Gammapy: present status and future roadmap

B. Khélifi,[a,1,*] **R. Terrier,**[a] **A. Donath,**[b] **A. Sinha,**[c] **Q. Remy**[d] **and F. Pintore**[e]

[a]*Université Paris Cité, CNRS, Astroparticule et Cosmologie, F-75013 Paris, France*

[b]*Center for Astrophysics | Harvard & Smithsonian, Cambridge, U.S.A.*

[c]*IPARCOS Institute and EMFTEL Department, Universidad Complutense de Madrid, E-28040 Madrid, Spain*

[d]*Max-Planck-Institut für Kernphysik, P.O. Box 103980, D 69029 Heidelberg, Germany*

[e]*INAF IASF Milano, 20133 Milano, Italy*

E-mail: khelifi@in2p3.fr

Since its start in 2014, the lightweight open source Python library Gammapy has come a long way to become a popular data analysis package for high-energy astrophysics. Selected as the official CTAO Science Analysis tool, it is also an approved analysis software within the H.E.S.S. and MAGIC collaborations. The first long-term version, Gammapy v1.0 was released on late 2022. It is compliant with several well-established data conventions in high-energy astrophysics, and provides serialised data products that are interoperable with other software. Event lists and instrument response functions curated within the same format from various instruments can be reduced to data binned in energy, time or spatial coordinates. Thereafter, the flux and morphology of one or more gamma-ray sources can be estimated using Poisson maximum likelihood fitting and assuming a variety of spectral, temporal and spatial models. Flux points, likelihood profiles and light curves extractions are supported. Complex user defined likelihoods and models can also be implemented. In this contribution, we will highlight the main features of Gammapy v1.0, including data reduction and analysis examples from different space and ground-based instruments, applications of various background rejection techniques, and a simultaneous fitting across multiple instruments with astrophysical models. We will also present our plans for the future, showcasing new features such as the support of different event types, unbinned likelihood analysis, spectral unfolding and transient source detections. In addition to an improved API with distributed computing for scalable analysis, enhanced support for all-sky instruments like Fermi-LAT and HAWC is foreseen.

---

[1] For the Gammapy team

[*]Speaker

1. Gammapy, a Python package for γ-ray astronomy

Gammapy[1] is an open Python library [1] aiming to derive astrophysical products and catalogs from very-high energy (VHE) gamma-ray high-level data, such as those from H.E.S.S., MAGIC, VERITAS, HAWC and also Fermi-LAT. These high-level data are produced by *Imaging Atmospheric Cherenkov Telescopes* (IACTs) or *Water Cherenkov Detectors* (WCD) as outputs of their low-level analysis pipelines, consisting of the steps of calibration, reconstruction and background reduction. These high-level data contain event list, Instrument Response Functions (IRFs) and their associated metadata, or also data cubes like an exposure map. Gammapy reads data curated with the `gamma-astro-data-format` (GADF) format [2] and serialised into FITS [3].

This package is built within the Python ecosystem and has as main dependencies `numpy` [4] for the n-dimensional data structures, `astropy` [5] (to which Gammapy is affiliated) for the astronomical functionalities, `scipy` [6] for numerical algorithms, `iminuit` [7] for numerical minimisation and `matplotlib` [8] for visualisation. Figure 1 summarises the core ideas of Gammapy.
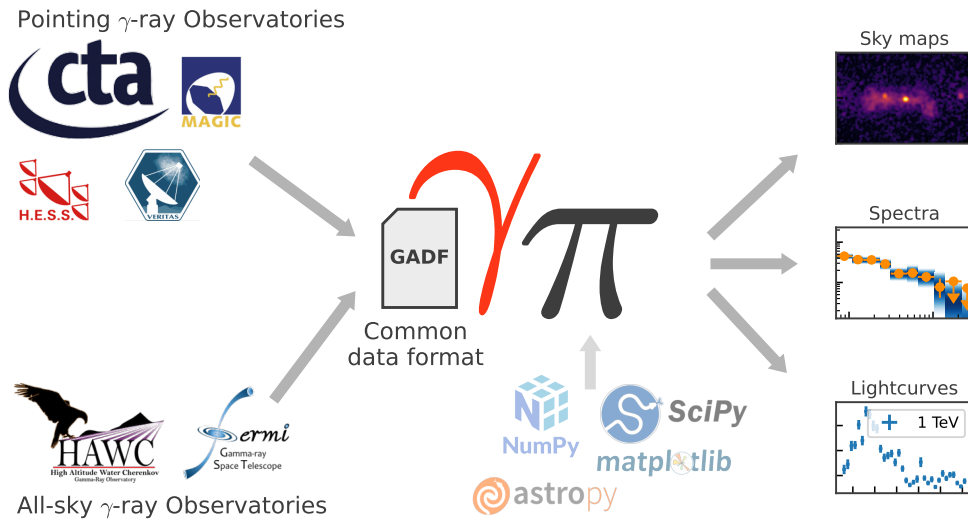

Figure 1: Core idea and relation of Gammapy to different γ-ray instruments and the *gamma astro data format* (GADF).

