THE AEROSPACE CORPORATION

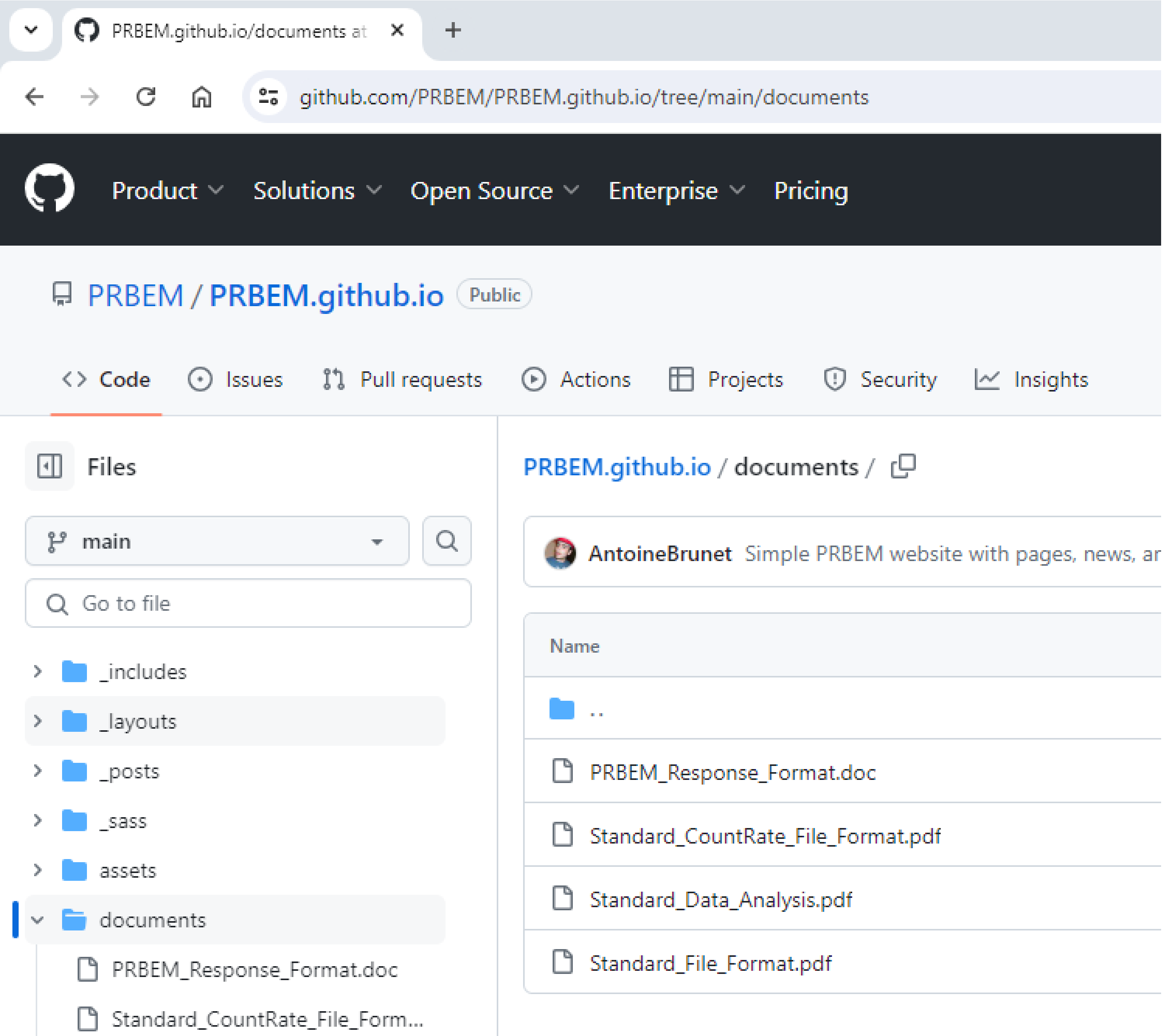
PRBEM sensor response format & response function library

Paul O'Brien
Space Sciences Department

17 July 2024

Introduction

- The PRBEM Response Function File Format provides a standard way to describe the response of a sensor to one or more species as a function of energy and sensor angles
- The PRBEM Response Function Library provides tools for generating and manipulating response functions to perform scientific analyses such as bowtie analysis, spectral inversion, angular inversion, and data assimilation
  - Fully available in Matlab
  - In development in Python



PRBEM Docs

There are several standards documents

- **PRBEM_Response_Format – specifies response file format**
- Standard_CountRate_File_Format – specifies count rate file (level 1)
- Standard_Data_Analaysis – specifies data analysis (cleaning) methods
- Standard_File_Format – specifies PRBEM flux file format (level 2)

