European Commission

Ref. Ares(2020)7201907 - 30/11/2020

THE FRAMEWORK PROGRAMME FOR RESEARCH AND INNOVATION
HORIZON 2020

Beyond PLANCK

Deliverable 4.3

Commander 4D map interface

Authors Anna-Stiina Suur-Uski

Date 25 November 2020

Work Package WP4 - Mapmaking

DocId [xxx-xxx-xxx]

HELSINGIN YLIOPISTO
HELSINGFORS UNIVERSITET
UNIVERSITY OF HELSINKI

planetek hellas

Revision History

Version	Authors	Date	Changes
1.0	Anna-Stiina Suur-Uski	25 October 2020	Initial Version

Contents

Overview

For Planck, the instrumental beam responses are significantly asymmetric, as described by Planck Collaboration IV (2016). Hence, standard Planck frequency maps represent the sky as convolved with an instrumental beam, resulting in a unique anisotropic smoothing pattern that depends directly on the beam geometry and scanning strategy of the instrument. Two of the most important and visually obvious effects of this are temperature-to-polarization leakage and deformed point sources. Both of these two effects may be corrected through explicit beam deconvolution, resulting in maps with well-behaved beam responses, albeit at the cost of more complicated noise properties. For BeyondPlanck beam deconvolution work we use the artDeco beam deconvolver (Keihänen & Reinecke 2012), which has already been applied successfully to Planck 2018 LFI data (Keihänen et al. 2019).

The artDeco algorithm is built on the assumption that noise in the input data is uncorrelated, in other words the input TOD is assumed to contain only astrophysical signals and white noise. We achieve the required removal of correlated noise by using the standard map-making, destriping, as a preprocessing step.

The artDeco reads in the destriped TOD and detector pointing in the 3D map format, which are derived from the 4D maps. While a usual two-dimensional HEALPix map is a function of two angles θ and φ (latitude and longitude), a 3D map contains also the third pointing angle ψ, which specifies the detector orientation. A 4D map includes the time information in addition to the three angles, in the form of a pointing period index (PID). In these TOD objects the signal values are coadded, not averaged, in any given TOD object element. We also keep count of the number of TOD samples that fall into each TOD object element. The flagged samples are discarded from all of these TOD objects, and only elements with non-zero hit counts are stored.

Commander3, the analysis pipeline developed in BeyondPlanck project (BeyondPlanck I), writes the destriped TOD into one HDF5 object per core. In this deliverable we have written an interface between Commander3 4D map data and 4D map FITS files as a python tool.

Data

For each Gibbs chain destriped TOD from Commander3 is available in a compressed data format, one HDF5 object per core. We further convert these into standard 4D map objects, one per horn per sample, using a python tool developed in this deliverable.

Tool

The tool, HDF5to4Dmap, is written in python 3.8 and is available in the artDeco directory in the BeyondPlanck Gitlab repository.

- https://gitlab.com/BeyondPlanck/repo/tree/master/artdeco/HDF5to4Dmap

The tool requires following python modules:

- H5py - https://www.h5py.org/ [Python interface to the HDF5 data format.]
- nympy - https://numpy.org/ [Used to work with arrays and mathematical functions.]
- astropy - https://www.astropy.org/ [Used to write data into file in the FITS format.]
- decimal - https://docs.python.org/3/library/decimal.html [Decimal floating point arithmetics.]
- glob - https://docs.python.org/3/library/glob.html [Unix style pathname pattern expansion.]
- multiprocessing - https://docs.python.org/3/library/multiprocessing.html [Process-based parallelism]
- subprocess - https://docs.python.org/3/library/subprocess.html [Subprocess management]
- collections - https://docs.python.org/3/library/collections.html [Container datatypes]
- time - https://docs.python.org/3/library/time.html [Time access]
- sys - https://docs.python.org/3/library/sys.html [System-specific parameters and functions]

The tool assumes following naming conventions:

- HDF5 file: tod_4D_chain0001_proc*.h5
- FITS file:: tod_4D_{FREQ}_{HORN}_chain0001_k{SAMPLE}.fits

Usage

To use HDF5to4Dmap tool:

- `module load python/3.8`
- `python HDF5to4Dmap.py 'dir_tod_4D' 'frequency' 'chain' 'sample'`
    - For example:
      `python HDF5to4Dmap.py '/mn/stornext/u3/hke/xsan/commander3/v2/chains_BP8_c51/' '30' '0001' '000001'`
- In the BeyondPlanck analysis we use the HFD5to4Dmap tool from a shell script to loop over all samples.

Validation

For a single 30 GHz sample the HDF5to4Dmap conversion takes roughly 16 minutes, and as an output we obtain one 4D map file per horn. We can further bin the 4D map into standard HEALPix map to compare with Commander3 map from the same data.