Gammapy is a mature open package built by the VHE community. It has a well defined organisation with Lead Developers, sub-package maintainers and a Coordination Committee, contributors[2] from different VHE experiments and other domains that are improving the package, the core code, its documentation or its test environment. In parallel, its scientific use is in constant progression[3]. In 2021, the CTA Observatory[4] (CTAO) has selected Gammapy as the core library of its Science Analysis Tool (SAT) [9]. It was granted the jury prize in 2022 of the open science awards ceremony of the French Ministry of Higher Education, Research and Innovation [10].

---

[1] Gammapy web site: https://gammapy.org/

[2] The contributors list can be found here.

[3] List of references using Gammapy: link

[4] CTAO web site: https://www.cta-observatory.org/

2. The Long-Term Stable version and the current v1.1 version

After the establishment of a roadmap in 2018[5], the Gammapy team has released in November 2022 its **first Long-Term Stable (LTS) version, the v1.0** (and its minor bug release v1.0.1 in March 2023). An accompaning paper has been very recently accepted by A&A and is about to be published[6]. This LTS version will be supported with as many minor bug releases as necessary (following the convention `v1.0.x`), until the release of the next LTS version (see the section 3). This release is light-weight with about 50,000 lines of code, of which 34% being the Python code, 22% being the code test functions, 26% on the code documentation and 9% on the package documentation.
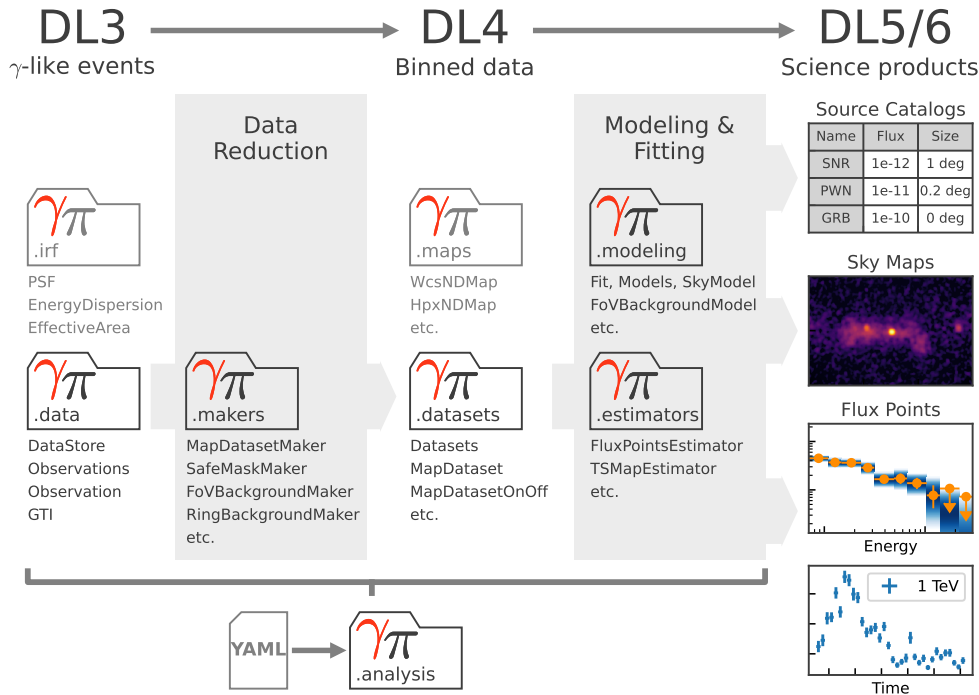

Figure 2: Data flow of the Gammapy package: for each data level following the GADF definitions, the container classes are mentionned; the flows between the levels are indicated in grey with their associated classes.

