, etc. * The code example Figure how to make a counts map, to explain how the package works.

3. Applications

Each application example is a notebook in the online material: We could have one analysis as Python scripts instead of notebook in the online material. At the start of this section, point to gammapy-paper repo on Github and say that there's a Binder where people can try the examples online.

TODO: mention other application examples (joint Crab paper, HESS validation paper, HGPS, ...) here or in a subsection "other applications" at the end of this section?

3.1. Source detection

See Figure 3.

Ref: (Stewart 2009)

3.2. CTA simulation

CTA application example. 3D simulate and fit using public prod3 IRFs. Diffuse emission + maybe a shell -> image and spectrum come out.

3.3. HESS

Maybe HESS Light curve using PKS flare from HESS

3.4. Fermi

Fermi: Galactic center, as in our notebook, same region as CTA.

4. Gammapy project

Infrastructure etc.

4.1. Development, testing

-Github, pytest, CI, PIGs?

4.2. Documentation

- Notebooks

4.3. Software distribution and user support

- Pip, conda, versions, gammapy download

4.4. Community

TODO: Figure: Screenshot of Jupyter notebook or docs with notebook, could show the interactive maps view

```
m = Map.read("diffuse.fits")
m.plot_interactive()
```

5. Summary and Outlook

Summary what we have in v0.9 and presented in this paper.

Roadmap to v1.0, about half a page.

Short conclusion: Gammapy has potential to be the Python package for gamma-ray astronomy.

Acknowledgements. We would like to thank the Numpy, Scipy, IPython and Matplotlib communities for providing their packages which are invaluable to the development of Astropy. We thank the GitHub team for providing us with an excellent free development platform. We also are grateful to Read the Docs (https://readthedocs.org/), and Travis (https://www.travis-ci.org/) for providing free documentation hosting and testing respectively. Finally, we would like to thank all the Gammapy users that have provided feedback and submitted bug reports. TODO: copy over stuff from http://docs.gammapy.org/en/latest/about.html#thanks.

References

Astropy Collaboration, Robitaille, T. P., Tollerud, E. J., et al. 2013, A&A, 558, A33

Owen et al., E. 2015, PoS(SciNeGHE2014), 34 [arXiv:1506.02319]

Pence, W. D., Chiappetti, L., Page, C. G., Shaw, R. A., & Stobie, E. 2010, AAP, 524, A42+

Raue, M. & Deil, C. 2012, in American Institute of Physics Conference Series, Vol. 1505, American Institute of Physics Conference Series, ed. F. A. Aharonian, W. Hofmann, & F. M. Rieger, 789–792

Refsdal, B., Doe, S., Nguyen, D., & Siemiginowska, A. 2011, in 10th SciPy Conference, 4 – 10

Refsdal, B. L., Doe, S. M., Nguyen, D. T., et al. 2009, in 8th SciPy Conference, 51 – 57

Stewart, I. M. 2009, A&A, 495, 989

tbd (all contributors to the spec), A. ????, Data formats for gamma-ray astronomy : Version 1.0

Zabalza, V. 2015, ArXiv e-prints [arXiv:1509.03319]

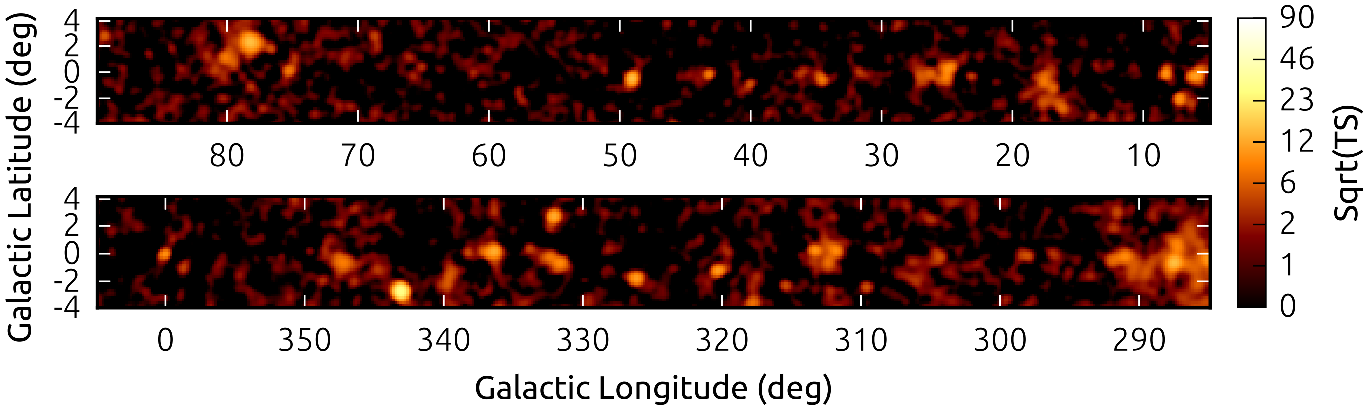

Fig. 3. Gammapy application example: A *Fermi* survey TS map of the inner Galactic plane region.

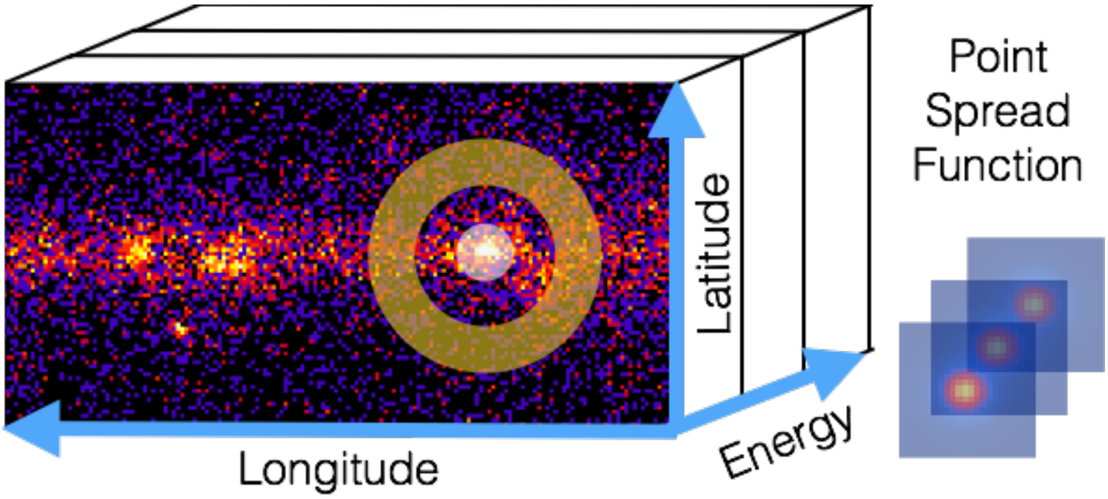

Fig. 4. Gammapy data model illustration. Binned analysis of lon-lat-energy cube data is supported via joint likelihood analysis of one image per energy bin. On-off-region based spectral analysis is supported as well.