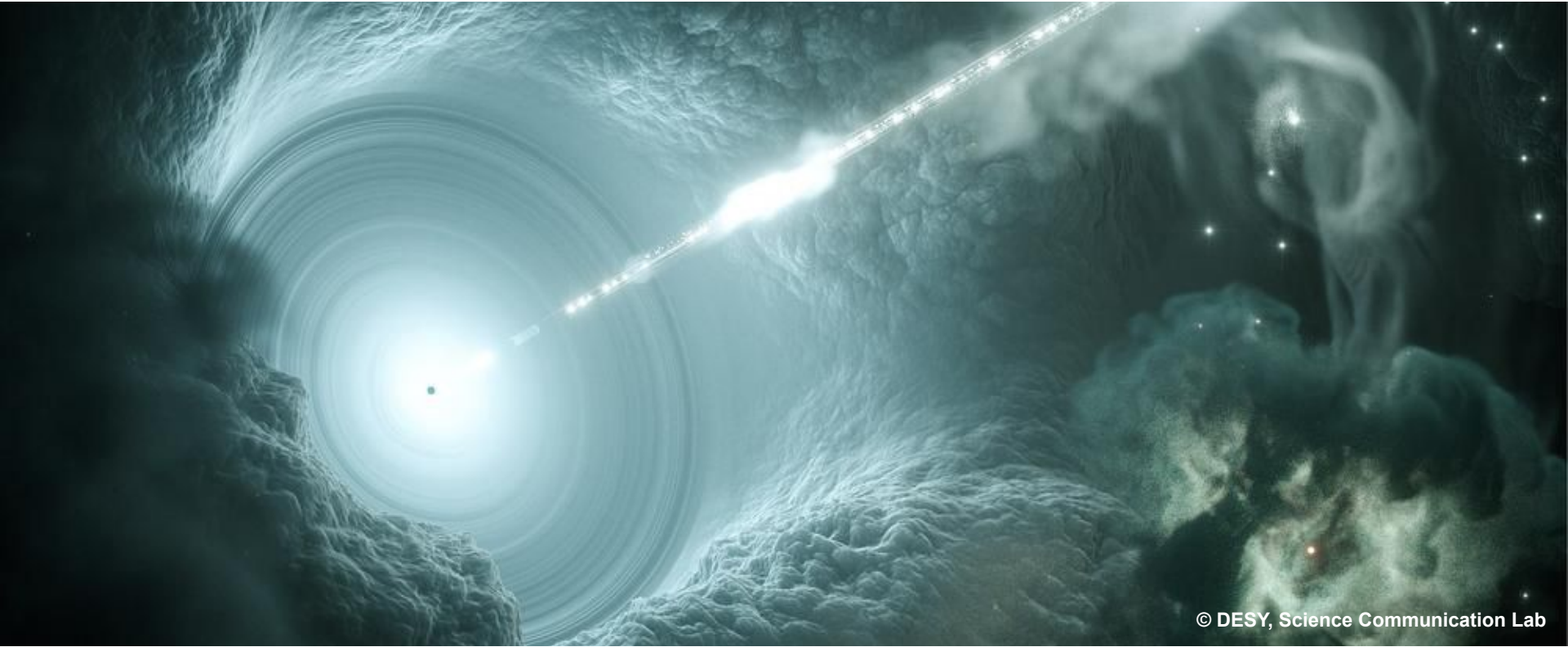




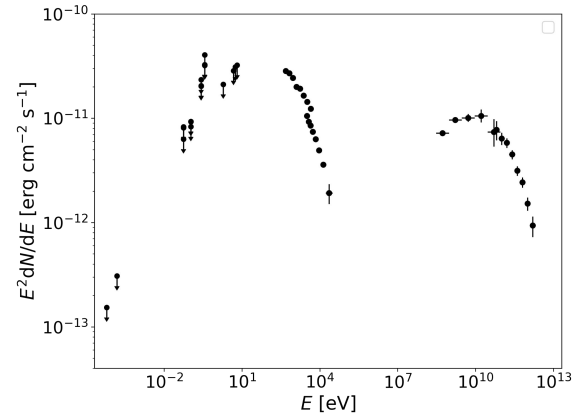
From Photons to Neutrinos: Advancing Multi-Messenger Blazar Modeling with Gammapy

Lea Heckmann, Justin Albinet, Matteo Cerruti, Bruno Khélifi



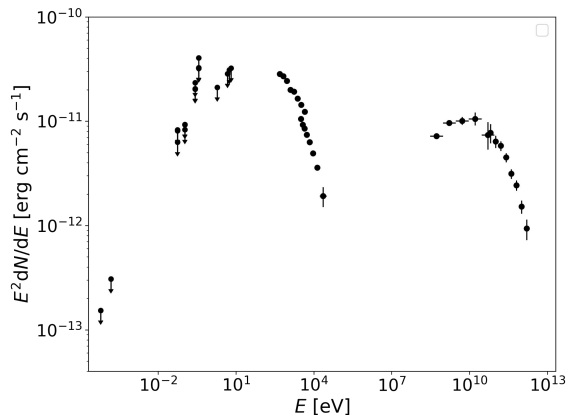
Current AGN modelling

- Combining MWL data on flux point level:



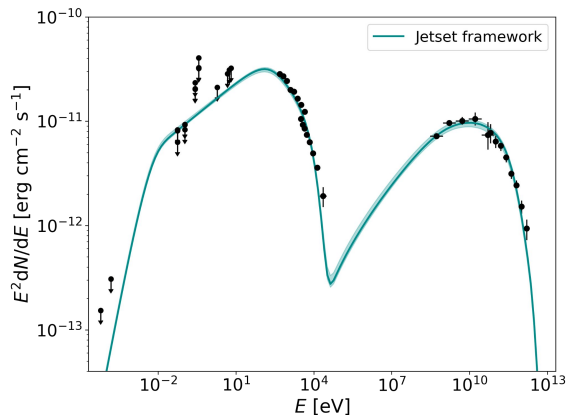
Current AGN modelling

- Combining MWL data on flux point level:
 - We construct flux points for single instruments separately
 - using different frameworks for different instruments
 - assuming simplistic models (e.g. power-law) – single instruments cannot constrain physical models (e.g. SSC)



