Ref. Ares(2018)6167603 - 01/12/2018

Deliverable 3.1:

Gain Estimation module

Authors Samuele Galeotta
Daniele Tavagnacco
Andrea Zacchei
Michele Maris
Gianmarco Maggio

Date Nov 30th, 2018

Work Package WP3 – Gain Estimation Module

DocId [INAF_BP03-01-1.0]

Revision History

Version	Authors	Date	Changes
1.0	Samuele Galeotta Daniele Tavagnacco Andrea Zacchei	Nov 30th, 2018	Prototype

Contents

1 Introduction

The main goal of the BeyondPlanck project is to build an end-to-end Gibbs sampler for the Planck LFI data, and use it to improve the overall calibration and fidelity of the final LFI sky maps.

Working Package 3, *Gain Estimation Module*, serves as the framework that provides conversion of time-ordered streams of voltages into time-ordered streams of thermodynamic temperatures with all the known instrument systematics removed. This working package will be the first step in an iterative pipeline as described in the Gibbs Sampling Integration (Working Package 1). The main objective of this WP is to compute the conversion factors and to provide to the subsequent WPs input data calibrated in thermodynamic temperatures, removing optical effects.

Working Package 3, *Gain Estimation Module*, is organized in two steps:

- Collection or creation of all the elements required for the thermodynamic calibration like beam model, Cosmological dipole signal and foreground model to be used for initialization and identification of the main systematics to be removed at timelines level.
- Run of a stand-alone software, *Data Calibration*, that from raw time ordered data (output of WP2) and using the previous described input will be able to calibrate in thermodynamic unit the Planck LFI timeline. The calibrated timelines are saved for further use in the mapmaking module.

2 Collection of data required

Input data necessary to the Gain estimation module can be divided in two different categories, *fixed input* are those that does not change between one and the subsequent iteration, and *variable input* are those subjected to the iteration itself.

2.1 Fixed input

Inputs related to physical effect that are kept fixed in the iteration loop.

- Calibration Mask: full sky mask with pixels to be excluded from the analysis because of the presence of the Galactic signal. It is the same used by WP2 and WP4.
- Velocity File: contains Planck satellite velocity registered during the mission. It used to compute the Orbital Dipole, the signal due to the satellite motion around the Sun.
- Convolution Parameters: set of parameters to compute the systematic signal coming from the optic of the Planck telescope.
- Subsampled housekeeping: set of subsampled housekeeping data containing temperature variations around the LFI horns.

2.2 Variable input

The main goal of BeyondPlanck is to build an iterative procedure to improve overall calibration. In order to achieve the result we plan to update the input sky model to the calibration procedure, this means that the following input will change in each iteration:

- Dipole parameters: amplitude and direction of the Solar Dipole signal, will come from Component Separation (WP6)
- Galaxy components ringsets: computed using *TotalConvolver* inside the Gibbs Sampler (WP1)

3 Gain Module

The *Data calibration* software uses the input defined in the previous paragraph and the Level2 data coming from *Data selection* (WP2). It apply an algorithm for the computation of calibration factors to convert the data from voltages to thermodynamic temperatures. An overview of *Data Calibration* is shown in figure 1.

