DroPS - Deriving r from Power Spectra

Zhiqi Huang

September 30, 2025

1 Introduction

DroPS measures primordial gravitational wave (tensor-to-scalar ratio r) from cosmic microwave background (CMB). It is primarily designed for ground-based CMB experiments with low resolution, such as BICEP/Keck [1], AliCPT [2], Simons Observatory LAT [3], and CMB-S4 LAT [4].

2 Installation

The instruction here has been tested on Ubuntu-24.04.3LTS. A bit twists may need to be done if you are working with Windows or Mac-OS.

2.1 Installing Tools and Libraries

Install the following packages and libraries with Synaptic Package Manager (or "sudo apt install"):

- git
- gcc
- gfortran
- cmake
- python3-pip
- python-is-python3

- python3-venv
- openmpi-dev
- libxcb-cursor0
- libcfitsio-dev
- libgsl-dev
- libfftw3-dev
- libfftw3-mpi-dev
- libhealpix-dev

2.2 Set up a python virtual environment

Create a directory for python virtual environment in your work path (hereafter denoted as YourWorkPath)

```
mkdir YourWorkPath/.work
```

Create the python virtual environment

```
python -m venv YourWorkPath/.work
```

Activate the virtual environment

```
source YourWorkPath/.work/bin/activate
```

On windows you may need to run

```
YourWorkPath/.work/Scripts/activate.bat
```

in cmd.exe or

```
YourWorkPath/.work/Scripts/activate.psl
```

in PowerShell.

When you are done with your work, exit the terminal or use

```
deactivate
```

to exit.

If you are not working with other python projects. You may want to activate the virtual environment automatically with the terminal

```
echo “source YourWorkPath/.work/bin/activate”    ~/.bashrc
```

2.3 Install requirements

Activate the virtual environment either manually or automatically as described in the previous subsection.

Upgrade pip for the latest info of packages

```
pip install --upgrade pip
```

Now enter your work path where you want to install DroPS

```
cd YourWorkPath
```

Get the DroPS repository

```
git clone https://github.com/zqhuang/DroPS
```

Now enter the DroPS directory

```
cd DroPS
```

Install all dependences

```
pip install -r requirements.txt
```

2.4 Hack pysm3

Hacking a python package is probably against the basic idea of python, but we are doing it anyway to improve the efficiency of CMB simulations. If you only want to analyze maps, however, you can skip this "unpleasant" step.

Enter the DroPS directory

```
cd YourWorkPath/DroPS
```

Move the cmb.py file in the pysm3 package to somewhere else

```
mv PATH_TO_pysm3/models/cmb.py cmb_backup.py
```

and replace it with the cmb.py file that comes with DroPS

```
cp cmb.py PATH_TO_pysm3/models/
```

Here PATH_TO_pysm3 stands for the path where pysm3 was installed. On Ubuntu 24.04.3LTS, you may find PATH_TO_pysm3 is

YourWorkPath/.work/lib/python3.12/site-packages/pysm3

It may be slightly different in your case if you are not using Ubuntu24.04.3LTS. You can find out the path by doing

```
sudo apt install plocate
```

and

```
locate pysm3
```

3 Simulating sky maps

3.1 Generate a TOD filtering model

A critical step in processing data from ground-based CMB experiments is the filtering of contaminating ground and atmospheric signals from the time-ordered data (TOD). To simulate such a filtering process, you need information about the site of CMB experiment, which we do not actually have at the moment. Fortunately the gross effect of the filtering process is known. It suppresses the large-scale (low multipole) power of maps and makes the maps more non-Gaussian by mixing different Fourier modes.

If you are not keen about simulating precise filtering effect for a specific experiment, you may use the "mock filtering" tool that comes with DroPS to generate a filtering matrix:

```
python mock_filtering.py
```

Follow the prompt and enter the healpix resolution (nside, 128 for testing, 256/512 for serious simulations) and the file name for the filtering matrix (e.g. filter_128.pickle).

3.2 Simulate noises, CMB, and foreground

To begin with, you can simulate noise/cmb/foreground maps with a 4-channel ground-based experiment

```
python simulate.py Test/test_sim_config.txt
```

Read the configuration file Test/test_sim_config.txt to understand how the experiment is specified.

In this step, you generate a lot of noise, CMB simulation maps based on the noise model and cosmology that are specified in the configuration file.

You also generate a foreground map in this step, based on the model ['d0', 's0'] that is specified in the configuration file. We are not able to generate "a lot of foreground maps", as we do not really understand the details of the statistics of the Galactic emission. This ['d0', 's0'] foreground map only captures the gross feature of the Galactic emission. The "actual foreground" that we will analyze in the next section can be different from the simulated one.

To understand what ['d0', 's0'] means. Follow the pysm3 documentation at `https://pysm3.readthedocs.io/`.

4 Analysing sky maps

4.1 Analyzing one simulation

Generate the simulated sky with, e.g.,

```
python simulate.py Test/test_sim_config.txt maps/test_ 0.01 999
```

You can replace maps/test_ with your preferred prefix for the output maps, 0.01 with your preferred fiducial r, and 999 with your preferred random seed. To test whether DroPS can deal with a spatial variation of the foreground, you may also replace the ['d0', 's0'] foreground model with ['d1', 's1'] in the configuration file.