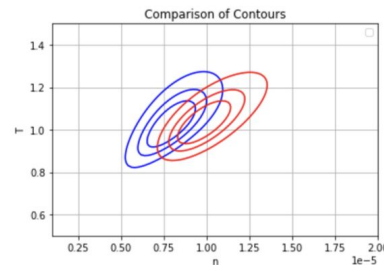
Current AGN modelling

- Combining MWL data on flux point level:
 - We construct flux points for single instruments separately
 - using different frameworks for different instruments
 - assuming simplistic models (e.g. power-law) – single instruments cannot constrain physical models (e.g. SSC)
 - Selection of the physical model based on flux points



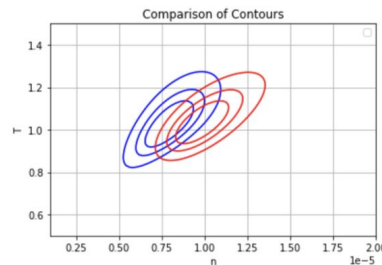
Current AGN modelling

- Combining MWL data on flux point level:
 - We construct flux points for single instruments separately
 - using different frameworks for different instruments
 - assuming simplistic models (e.g. power-law) – single instruments cannot constrain physical models (e.g. SSC)
 - Selection of the physical model based on flux points
- Fitting on flux points adds a bias:
 - Simple test with a blackbody spectrum using X-rays (XSPEC [\(1999ascl.soft10005A\)](#)) shows a bias of 1 sigma



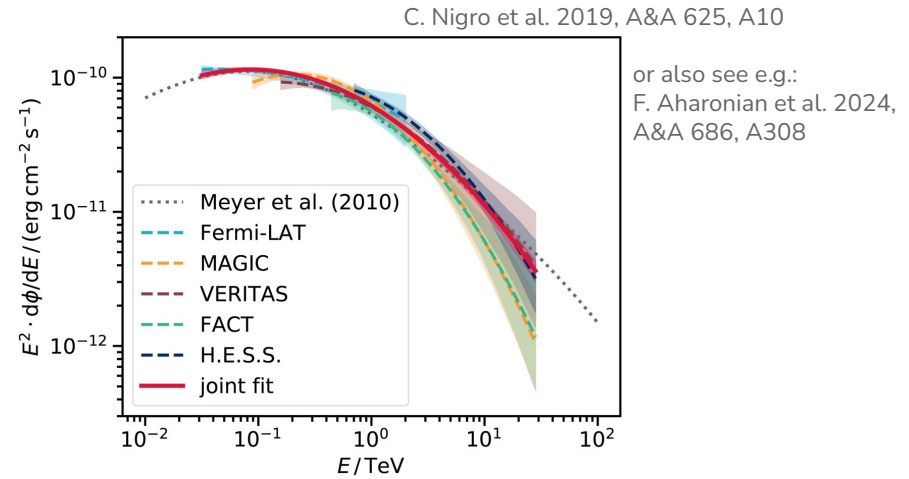
Current AGN modelling

- Combining MWL data on flux point level:
 - We construct flux points for single instruments separately
 - using different frameworks for different instruments
 - assuming simplistic models (e.g. power-law) – single instruments cannot constrain physical models (e.g. SSC)
 - Selection of the physical model based on flux points
- Fitting on flux points adds a bias:
 - Simple test with a blackbody spectrum using X-rays (XSPEC [\(1999ascl.soft10005A\)](#)) shows a bias of 1 sigma
- **Long-term goal: Enable event-based multi-messenger fits in Gammapy**
 - We start with blazars and JetSet as an example here



Gammapy & MWL data

- Works nicely for gamma-ray data and joint instrument fits:

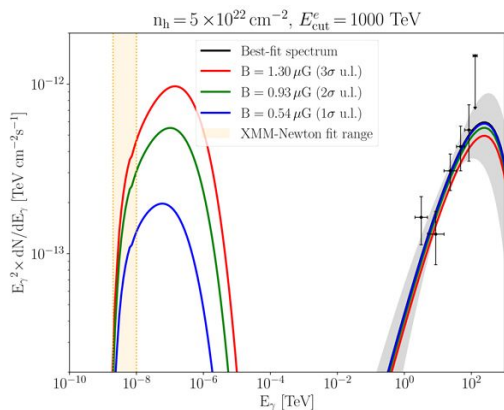


Gammapy & MWL data

- Works nicely for gamma-ray data and joint instrument fits:

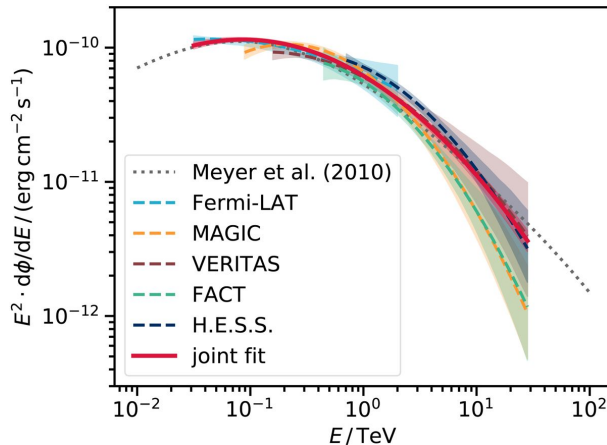
- Extension to X-rays:

<https://github.com/registerrier/gammapy-ogip-spectra>



L. Giunti et al. 2022, A&A 667, A130

C. Nigro et al. 2019, A&A 625, A10



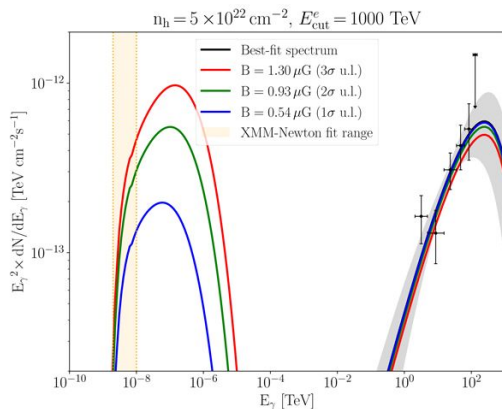
or also see e.g.:
F. Aharonian et al. 2024,
A&A 686, A308

Gammapy & MWL data

- Works nicely for gamma-ray data and joint instrument fits:

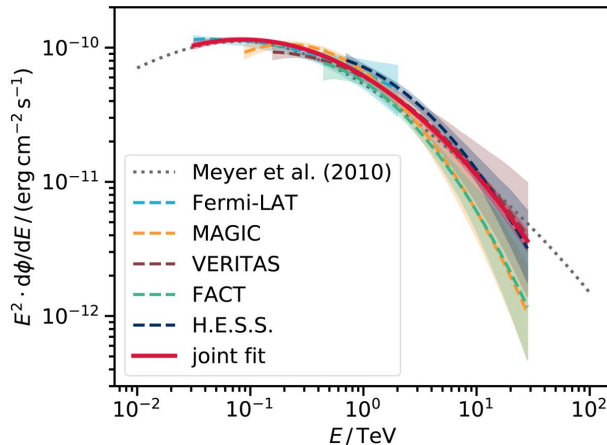
- Extension to X-rays:

<https://github.com/registerrier/gammapy-ogip-spectra>



L. Giunti et al. 2022, A&A 667, A130

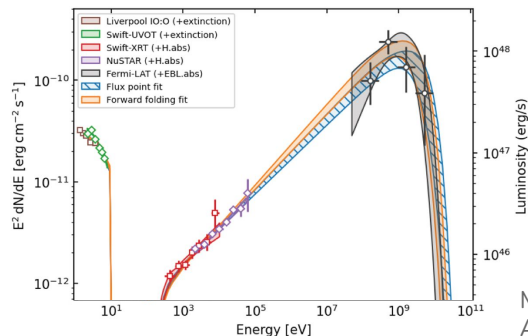
C. Nigro et al. 2019, A&A 625, A10



or also see e.g.:
F. Aharonian et al. 2024,
A&A 686, A308

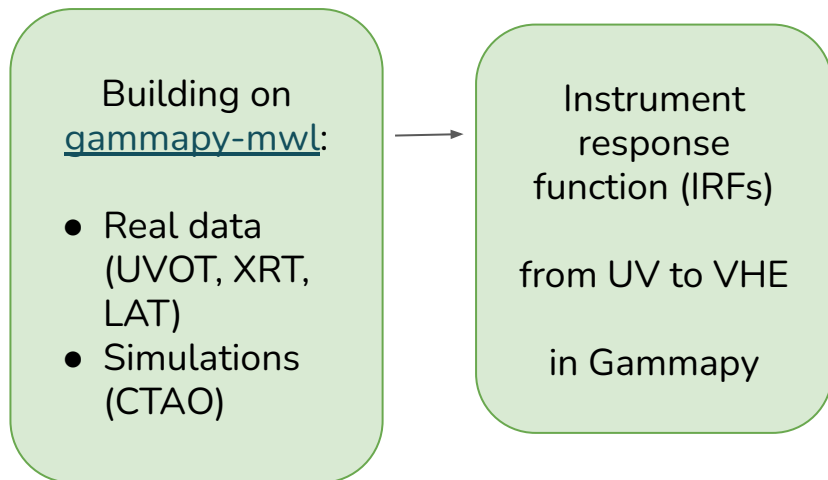
- Extension to UV/ optical:

https://github.com/mireianievas/gammapy_mwl_workflow

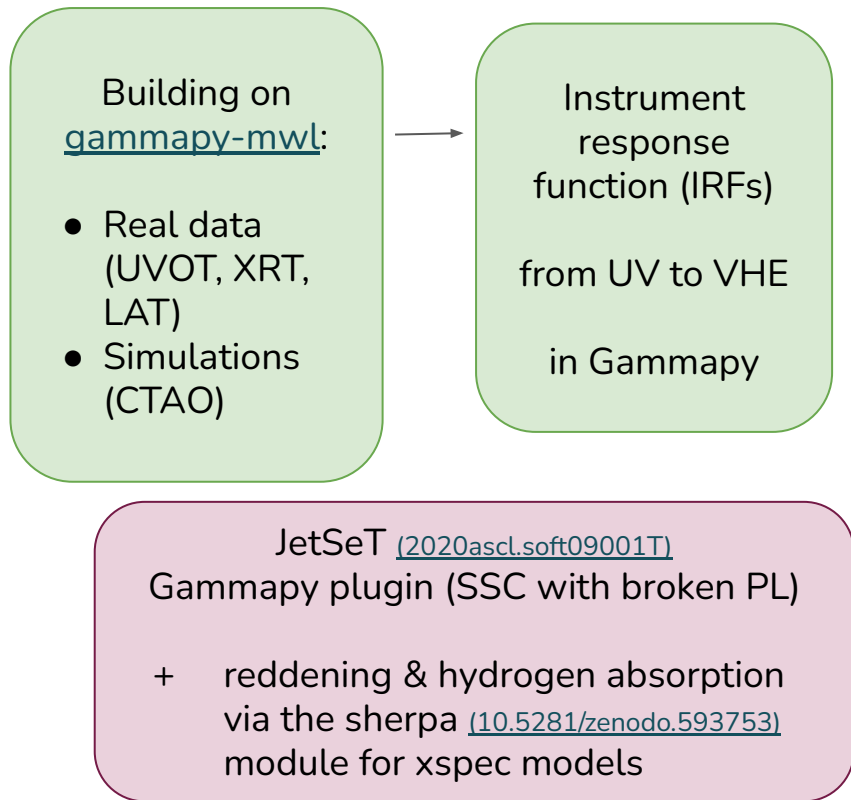


M. Nieves Rosillo et al. 2025,
A&A, 693, A287

New AGN modelling approach with Gammapy



New AGN modelling approach with Gammapy



New AGN modelling approach with Gammapy

Building on
[gammapy-mwl](#):

- Real data (UVOT, XRT, LAT)
- Simulations (CTAO)



Instrument
response
function (IRFs)