Response file format

Top-level data in a response file:

- FORMAT_VERSION – string providing the version X.Y.Z for the response file format
- CHANNEL_NAMES – an array of strings given channel names
- SPECIES – an array of strings giving species names
- L_UNIT – length unit (e.g., 'cm')
- E_UNIT – energy unit (e.g., 'MeV')
- REFERENCES – list of paper references and websites (list of strings)

Default values of the channel-specific variables can be supplied as top level variables (without the leading period) or as species-specific variables (without a channel name or leading period)

Channel-specific data in a response file:

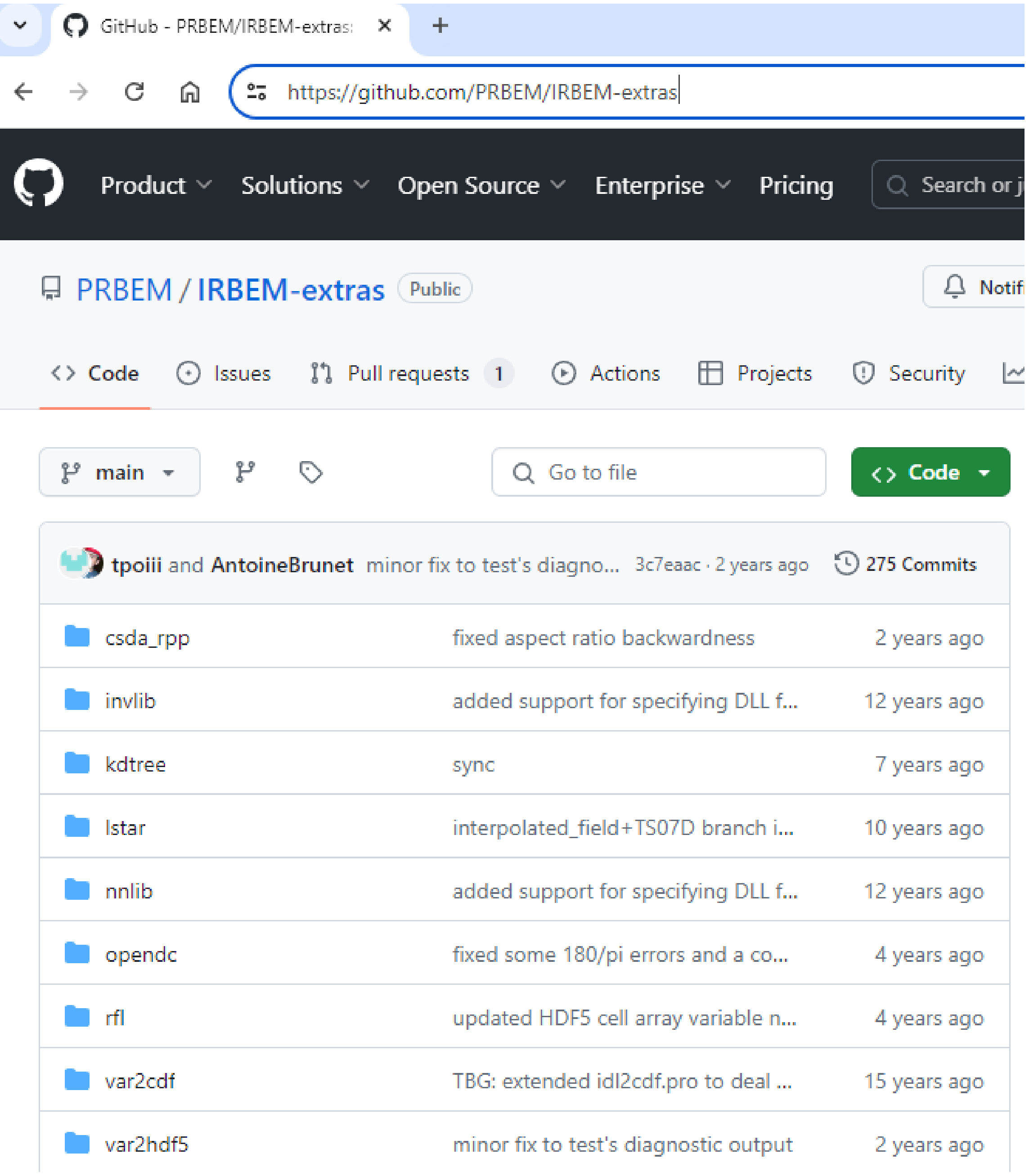
- DEAD_TIME_PER_COUNT – dead time per count, (sec)
- DEAD_TYPE – type of dead time (string, 'BLOCKING', or 'NONBLOCKING', 'AGGREGATE_BLOCKING', 'AGGREGATE_NONBLOCKING'") [At this time, description of AGGREGATE dead times is not fully defined because it involves some scheme of aggregating expected counts across multiple coupled data channels. For each coupled list of channels, there will be a variable called COUNTS_TOTAL_MAX – maximum counts across all channels (scalar)]
- COUNTS_MAX –maximum counts the data channel
- CROSSCALIB – cross calibration "fudge factor" (scalar)
- CROSSCALIB_RMSE – RMS error of the natural log of the counts for each channel, from calibration against a gold standard (scalar)