The design of the data flow has been settled, as illustrated in Fig. 2. The associated work flow consists of two steps:

- the Data Reduction: from user's choices of data selection (or filtering), of geometry (spatial, spectral or temporal) and of analysis type, the IRFs and the event list are selected, projected and reduced into data cubes and stored into a data container class inheriting from the class *Dataset*, that is the central element of Gammapy,
- the Modeling and Fitting: from user's choices of analysis and its associated parameters, source model(s) is(are) associated to Dataset(s) and their parameters are estimated using Poisson maximum likelihood fitting on the reduced data.

---

[5] Gammapy 1.0 roadmap: PIG 5

[6] The paper draft can be found in this repository.

3

Compared to its previous release (v0.20.1), most of the changes[7] are in the package infrastructure (e.g tutorial gallery relying on `Sphinx gallery`, better compliance with astropy), which permits a simplification of the maintenance. One can highlight that the support for HAWC data analysis has been highly improved, as illustrated in [11] showing the use of Gammapy by the HAWC collaboration.

The package allows to produce different types of VHE astrophysical products using algorithms that have been developed by the TeV community, the GeV community or the X-ray community:

1D spectral analyses From any spatial region, a spectral model can be adjusted using IRFs made with energy-dependent directional cuts or without and by estimating the background level with the Reflected Background method [12]; point-source sensitivity can be computed using ON and OFF estimations; flux points with their likelihood profiles can be derived from any spatial region; 1D spectrum simulation can be performed.

2D image algorithms Gammapy can produce maps of significance (using *wstat* or *cash* statistics[8]) and make source detection; excess maps can be derived by estimating the background with the Ring Background method [12], allowing then 2D map fitting.

Time domain algorithms Tools permit to extract light curves (flux as a function of time) with different binning strategies, to realise pulsar analysis or to simulate time-variable sources.

Data cube analyses Gammapy permits to fit spatial, spectral and potentially temporal models on data cubes using background models delivered with the IRFs; flux profile estimations are offered, such as data cube simulations; multi-instrument joint 3D and 1D analysis has been improved.

Global features Gammapy is delivered with a repertoire of source models[9], that describe spatial, spectral and temporal behaviours: analytical or template models are available; for each fitted parameters, one can estimate their likelihood profiles and also the contour errors between correlated parameters; home-made models or likelihood functions can be used in any analysis; an interface to dark matter spatial and spectral models and to the `naima` library [13] is provided, and also to the open TeV source catalog `gamma-cat`[10] (that can be visualised also via the web site http://gamma-sky.net/).

Since this LTS, a new feature version of Gammapy has been released in June 2023, the **v1.1**. This version has improved the user interface for some classes and functions (the Application Programming Interface, API). So a deprecation warning system has been set up to inform the users. Among the changes[11], a support for energy-dependent temporal models using templates is introduced for the event sampling (ie simulations), and also for multiprocessing. HAWC tutorials have been added, improving our interoperability capabilities. One of the best illustrations (see Fig. 3) is shown by the HAWC collaboration in [11] with the joint fit of Crab nebula VHE data from Fermi-LAT, MAGIC, VERITAS, FACT, H.E.S.S. and HAWC to derive a spectral energy

---

[7] See the v1.0 release notes.

[8] See the Gammapy statistics page

[9] https://docs.gammapy.org/1.0.1/user-guide/model-gallery/index.html

[10] Github repository: https://github.com/gammapy/gamma-cat

[11] See the v1.1 release notes

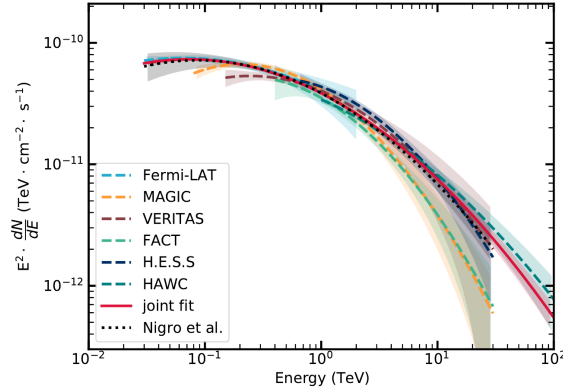distribution with a proper statistical treatment of the data.

Figure 3: Fig. 13 of [11]: Crab nebula spectral energy dispersion derived with Gammapy.