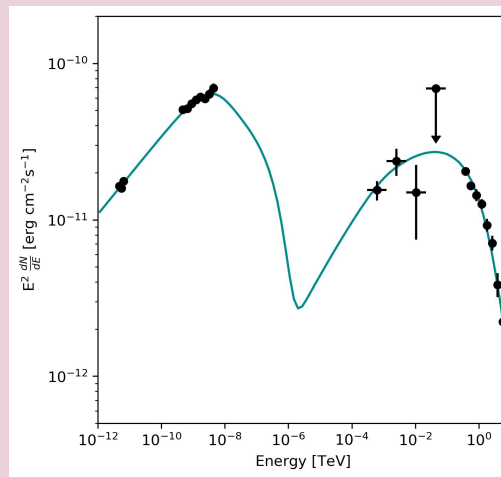
from UV to VHE

in Gammapy

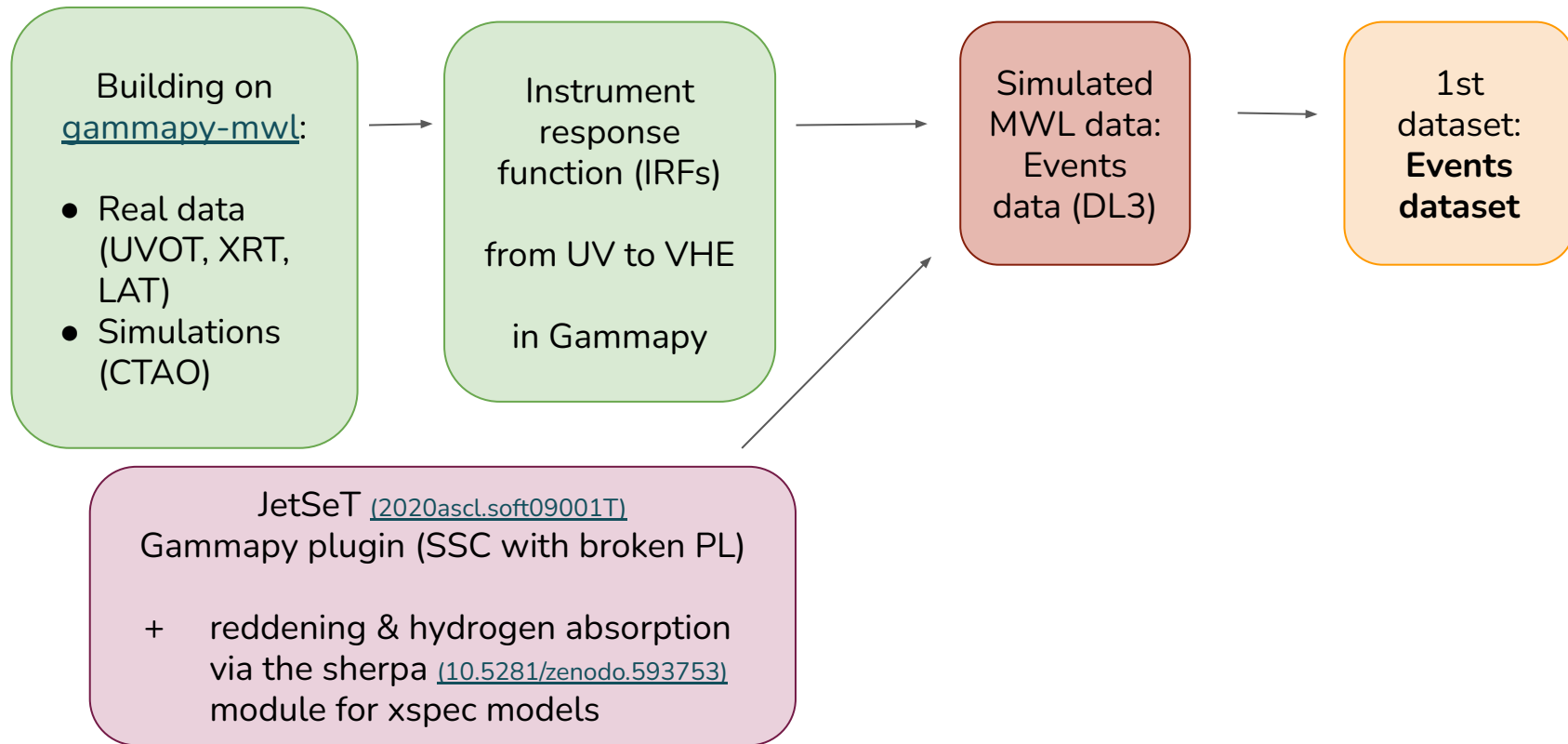
JetSeT ([2020ascl.soft09001T](#))
Gammapy plugin (SSC with broken PL)

+ reddening & hydrogen absorption
via the sherpa ([10.5281/zenodo.593753](#))
module for xspec models

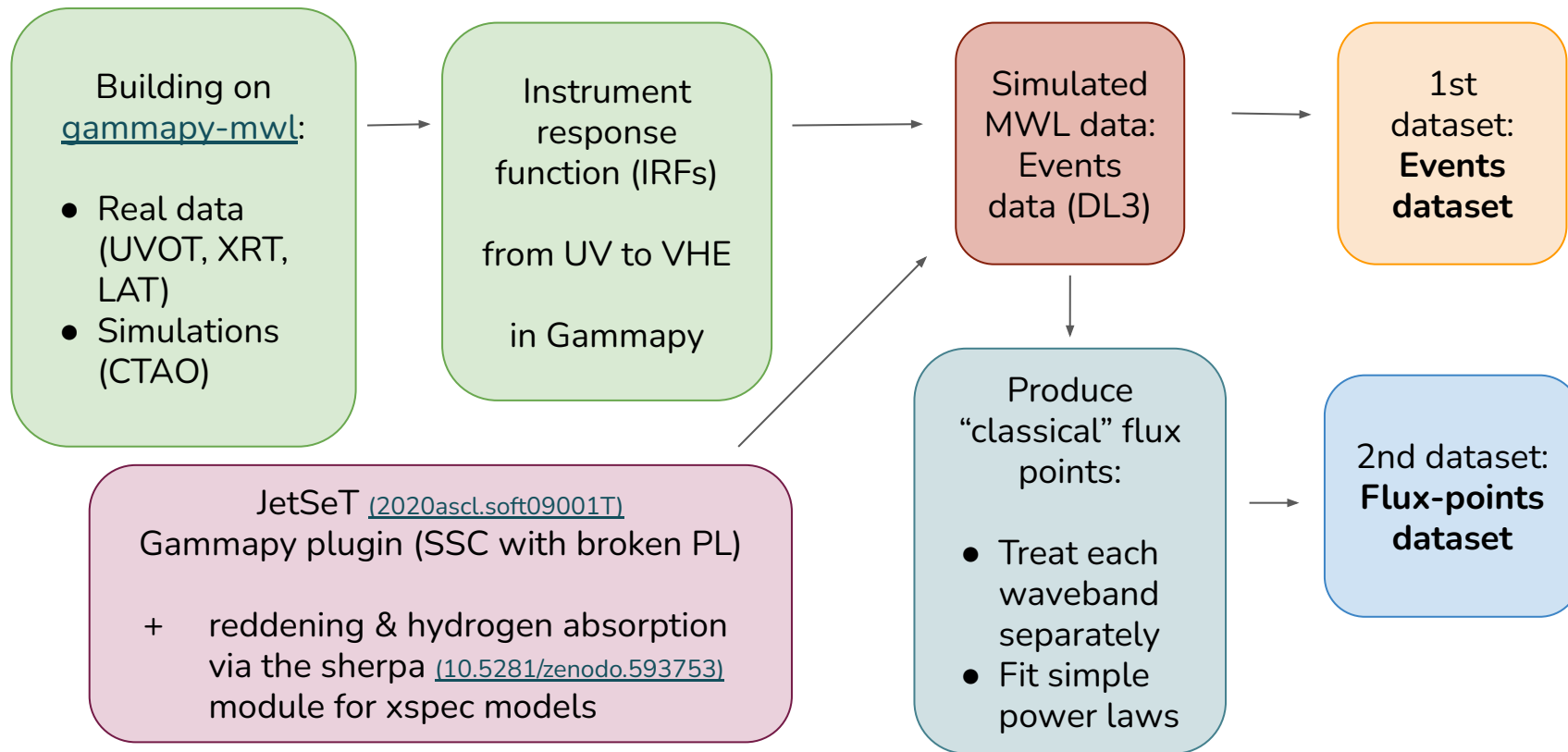
First example use case
presented here:
bright EBL