- TH_TYPE (optional) – type of THETA (polar angle) response
  - "TBL" – table, requires TH_GRID and A(THETA)
  - "PINHOLE" – delta function at THETA=0, requires G, BIDIRECTIONAL
  - "CYL_TELE" – cylindrical telescope, requires R1, R2, D, BIDIRECTIONAL
  - "DISK" – disk geometry, requires R1, BIDIRECTIONAL
  - "SLAB" – slab geometry, requires W1, H1, BIDIRECTIONAL

- The response file provides all the information needed to compute the sensor response as a function of incident energy and sensor angles
- Global metadata such as data units, grids, supported species, channel names
- For each data channel, it provides the response to one or more species
- Common idealized channel types are supported (e.g., integral energy channel, cylindrical telescope geometry)
- 1-, 2-, and 3-D tables are used for non-analytical aspects of response

- Format was originally CDF and .mat, but migrating to HDF5 because of its greater capacity and flexibility

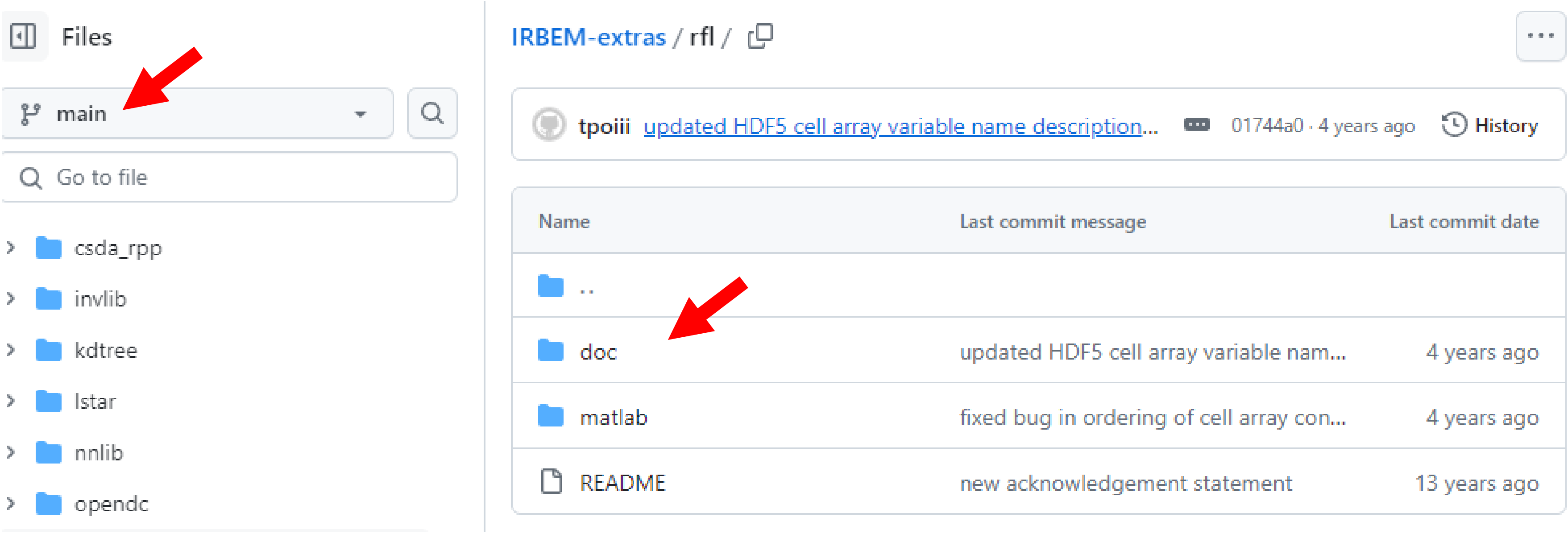
PRBEM Extras

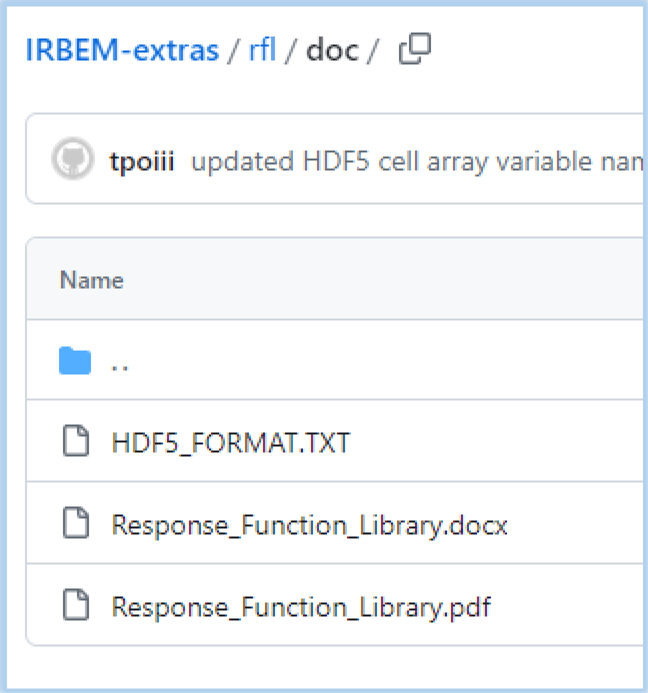


There are several independent “extras” available in the PRBEM project at github

- csda_rpp – single event effects modeling
- invlib – spectral and pitch angle inversions
- kdtree – fast multi-dimensional nearest neighbors lookup
- lstar – fast L* using grid approach
- nnlib – small neural network library