3. The roadmap for the second LTS release (v2.0)

According to the development plan of Gammapy, the LTS v2.0 is targeted for the Fall 2024. A roadmap has been written[12] and details the improvement plan on use cases, features, documentation and insfrastructure:

- *event type* handling: for IACTs and WCDs, events can be tagged according to some properties (e.g. their angular reconstruction accuracy, the number of hit sub-detectors) and be associated to specific IRFs in order to improve the analysis sensitivity (e.g. [14]),
- unbinned spectral or 3D analysis: when gamma-ray numbers are limited, for example for transient and flaring sources, unbinned likelihood analysis provides an improvement of the analysis,
- transient source detection: specific time-domain algorithms to detect variable sources and characterise their variability such as Bayesian Excess Variance,
- spectral unfolding: extraction of the intrinsic source spectrum with minimal hypothesis on its shape,
- morphology estimation: tools to measure extension profiles and their associated significances, to determine energy-dependent morphology,
- handling of systematic effects: from systematic errors stored into IRFs or by adding systematic effects on the IRFs, one could quantify these effects on the final products,
- nuisance parameters and priors for Bayesian analysis: by adding a systematic effect of unknown amplitude (e.g. a bias in the absolute energy scale), one could estimate the impact of this effect on the parameters estimation assuming a prior distribution of the nuisance parameter.

The Gammapy team aims also to make significant improvements on the configurable interface of the analysis that uses YAML files, on distributed computing for scalable analysis (using e.g.

---

[12] See the Proposal of Improvement of Gammapy: PIG 26

Ray [15]), on the handling of the future *Very-high-energy Open Data Format* (see Section 5) (clarifying our internal data format and by adding I/O functions), on the handling of meta data and provenance information to respect the FAIR principles for Reseach Softwares [16], on the DevOps tools to facilitate the maintenance and the development of the package (documentation, CI, benchmarks).

4. The preparation of the CTAO contribution

Gammapy has been selected as the core library of the open Science Analysis Tool of CTAO. The SAT will contain the science tools features of Gammapy and also the functionalities needed for some CTAO operations, such as the real time analysis pipeline. For the CTAO construction, a group of institutions that are strongly involved in Gammapy development and organization have committed to contribute all features to Gammapy to fulfill the minimal requirements of the CTAO as In-Kind Contribution. The acceptance by CTAO of a Gammapy release will follow the CTAO management plan that contains a Preliminary Design Review, a Critical Design Review and an Acceptance Review.

While maintaining Gammapy as an independent open research software aiming for interoperability between instruments, the Gammapy team is now working in close cooperation with the CTAO computing team. Regular technical meetings permit to define in detail the future SAT features in the context of the global computing needs of CTAO and to collaborate for the improvement of the Gammapy API and documentation. And at the management level, two CTAO representatives are now members of the Gammapy Coordination Committee[13].

In the meantime, CTAO is preparing the community to use the future data for scientific purposes. An open Science Data Challenge (SDC) is under consideration with objectives to test some CTAO tools (documentation, data dissemination tools, software distribution), to train the CTA consortium and the wider scientific community to use the SAT to analyse data, and to explore the CTAO expected performances. In this perspective, Gammapy as a core library of the CTAO SAT will be used to produce the simulated data of this future SDC and to analyse the SDC data.

In some regards, this SDC is a stimulating technical challenge for the Gammapy developers. Even if most of the needed features are already present for the data analysis, the developer team is polishing the simulation capabilities with a friendly API, in particular in the time domain. We are also improving the general documentation and the API description, as well as the tutorials. The developers team anticipates an increased participation to schools or to Gammapy hands-on sessions[14] and an increased number of general presentations[15].

5. Summary and conclusion

Gammapy is an open Python package for VHE γ-ray astronomy aiming to derive astrophysical VHE products from high-level data produced by VHE facilities. It uses high-level data curated

---

[13] The composition of the SC can be found here
[14] List of the past Gammapy hands-on sessions or schools
[15] List of the Gammapy general presentations

under a VHE data format, the `gamma-astro-data-format` (GADF). The light-weight Gammapy package is officially in use by several TeV facilities and has been chosen by the CTA Observatory to be the core library of its future open Science Analysis Tool. With the common data format GADF, it permits multi-instrument analyses by supporting joint fit of their data. This mature research software can be used for data from ASTRI, CTA, FACT, Fermi-LAT, HAWC, H.E.S.S., MAGIC, VERITAS. The LHAASO, SWGO, KM3NeT facilities are also testing and evaluating Gammapy. With more than 70 references to Gammapy and with its use for academic purposes, this software grows in popularity and is recognised as a major research software by the VHE community.

The Gammapy team released the first Long-Term Stable version, the v1.0, that offers users a maintained and stable package for their scientific analyses. Its stable user interface associated with minor bug releases offers a comfortable framework for physicists. This LTS offers all the standard and historical VHE algorithms, as well as algorithms coming from other astrophysics domains like the data cube analysis. It also allows a proper statistical treatment of joint analyses of high-level data of different types.