Now analyze the sky maps with

```
python mainpipe.py Test/test_ana_config.txt maps/test_
```

Read the configuration file Test/test_ana_config.txt to understand how to analyze the maps with different settings.

4.2 Analyzing many simulations

In this step we will simulate the sky with may different random seeds, and anlyze all the simulations. The purpose is to test whether the measured scalar-to-tensor ratio r is biased or not.

First you clean up the log file for r:

```
rm Test/r_logfile.txt
```

Now run the simulations with the bash script that comes with DroPS

```
./sim.sh
```

Analyze all the simulations with

```
./ana.sh
```

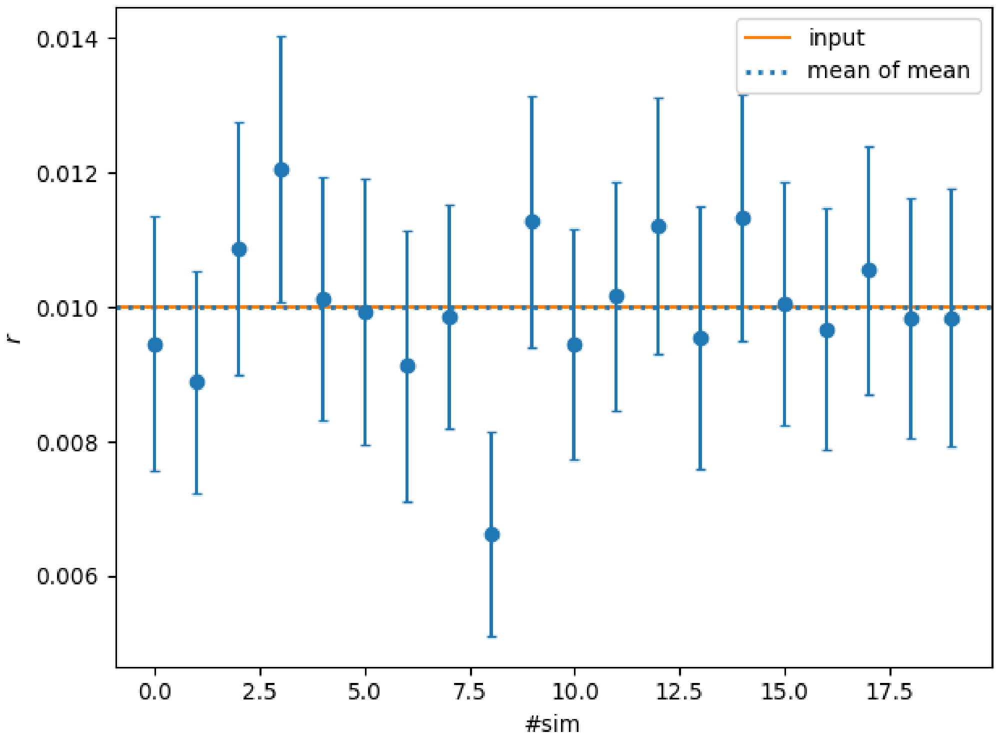

Figure 1: Reconstructed r for 20 skys with different random seeds. For each sky, the r value is reconstructed by comparing the sky with 300 simulations. The 20 skys are simulated with foreground model ['d1','s1'] (spatially varying SED of synchrotron and dust emission), while ['d0', 's0'] (fixed SED) is used in the 300 simulations.

Plot the result

```
python plot_rs.py Test/r_logfile.txt 0.01
```

Here 0.01 is the fiducial value of r used in simulations (see sim.sh). The "mean of mean" is supposed to be very close to the fiducial value, as shown in Figure 1.

References

[1] P. A. R. Ade, Z. Ahmed, M. Amiri, et al. BICEP/Keck XV: The BICEP3 Cosmic Microwave Background Polarimeter and the First Three-year Data Set. *Astrophysical Journal*, 927(1):77, March 2022.

[2] M. Salatino, J. Austermann, K. L. Thompson, P. A. R. Ade, et al. The design of the Ali CMB Polarization Telescope receiver. In Jonas Zmuidzinas and Jian-Rong Gao, editors, *Millimeter, Submillimeter, and Far-Infrared Detectors and Instrumentation for Astronomy X*, volume 11453 of *Society of Photo-Optical Instrumentation Engineers (SPIE) Conference Series*, page 114532A, December 2020.

[3] Tanay Bhandarkar, Saianeesh K. Haridas, Jeff Iuliano, et al. Simons Observatory: Characterization of the Large Aperture Telescope Receiver. *Astrophysical Journal Supplement Series*, 279(2):34, August 2025.

[4] Patricio A. Gallardo, Kathleen Harrington, Roberto Puddu, et al. Overview of the optical design of the CMB-S4 large aperture telescopes and camera optics. In Heather K. Marshall, Jason Spyromilio, and Tomonori Usuda, editors, *Ground-based and Airborne Telescopes X*, volume 13094 of *Society of Photo-Optical Instrumentation Engineers (SPIE) Conference Series*, page 130942F, August 2024.