- opendc – open diffusion code & utilities
- **rfl – sensor response function library**
- var2cdf – share variables between languages via CDF
- var2hdf5 – share variables between languages via HDF5

RFL – Document (main branch)





- Response function library document describes the main equations for analytical models of sensor response, conversion between sensor and magnetic field angles, etc
- Has been used to provide sensor response for MagEIS, RPS, AeroCube-6, REACH, etc

RFL – Document, (main branch) continued

These quantities define the response function R below:

$$y \approx \lambda = \frac{1}{X_{cal}} \int_0^\infty \int_0^{2\pi} \int_0^\pi R(E,\theta,\phi) \int_{t_1}^{t_2} j(E,\alpha(\theta,\phi,t),\beta(\theta,\phi,t))dt \sin\theta d\theta d\phi dE$$

When energy and angular response are separable, we have:

$$y \approx \lambda = \frac{1}{X_{cal}} \int_0^\infty \varepsilon(E) \int_0^{2\pi} \int_0^\pi A_{eff}(\theta,\phi) \int_{t_1}^{t_2} j(E,\alpha(\theta,\phi,t),\beta(\theta,\phi,t))dt \sin\theta d\theta d\phi dE$$

We usually wish to compute weights h_{ijk}, such that the integral can be replaced by a sum:

$$y \approx \lambda \approx \sum_{ijk} h_{ijk} j(E_i,\alpha_j,\beta_k) = \sum_{ijk} h_{ijk} j_{ijk}$$

This is particularly helpful for inversion and data assimilation.

Manipulations to define h_{ijk} explicitly:

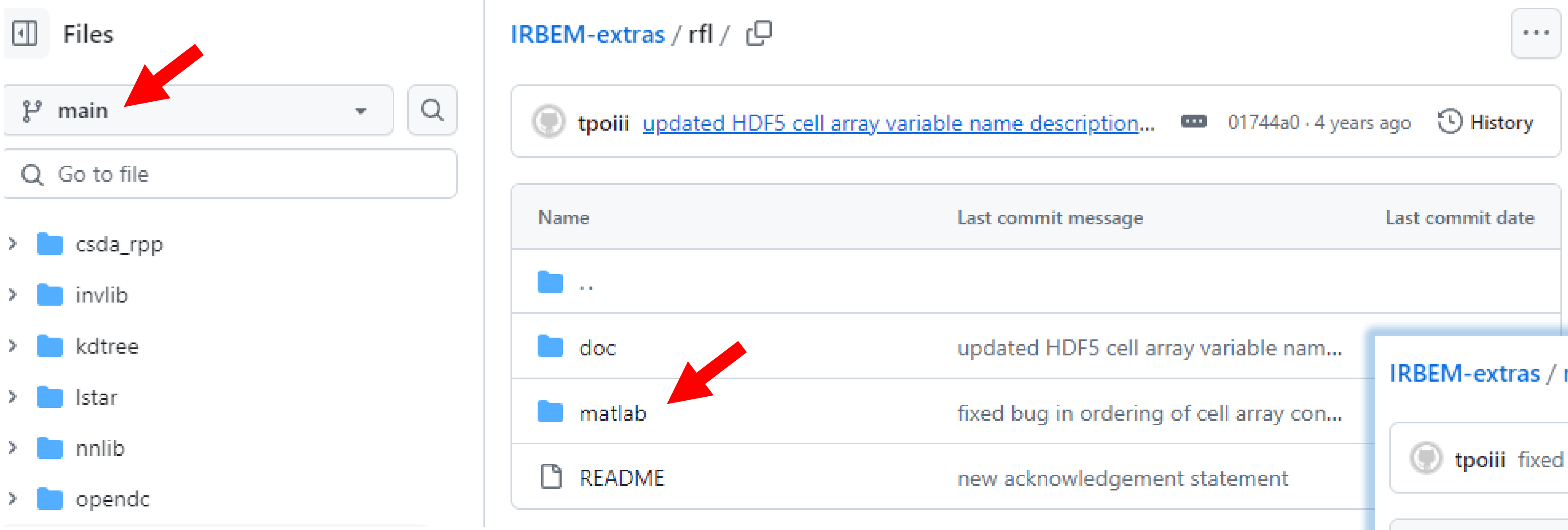
$$y \approx \lambda = \frac{1}{X_{cal}} \int_0^\infty \int_{-1}^1 \int_0^{2\pi} j(E,\alpha,\beta) \left[\int_{t_1}^{t_2} R(E,\theta(\alpha,\beta,t),\phi(\alpha,\beta,t))dt \right] d\beta d\cos\alpha dE$$

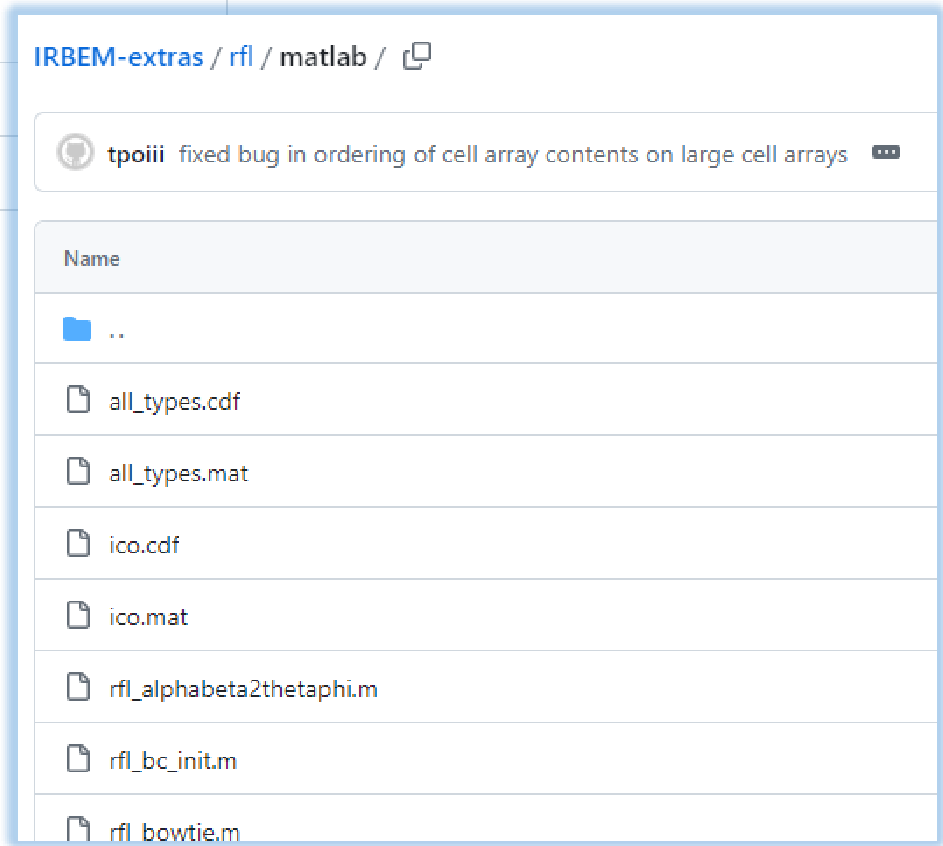
$$\approx \frac{1}{X_{cal}} \sum_{ijk} j_{ijk} \left[\int_{t_1}^{t_2} R(E_i,\theta(\alpha_j,\beta_k,t),\phi(\alpha_j,\beta_k,t))dt \right] \Delta\beta_k \Delta\cos\alpha_j \Delta E_i$$

$$h_{ijk} = \frac{1}{X_{cal}} \left[\int_{t_1}^{t_2} R(E_i,\theta(\alpha_j,\beta_k,t),\phi(\alpha_j,\beta_k,t))dt \right] \Delta\beta_k \Delta\cos\alpha_j \Delta E_i$$

