QUBIC Map-Making: Tutorial

Tom Laclavère

20 février 2025

Table des matières

1 Introduction **1**

2 General Organisation **1**

3 Frequency Map-Making **2**
3.1 Maps and Power Spectra 2
3.2 Fit tensor-on-scalar ratio 3

4 Component Map-Making **3**

1 Introduction

This document aims to provide a tutorial to explain how to perform data simulation, Map-Making, and tensor-on-scalar ratio estimation.
The explanations about the methods used, namely Frequency Map-Making (FMM) and Component Map-Making (CMM), can be found in the two following papers, respectively : **Spectral Imaging with QUBIC : building frequency maps from Time-Ordered-Data using Bolometric Interferometry** [1] and **Spectral Imaging with QUBIC : building astrophysical components from Time-Ordered-Data using Bolometric Interferometry** [2].
The details of the code will not be explained here, another document is planned to do that. This one aims only to explain how to run the code and change the simulation parameters.

2 General Organisation

The qubicsoft is divided into different repositories. The core of the code is located in **/lib** repository. It contains code to simulate each part of the instru-

1. https ://arxiv.org/abs/2409.18698
2. https ://arxiv.org/abs/2409.18714

ment, and also the code necessary to perform FMM or CMM.
To prevent the user from running these files himself, we build running files to simplify the process and to be used as a job launcher on computation centers. The running files are located in the following path : **/scripts/MapMaking/src/**, they are called **run_cmm.py**, **run_fmm.py** and **run_fit_fmm.py**. In addition to them, we decided to use parameter dictionaries to modify the simulation parameters, such that the user will never have to go into the code to run the simulations. These dictionaries are located in the repositories **FMM** and **CMM**. I will not describe here all the different parameters, as they all have a small description within the files. If one of them is not clear, do not hesitate to discuss that on RocketChat to improve it.

3 Frequency Map-Making

3.1 Maps and Power Spectra

To run the Frequency Map-Making, you just have to write the following line into your terminal : **python run_fmm.py /path_to_params_file**.
We suggest saving your parameters file as a .txt file, to keep the configurations that you often use, but you can also pass the path of the .yml file, the result will be the same.
Once you run the simulation, you will see a repository in **/FMM** named after the parameter **path_out**. This repository will contain two or three repositories :

- **maps :** will contain the pickle file where all the relevant pieces of information regarding the simulation are stored. ("maps_in" = input maps, "maps"=reconstructed maps, "maps_noise"=residual maps, "tod"=Time-Ordered-Data, "nus"=frequency of the reconstructed maps, "coverage"=coverage, "center"=center of the observation patch, "parameters"=parameters dictionary, "fwhm_in"=resolution of the input maps, "fwhm_out"=output resolutions (if we don't apply convolutions during reconstruction, array of zeros), "fwhm_rec"=resolution if the reconstructed maps (equal fwhm_out if convolutions during reconstruction, evaluate through analytic formula otherwise), "duration"=map-making duration, "qubic_dict"=QubicDict instance used during map-making)
- **png :** will contain a figure of the input, reconstructed, and residual maps
- **spectrum :** will contain the pickle file where all the relevant pieces of information regarding the cross-spectra computation are stored. ("nus"=frequency of the maps, "ell"=multipole values, "Dls"=Dl of the reconstructed maps, "Nl"=Dl of the residual maps, "parameters"=parameters file).

An important comment is that you can run the map-making without computing the spectra, but you can also compute the spectra from a file containing your maps. For this second option, you have to give an additional argument when running the command : **python run_fmm.py /path_to_params_file /path_to_maps_file**.

3.2 Fit tensor-on-scalar ratio

Once you computed your power spectra, you can fit the tensor-on-scalar ratio r on them. You have another parameters file dedicated for fitting in **/FMM/**, and another running file called **run_fit_fmm.py**. The full command you have to run is : **python run_fit_fmm.py /path_to_fitting_params_file /path_to_spectra_directory**.

Important note : you will need multiple spectra files within your spectra directory to perform the fitting.

4 Component Map-Making

The CMM works the same way as the FMM. The main difference is the parameters file which is way more complicated for the CMM (because the code is way more complicated also), and some of them are a bit hard to understand if you don't look at the code. This algorithm is still evolving, and then I hope that we will improve on this part.
Additionally, a file that is not downloadable on GitHub (because its size ¿ 100 Mo) is needed to run the CMM. You can download it following : `https://box.in2p3.fr/s/jMRniCBgi9g9o8K`, after that, you will need to move it to the repository : **/qubic/data**.
The command you have to run is : **python run_cmm.py /path_to_params_file**.
During CMM, we chose to let the possibility of making plots during the loops of the reconstruction. In **/CMM**, the repository **/jobs** will contain these plots.
The final result of the simulation will be stored in a pickle file, at the location given in the parameters.