New AGN modelling approach with Gammapy



New AGN modelling approach with Gammapy



New AGN modelling approach with Gammapy

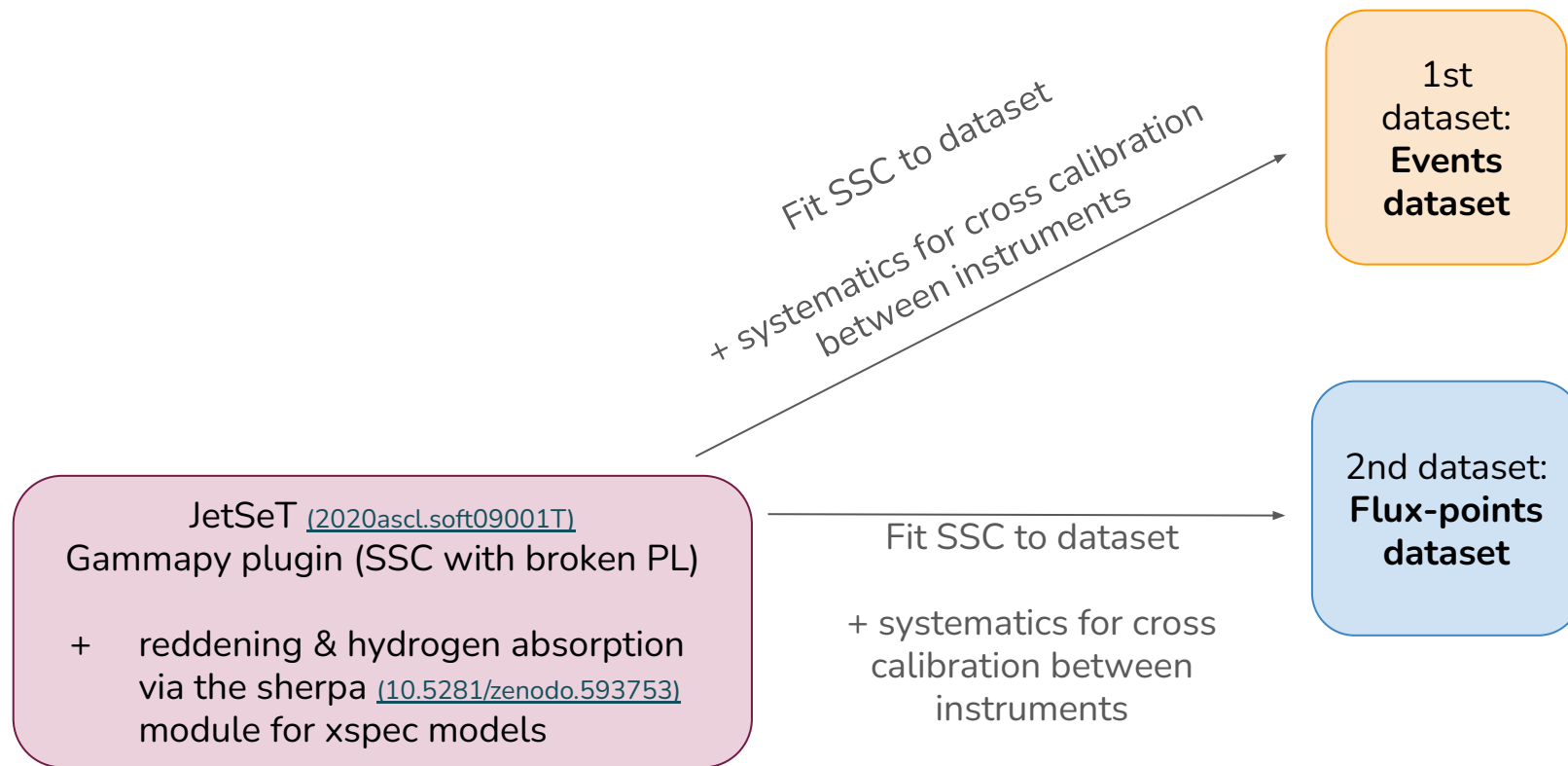
1st
dataset:
**Events
dataset**

2nd dataset:
**Flux-points
dataset**

JetSeT ([2020ascl.soft09001T](https://arxiv.org/abs/2020ascl.soft09001T))
Gammapy plugin (SSC with broken PL)

+ reddening & hydrogen absorption
via the sherpa ([10.5281/zenodo.593753](https://doi.org/10.5281/zenodo.593753))
module for xspec models