- The response function describes the sensor response (effective area) as a function of energy and 2 sensor angles: θ (polar) and φ (azimuth)
- The file format allows various ways to specify this response function, from a 3-D table, to an idealized omnidirectional integral channel, and many approximations in between (e.g., two-element telescope)
- From a specified sensor response, it is possible to use the library to generate numerical integration weights as a 1-D, 2-D, or 3-D function/table in (E,θ,φ) or (E,α,β) systems
- These weights can be used and re-used in spectral inversion, angular inversion, data assimilation, and bowtie analysis

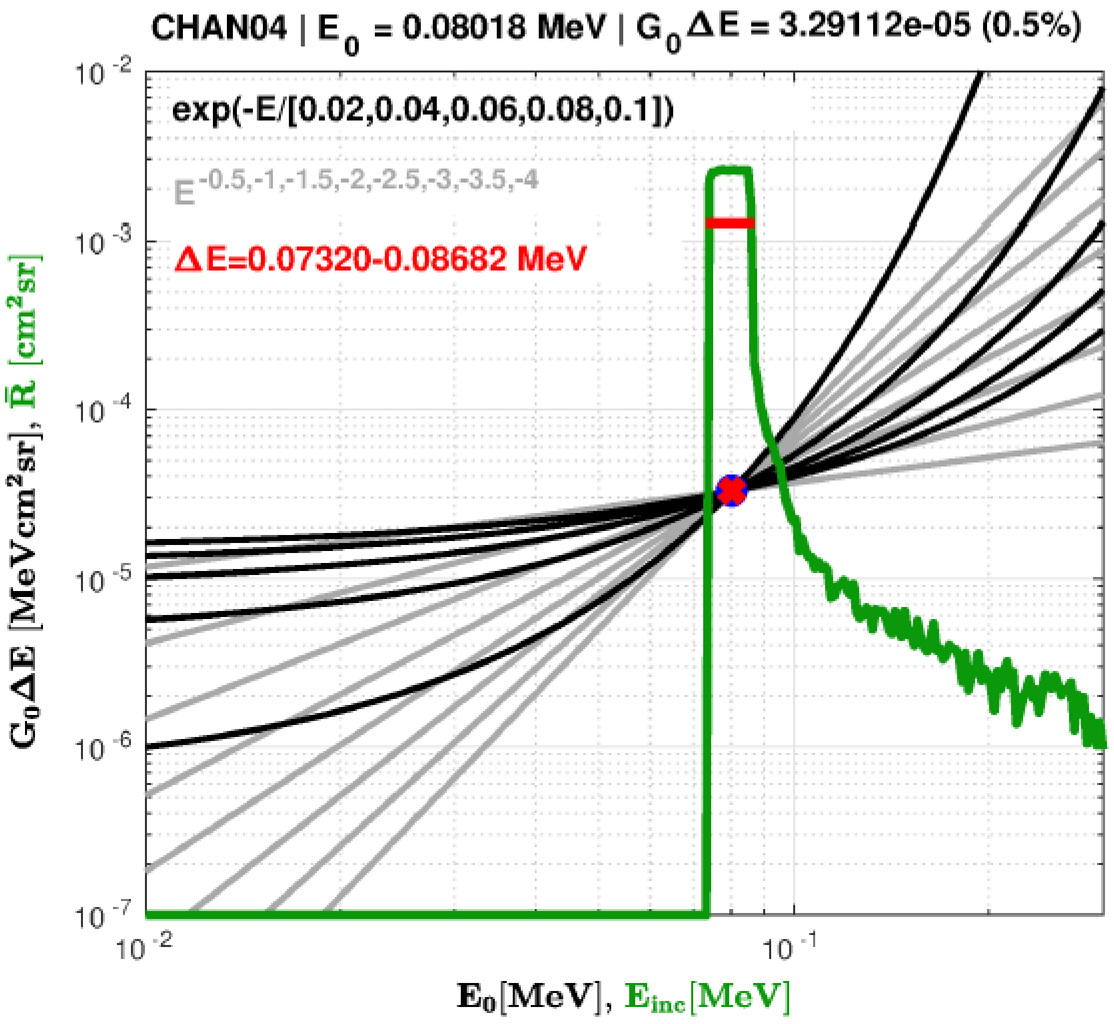
RFL – Matlab (main branch)