In Fig. 1 we show the difference between Commander3 and map binned from 4D map FITS data. We observe minor differences between these maps, and the differences are expected to arise from ...

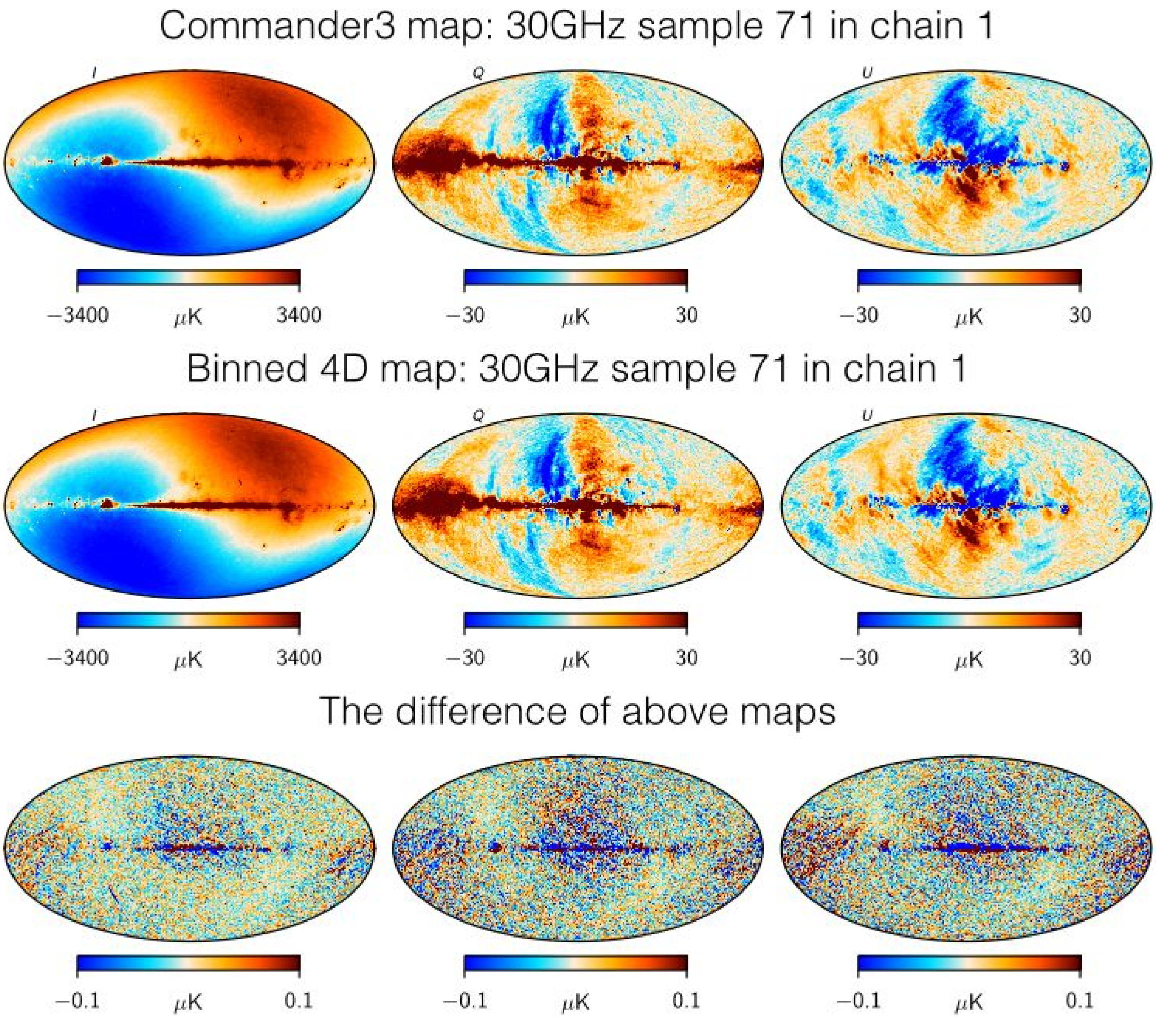

Figure 2: Top row: Commander3 map. Middle row: Binned 4D map.Bottom row: The difference between the Commander3 and binned 4D map.

References

Planck Collaboration: Planck 2015 results IV. Low Frequency Instrument beams and window functions, A&A 594, A4, 2016

E. Keihänen and M. Reinecke. ArtDeco: a beam-deconvolution code for absolute cosmic microwave background measurements, A&A 548, A110, 2012

E. Keihänen et al. Beam deconvolved Planck LFI maps, A&A 632, A1, 2019

BeyondPlanck I. Global Bayesian analysis of the Planck Low Frequency Instrument data, BeyondPlanck Collaboration, 2020, A&A, submitted, [2011.05609]