New AGN modelling approach with Gammapy



Gammapy & SSC

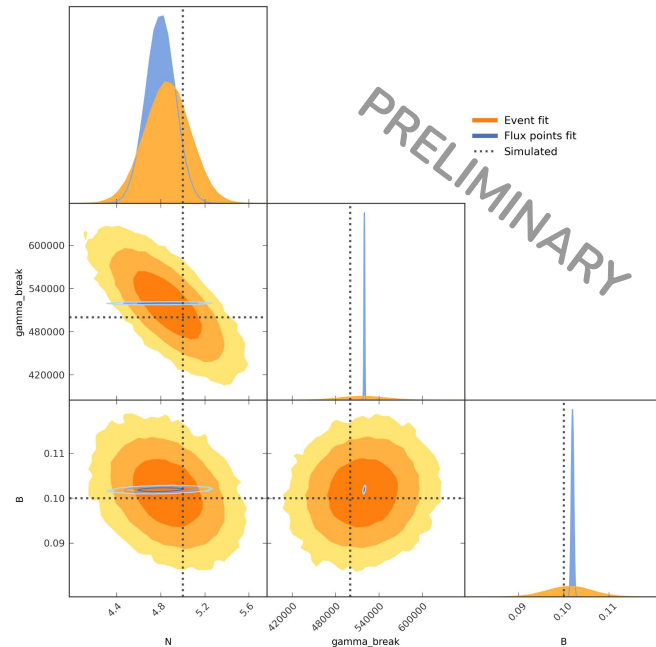
Status last year:

- SSC fits with jetset: 3-5 parameter fits
 - Flux-points fits 3-5x faster
- Events fits show better convergence and less bias

Gammapy & SSC

Status last year:

- SSC fits with jetset: 3-5 parameter fits
 - Flux-points fits 3-5x faster
- Events fits show better convergence and less bias

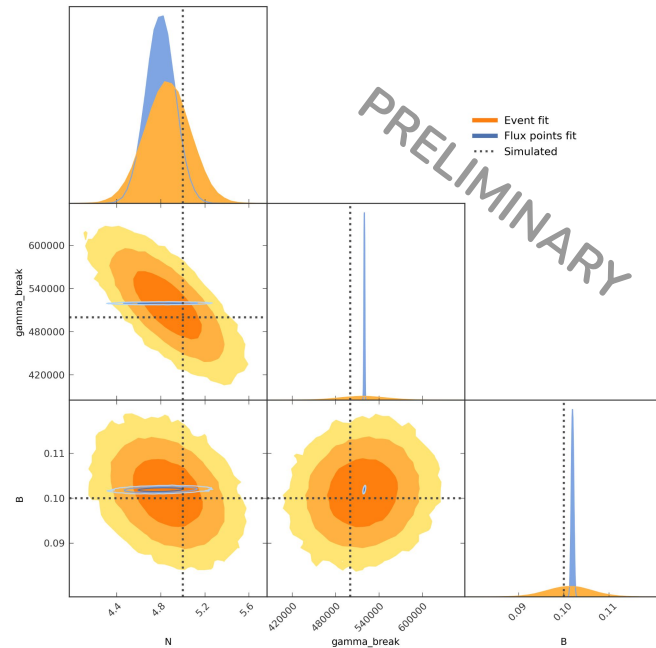


→ now updated to gammapy 2.0 & starting parameters further away from the true value

- ~16 min for events fits for 3 free parameters
- ~10 min for flux fits for 3 free parameters

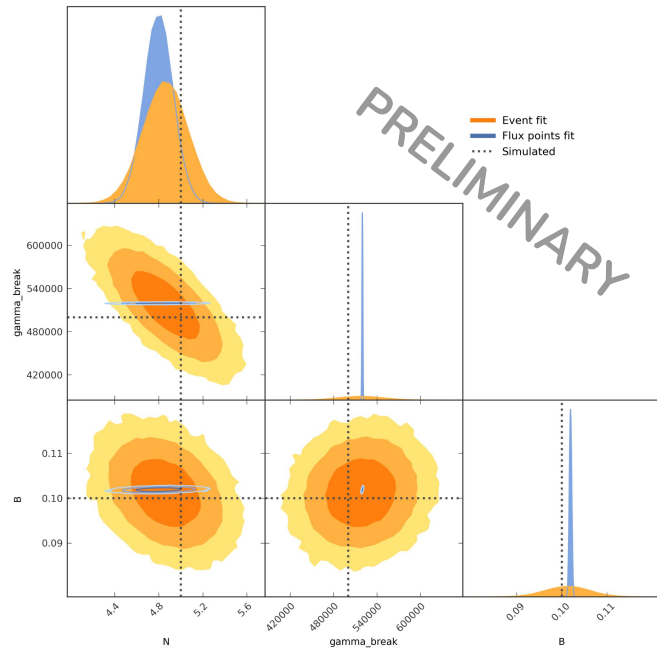
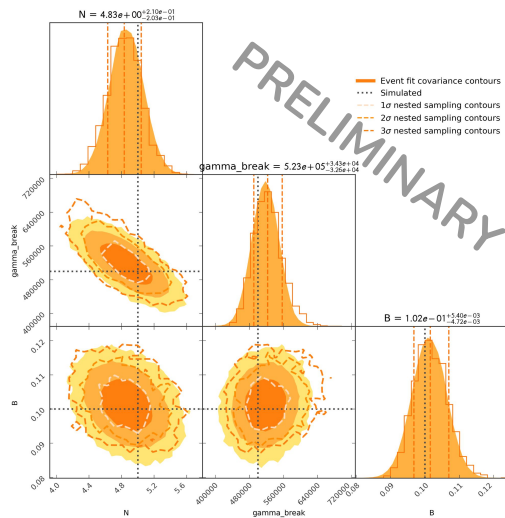
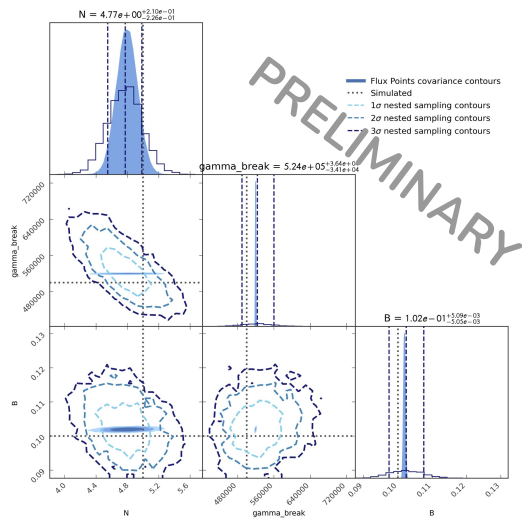
Gammapy & SSC