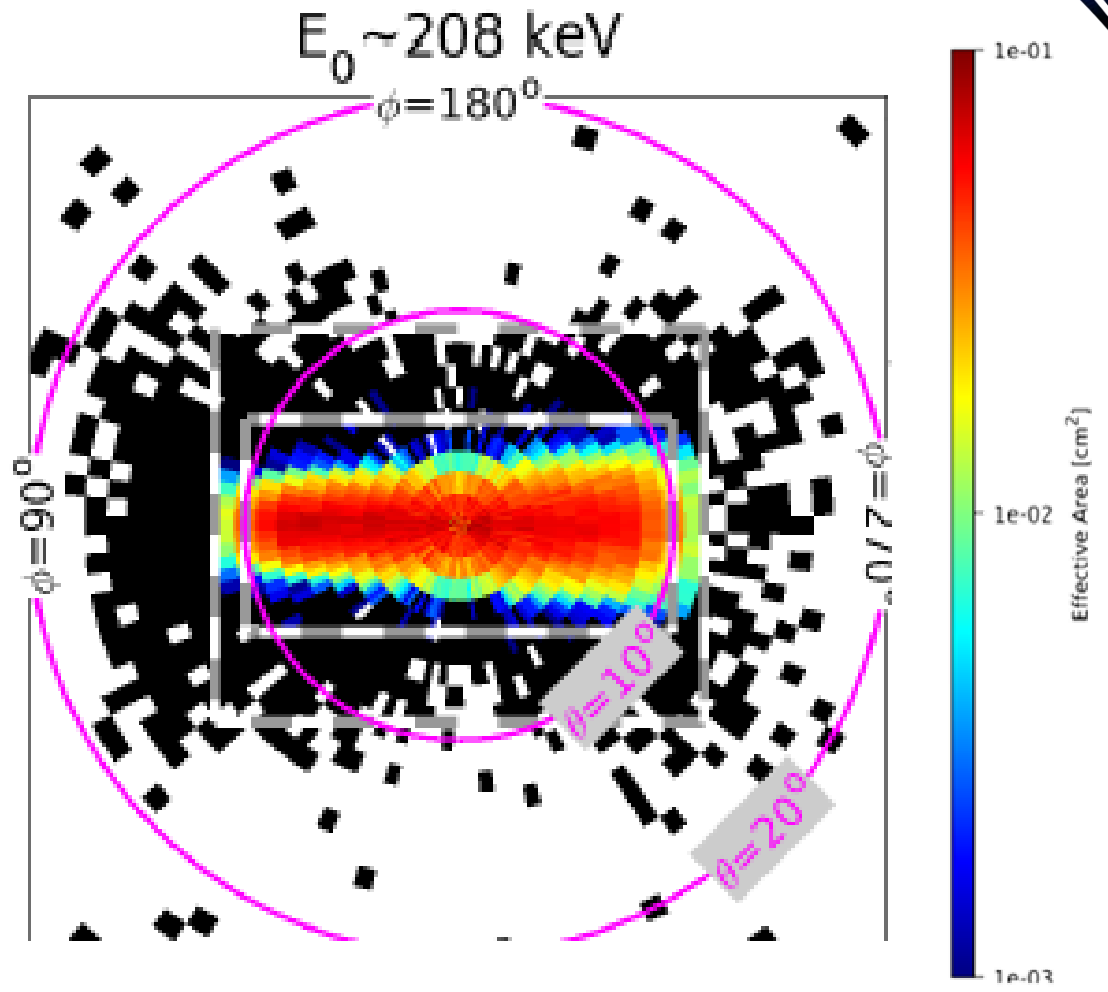


- Matlab code is mature and on the main branch
- Functions are provided to compute the numerical integration weights (trapezoidal method) for integrals over (E, α, β, t) or (E, θ, φ)
- Multiple examples, e.g., ICO (rfl_make_ico.m), SAMPEX PET (rfl_make_sampex_pet.m)
- Bowtie analysis (rfl_bowtie.m)
- Rotations from (α, β) to/from (θ, φ), Euler angles
- Load/save to CDF, HDF5, .mat response files
- Generate CSV energy response file for isotropic approximation