In parallel, the project developers team is working to improve the capabilities of the package, e.g. with the support of Bayesian analysis or unbinned analysis. Until the release of the new LTS, several feature releases are expected and will contain new functionalities and scientific algorithms. It is also expected that the high-level data format will evolve as the GADF initiative has ended. Eleven gamma-ray and neutrino experiments have created a new initiative, called *Very-high-energy Open Data Format* (VODF)[16] (see [17] for a complete description), aiming to create new standards for VHE astrophysics detectors that respect the FAIR principles [18] and follow as much as possible the IVOA[17] standards. In this context, the internal data model of Gammapy is being separated from the read data model with the use of an I/O layer dealing with formats and their versions.

To conclude, the Gammapy team is continuing to improve the open package by offering even more functionalities. As an open project, we encourage any reader to participate to the improvement of the package or to the animation of schools, while taking a special care to recognise any effort. The authors acknowledge again all contributors to Gammapy and all users for their valuable feedback.

References

[1] C. Deil, et al., *Gammapy - A prototype for the CTA science tools*, in proceedings of the *35th ICRC*, Busan, South Korea, PoS(ICRC2017)766, arXiv:1709.0175 [`doi:10.5281/zenodo.4701488`].

[2] C. Deil, et al., *Open high-level data formats and software for gamma-ray astronomy*, in proceedings of *the Gamma 2016 conference*, arXiv:1610.01884 [`doi:10.5281/zenodo.7304668`].

[3] D. C. Wells, E. W. Greisen, and R. H. Harten, *FITS - a Flexible Image Transport System*, Astronomy and Astrophysics Supplement *44*, 363 (1981).

[4] C. R. Harris, et al., *Array programming with NumPy*, Nature, *585*, 357 (2020) [`arXiv:2006.10256`].

---

[16] VODF web site: https://vodf.readthedocs.io/

[17] IVOA web site: https://www.ivoa.net/

[5] Astropy Collaboration, *Astropy: A community Python package for astronomy*, A&A, *558*, A33 (2013) [arXiv:1307.6212].

[6] P. Virtanen, et al., *SciPy 1.0: fundamental algorithms for scientific computing in Python*, Nature Methods *17*, 261–272 (2020) [arXiv:1907.10121].

[7] H. Dembinski, P. Ongmongkolkul, et al., *scikit-hep/iminuit*, Zenodo (2020) [doi:10.5281/zenodo.3949207].

[8] J. D. Hunter, et al., *Matplotlib: A 2D Graphics Environment*, Computing In Science & Engineering, *9*, 90 (2007).

[9] CTAO Observatory Press release, *CTAO Adopts the Gammapy Software Package for Science Analysis*, *CTAO News* (2021).

[10] French ministry of higher education, research and innovation, *Jury prize of open science awards for open source research software*, *News from the french research ministry* (2022).

[11] A. Albert, et al., *Validation of standardized data formats and tools for ground-level particle-based gamma-ray observatories*, A&A *667* A36 (2022) [arXiv:2203.05937].

[12] D. Berge, S. Funk and J. Hinton, *Background Modelling in Very-High-Energy gamma-ray Astronomy*, A&A *466* 3 (2007) [astro-ph/0610959].

[13] V. Zabalza, *naima: a Python package for inference of relativistic particle energy distributions from observed nonthermal spectra*, Proc. of *International Cosmic Ray Conference* 2015 [arXiv:1509.03319].

[14] T. Hassan, et al., *Performance of a proposed event-type based analysis for the Cherenkov Telescope Array*, Proc. of *International Cosmic Ray Conference* 2021 [arXiv:2108.10135].

[15] P. Moritz, et al., *Ray: A Distributed Framework for Emerging AI Applications*, Proc. of the *31th USENIX Symposium on Operating Systems Design and Implementation* (2018) [arXiv:1712.05889].

[16] M. Barker, et al., *Introducing the FAIR Principles for research software*, Scientific Data *9*, 622 (2022).

[17] B. Khélifi, et al., *The Very-high-energy Open Data Format: towards a shared, open data format in very-high-energy astronomy*, to be appeared in the Proc. of the *28th International Cosmic Ray Conference* (2023).

[18] M. D. Wilkinson, et al., *The FAIR Guiding Principles for scientific data management and stewardship*, Scientific Data *3*, 160018 (2016).

PoS(ICRC2023)959