- We checked minuit contours:
 - Events fits contours are trustable
 - Flux-points fits contours are not



Gammapy & SSC

- We checked minuit contours:
 - Events fits contours are trustable
 - Flux-points fits contours are not
 - but can be recovered with nested sampling (but still a bigger bias)



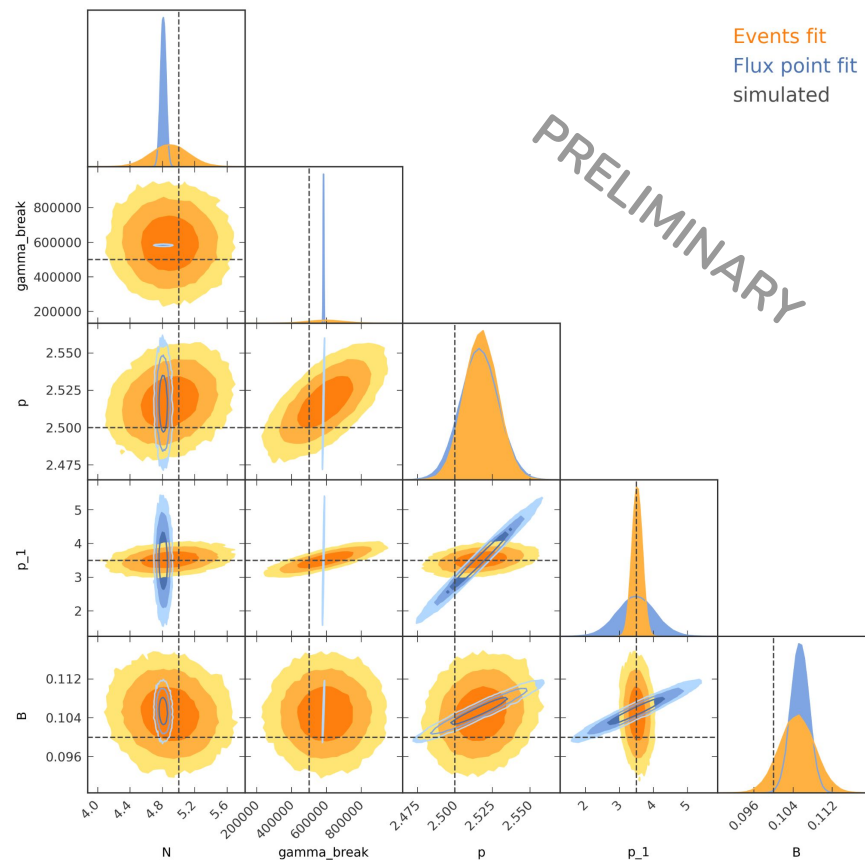
Nested sampling timing:

- Flux points: ~28 h
- Events: ~3 days

→ Events fit faster if one wants trustable uncertainties (no need for nested sampling)

Gammapy & SSC

- Fits with 5 free parameters:
 - Flux-points fit does not converge
 - Events fit converges in 45 min
 - Nested sampling takes more than 4 days



Grid interpolation

- Timing problem:
 - Results so far - fast models: JetSet timing inside Gammapy at one E: 0.02 sec
 - Slower/more complex models (e.g. hadronic) > 1 min → cannot be run on the fly

Grid interpolation

- Timing problem:
 - Results so far - fast models: JetSet timing inside Gammapy at one E: 0.02 sec
 - Slower/more complex models (e.g. hadronic) > 1 min → cannot be run on the fly
- Work-around methods:
 - Production of model grid + interpolation
 - Replaces the JetSeT Gammapy plugin model
 - 10 grid points per parameters, 200 in energy

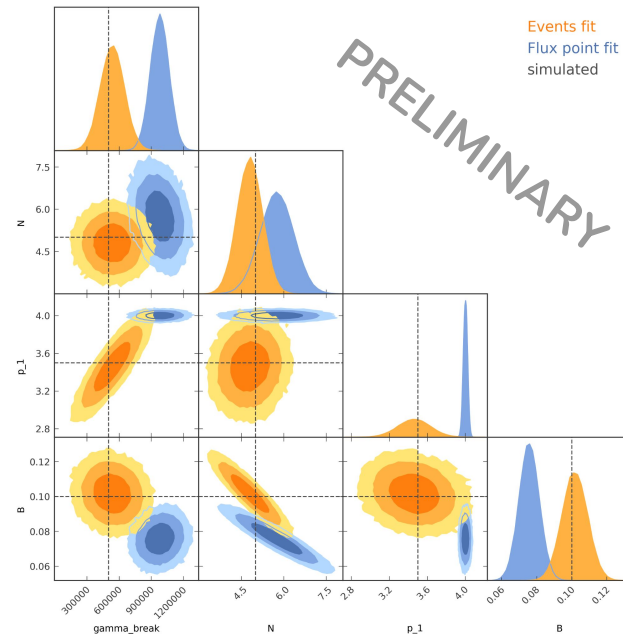
Grid interpolation

- Timing problem:
 - Results so far - fast models: JetSet timing inside Gammapy at one E: 0.02 sec
 - Slower/more complex models (e.g. hadronic) > 1 min → cannot be run on the fly
- Work-around methods:

Production of model grid + interpolation

→ Replaces the JetSeT Gammapy plugin model

 - 10 grid points per parameters, 200 in energy
 - Seems to work well for events fits
 - Clear bias for flux-points fit
 - Timing 4-parameter fit:
~14 h for events fit, ~22 h for flux-points fit
 - BUT overload memory for many free parameters