RFL – Matlab (main branch), applications



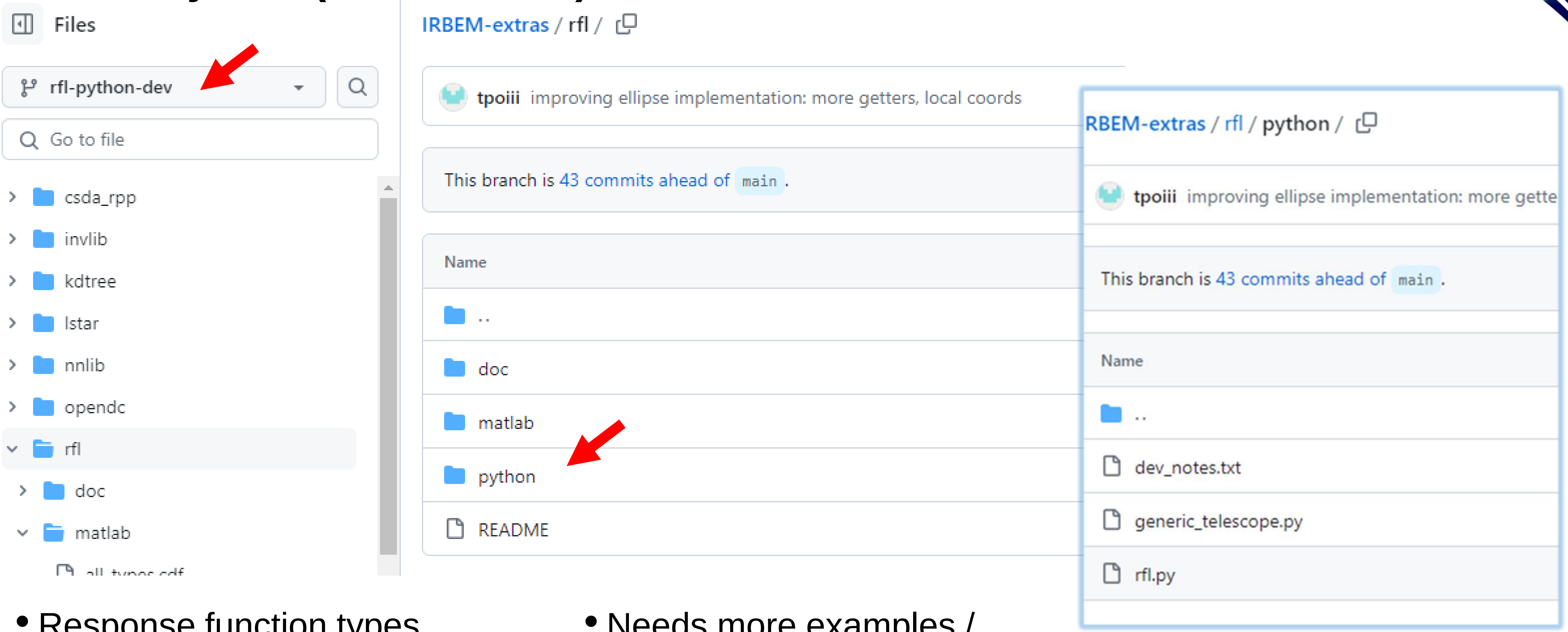
Bowtie analysis



Angular response at fixed energy

Figures from Claudpierre+2021 (ESM) https://doi.org/10.1007/s11214-021-00855-2, License CC BY 4.0

RFL – Python (main branch)



- Response function types implemented as class heirarchy
- HDF5 supported
- Started on generic multi-element telescope

- Needs more examples / cookbook
- Needs bowtie analysis
- Needs testing

Summary

- The Response Function File Format provides a standard way to describe the response of a sensor to one or more species as a function of energy and sensor angles
- The Response Function Library provides tools for generating and manipulating response functions to perform scientific analyses such as bowtie analysis, spectral inversion, angular inversion, and data assimilation
  - Fully available in Matlab
  - In development in Python

Abstract

Abstract: The PRBEM response function file format can be used to describe the 1-D, 2-D, or 3-D response of a radiation sensor to one or more particle species. It supports idealized response functions for single-detector and dual-detector sensors in both rectangular and cylindrical geometries. It also supports look-up table responses produced by particle transport codes. The Response Function Library is originally a Matlab code, presently being converted to Python. The library is used for various manipulations of response functions. It includes transforms from pitch-angle/gyrophase coordinates to sensor angles. It provides calculation of numerical integration weights for spectral inversion, angular inversion, and 3-D data assimilation (measurement matrices). It also provides a bowtie analysis capability. The response file format standard and response function library code are community-maintained at https://github.com/PRBEM/IRBEM-extras/tree/main/rfl.