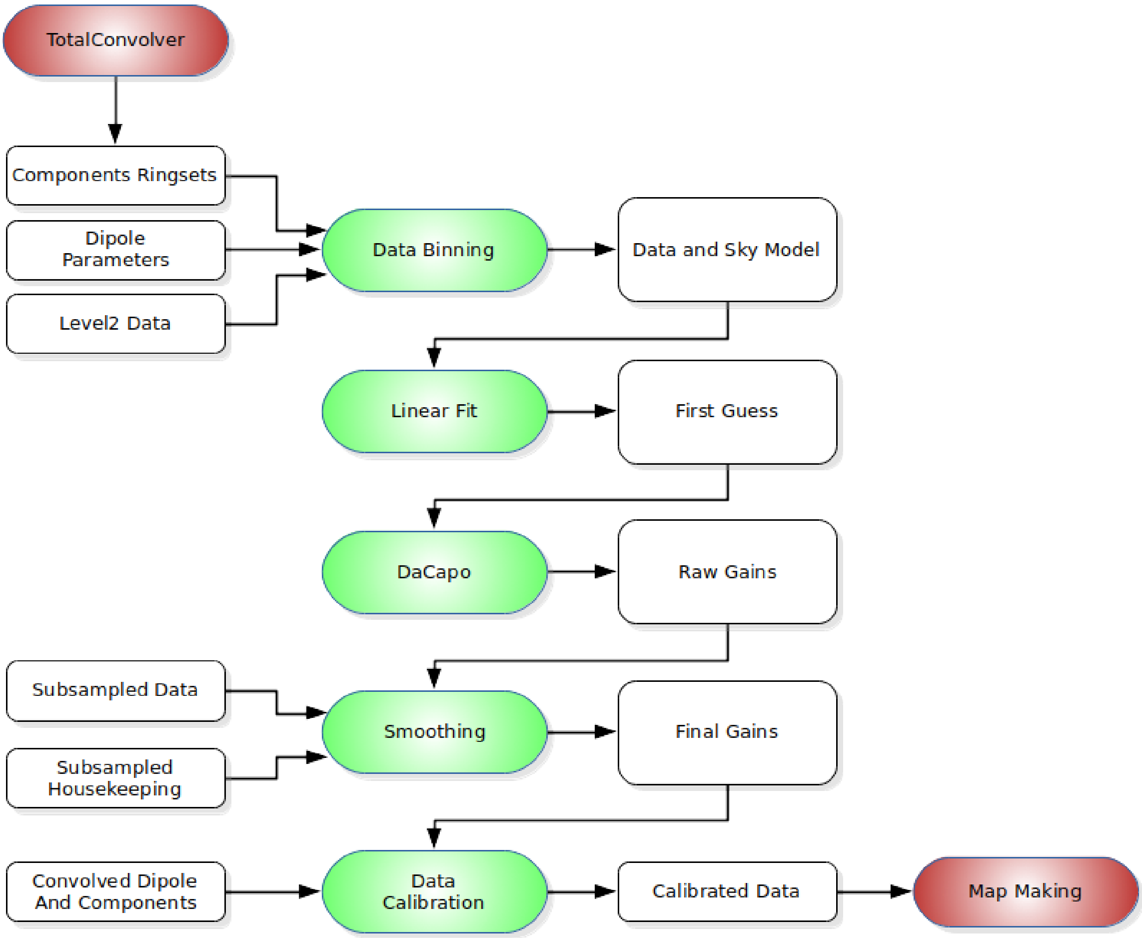

Figure 1 *Data calibration* module overview.

The *Data calibration* software is divided into five steps (green in figure 1), and is fully described in [1] and [3]:

1. *Data Binning*: this step creates timelines of the sky model convolving the Solar and Orbital Dipole using the Convolution parameters and summing the result to the Galactic components timelines extracted from the Galaxy components ringsets ([2]).

Then the resulting sky model and the Level2 data coming from WP2 to be used in the fitting procedure are subsampled to reduce their size. The procedure computes the mean of the data and the sky model coming from the same direction in the sky. this value is then used in the following, *linear fit, step.*

2. *Linear Fit*: perform a fit between the subsampled data and the subsampled sky model to compute a first guess of a conversion factor.
3. *DaCapo*: uses destriping techniques to iteratively improve the conversion factors. It produces an estimation of the Sky signal from the previous conversion factors, then it applies a destriping algorithm and use the results to apply a correction. The iteration continues until a convergence criterion is satisfied.
4. *Smoothing*: this step is a de-noising procedure to minimize the noise from the conversion factors stream. The result are the final conversion factors to be applied ([2]).
5. *Apply Conversion Factor*: apply the final conversion factors and remove the systematic signal coming from optic of the Planck telescope and the Orbital Dipole ([2]).

The output of *Data Calibration* software is a set of HDF5 files each one containing calibrated data.

4 Output data format

The file format is still in definition and tests are ongoing to define the optimal configuration. This section will then filled with the data format specification.

5 Conclusion

The *Data Calibration* software prototype is available in the BeyondPlanck GitLab repository and it can be executed with the following command:

```
mpirun -n N git_repository/calib/cmake.build/DataCalibration param.txt
```

where *N* is the number of processing nodes required and *param.txt* is the parameter file. A README file is available in the repository with the description of the keywords used in the parameter file.

The software is completely written in C++ language and uses most of the algorithms developed during the Planck project. The base algorithms have been extended by adding specific classes to handle FITS and HDF5 files to provide a portable and computing environment independent code.

6 References

[1] Planck 2018 results. II. Low Frequency Instrument data processing, arXiv:1807.06206

[2] Planck 2015 results. II. Low Frequency Instrument data processing, A&A 594, A2 (2016)

[3] Planck Explanatory Supplement, TOI processing LFI, https://wiki.cosmos.esa.int/planck-legacy-archive/index.php/TOI_processing_LFI