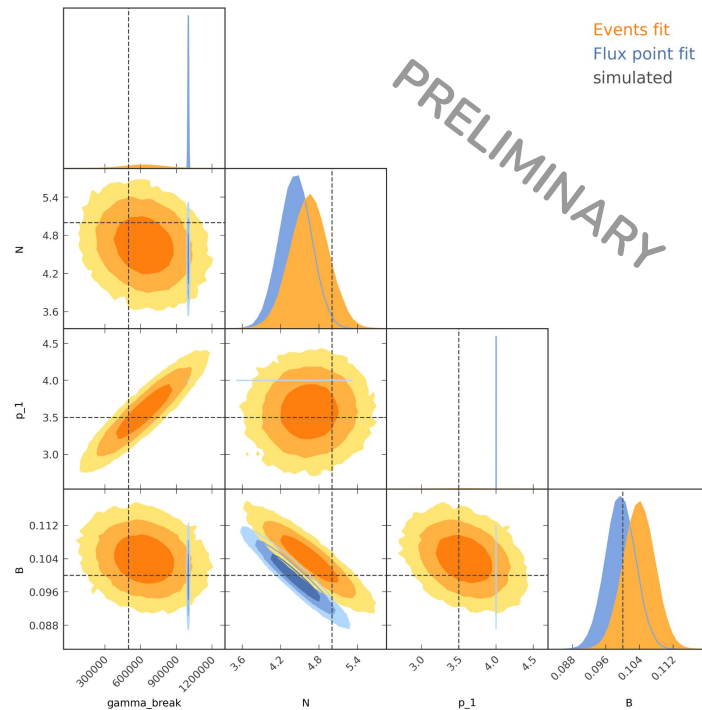


Grid interpolation

- Second work-around:
 - Move away from fits files: So far we use the local model format from xspec
 - But fits load the full HDU in memory
 - HDF5 file format [10.5281/zenodo.17808614](https://zenodo.org/record/17808614)
 - Index file + Data file
 - Subset of data in memory every fit run

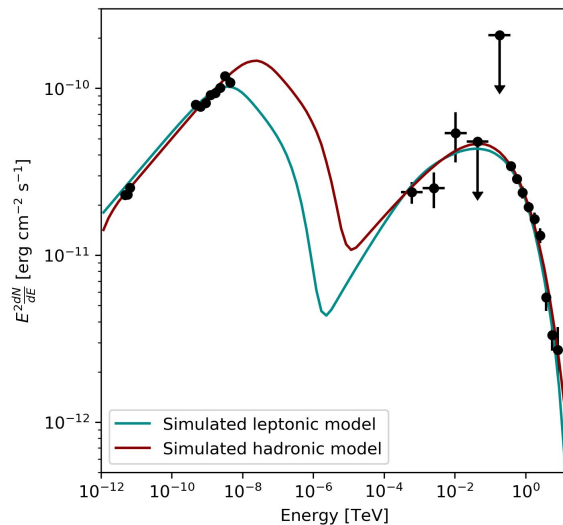
Grid interpolation

- Second work-around:
 - Move away from fits files: So far we use the local model format from xspec
 - But fits load the full HDU in memory
 - HDF5 file format [10.5281/zenodo.17808614](https://zenodo.org/record/17808614)
 - Index file + Data file
 - Subset of data in memory every fit run
 - Can produce grids up to 7 parameters + potential to write in parallel
 - Timing 4-parameter fit:
~7 h for events fit, ~14 h for flux-points fit



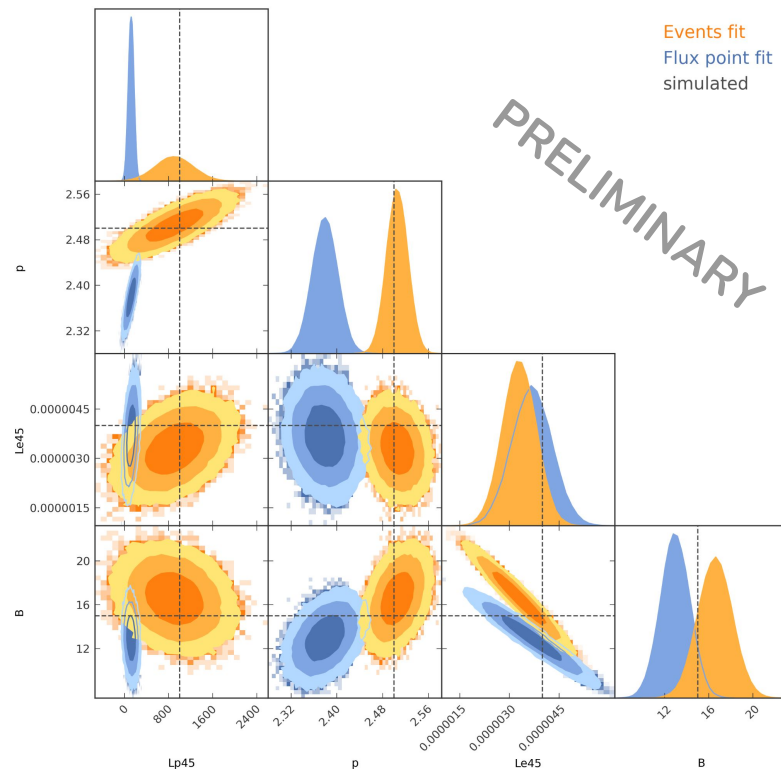
Hadronic fits

- Using the LeHa code [10.1093/mnras/stu2691](https://doi.org/10.1093/mnras/stu2691) instead of JetSeT for grid production
- p-Synchrotron model covering the same flux points as the SSC model



Hadronic fits

- Using the LeHa code [10.1093/mnras/stu2691](https://doi.org/10.1093/mnras/stu2691) instead of JetSeT for grid production
- p-Synchrotron model covering the same flux points as the SSC model
- Production of a hdf5 grid
- Timing 4-parameter fit:
~5 h for events fit, ~6 h for flux-point fit



Lessons learnt so far

- **Gammapy** enables us to **fit physical models directly on events**
 - Reducing biases and assumptions
 - Taking into account absorption, systematics or e.g. also ULs naturally
 - But we are still limited by the large amount of parameters
- For **fast models (e.g. leptonic ones)**
 - Fitting can be run by Gammapy on the fly
 - We also investigate various fitting methods and Bayesian nested sampling on events (limited by the Fermi analysis) → minuit preferred currently
- For **slow models (e.g. hadronic ones) – multimessenger fitting**
 - We are currently investigating grid models + interpolation methods

Outlook

- **Technical checks to be done**
 - Use different SED types (LBL, IBL, HBL & different luminosities)
 - Try N realization for each model
 - Summarize convergence, precision etc. for the different methods
- **Multimessenger fitting**
 - First neutrino analysis will soon be implemented in Gammapy (Km3Net)
 - Machine learning methods might be a promising alternative for slow models
- **Discussion points for the community:**
 - Model format - fits, hdf5,...
 - Type of machine learning - CNN, RNN, transformer,...
 - Machine learning for model production or also for fitting?

Thank you for your attention!

