UCB PHYS 229 Spring 2019 final project—21cmFAST and CLASS

Kolen Cheung

May 12, 2019

Contents

1 Introduction

21cmFAST is used to produce different ionization histories. CLASS is then used to produce their temperature and polarization angular power spectra and compare the results. The correctness of the result is however uncertain due to 21cmFAST being undocumented and strange behavior encountered from it, together with the fact that the glue to CLASS is less than perfect.

The code used in this report, including the generation and hosting of this document, is available on GitHub here. The data generated in this study is available on GitHub Releases here.

The canonical viewing format of this document is in HTML because of the use of interactive plots. PDF version is also provided, albeit the plot won't work there. Links to HTML version of Jupyter notebooks used in the study are included in this report for further reference.

1

1.1 Understanding CLASS

To ensure CLASS is setup correctly, `COM_PowerSpect_CMB-base-plikHM_TTTEEE-lowl-lowE-lensing-minimu`
available from Planck Legacy Archive (Collaboration et al. 2018) is compared.
In order to reproduce this result, these files available from CLASS on GitHub
(Blas et al. 2011) are used:

- `base_2018_plikHM_TTTEEE_lowl_lowE_lensing.ini`
- `cl_ref.pre`

together with setting `format = camb`.

The Python interface `classy` is also used to compare `class` results. Without the
`format = camb` option, they provides identical results. However, with `format = camb`, `classy`'s output is not changed, which may be due to a bug in the classy
module when parsing the setting. Matching results from with and without
`format = camb` setting, by manually rescaling C_l to D_l is not successful due to
unknown reason.

Therefore, in this study `class` is used directly with parameters modified from
`base_2018_plikHM_TTTEEE_lowl_lowE_lensing.ini` with `format = camb`.
`cl_ref.pre` is not used to sacrifice precision for manageable compute time.

See more from these notebooks:

- `class_check.ipynb`: used to understand CLASS and reproduce
  `COM_PowerSpect_CMB-base-plikHM_TTTEEE-lowl-lowE-lensing-minimum-theory_R3.01.txt`

- `class_vs_classy.ipynb`: comparing class and classy to make sure they
  provides identical results.

1.2 Understanding 21cmFAST

21cmFAST is hard to understand, due to the lack of documentations but
comments lying within the code base. After running programs listed under
`Programs/PROGRAM_LIST` from 21cmFAST on GitHub (Park et al. 2018), where
one program described there is not even available, `drive_logZscroll_Ts`
should be the one to use. For our purpose, since we are interested in
the ionization histories, x_e should be extracted from a file named similar to
`Output_files/Ts_outs/global_evolution_Nsteps40_zprimestepfactor1.020_L_X3.2e+40_alphaX1.0_f_`

The default z range in the code is from 6–30. `drive_logZscroll_Ts.c` could
be modified to change this, but when the z-range is changed to 1–30 instead,
the output does not make sense. For example, x_e would continue to rise as z
is decreasing below 6, but then drops again. Also, a bunch of other parameters
eventually becomes `nan` as z gets lower than 6. For these reasons the z range is
kept at there default 6–30.

Lastly, to match the cosmological parameters between those used in 21cmFAST and CLASS, 21cmFAST's header files is modified in a fork of 21cmFAST in this commit to match those defined in `base_2018_plikHM_TTTEEE_lowl_lowE_lensing.ini`.

See more from these notebooks:

- `21cmfast_understand_output.ipynb`: used to understand the output from 21cmFAST

1.3 21cmFAST to CLASS

In CLASS, `explanatory.ini` and `doc/manual/CLASS_MANUAL.pdf` documents everything. To inject x_e from that obtained in 21cmFAST, `reio_parametrization` should be set to either `reio_bins_tanh` or `reio_inter`. `reio_bins_tanh` is to interpolate using tanh, while `reio_inter` linearly.

`reio_bins_tanh` should in principle be better, however with 76 x_e values from 21cmFAST's output, it needs too much memory than available, and after binning to only a few bins, the interpolated x_e in CLASS is too step-wise to reproduce the input x_e.

`reio_inter` is not without its problem. From `explanatory.ini`, a few z, x_e values are needed to be specified at the boundary of this region. At the left boundary, e.g. if one set $z = 6$ to $x_e = -1$, it would interpolate x_e to 1 linearly between z steps.

It seems a better approach would be to interpolate linearly within $z = 6$–30, and interpolate with tanh on the left boundary. But the code provide no such option. In the end, `reio_inter` is used to interpolate linearly, and setting $x_e = -2, -1, 0$ at $z = 0, 6, 30$ according to the documentation, where the $z = 6$ location is chosen somewhat arbitrarily to coincide with the left boundary of the input such that it would be equivalent to ionized almost instantaneously at $z = 6$.

Although the resulting spectra from this input does not agree with `COM_PowerSpect_CMB-base-plikHM_TTTEEE-lowl-lowE-lensing-minimum-theory_R3.01.txt` mentioned in Understanding CLASS, the hope is that the differences between x_e from $z = 6$–30 can be represented in the differences in the final spectra output from CLASS, although we have no ways to verify that in this report.

See more from these notebooks:

- `class_custom_x_e.ipynb`: preliminary check on the CLASS x_e using output from 21cmFAST

- `classy_trials.ipynb`: check x_e from classy

2 Parameters

Parameters from 21cmFAST is left as default except for those cosmological parameters update mentioned above. CLASS' parameters from `base_2018_plikHM_TTTEEE_lowl_lowE_lensing.ini` is used, with the injection of x_e from 21cmFAST output. In addition, the first 7 parameters[1] from Table 1 (Park et al. 2018, p. 5) are varied, defined as these constants in `ANAL_PARAMS.H` and `HEAT_PARAMS.H`:

- `STELLAR_BARYON_FRAC`
- `STELLAR_BARYON_PL`
- `ESC_FRAC`
- `ESC_PL`
- `M_TURNOVER`
- `t_STAR`
- `L_X`

All these parameters are varied once at each boundary of "Allowed range" defined in Table 1 (Park et al. 2018, p. 5) to get a sense of the possible differences in these ranges. Hence there's 15 combinations—1 for default; 2 each per parameters, 1 for each boundary of the allowed range.

3 Logistics

In order to reproduce the result of this code, these repositories at respective tag/commit should be used:

- 21cmFAST at commit e81caff (Park et al. 2018)
- CLASS at v2.7.2 (Blas et al. 2011)

3.1 Workflow

Note that the parameters in `21cmFAST` are compile time constants, therefore a script to generate header files is used and a recursive makefile is used to compile all the programs first. In addition, modified `NUMCORES` and `RAM` in `INIT_PARAMS.H` to match that of the machine used.

1. within the `21cmFAST` directory, created a sub-directory `Parameter_spaces`, within which

    - notebook/21cmfast_param_gen.ipynb: generates 21cmFAST parameters by writing header files per parameter set,

    - bin/symlink.sh: symlink the rest of code needed to be compiled,

---

[1]a corresponding constant defining the last parameter cannot be found in the header files so it is dropped from this study.

- bin/makefile: `make compile -j` to compile `drive_logZscroll_Ts` per parameter set.

2. bin/makefile: `make 21cmfast` to run all programs[2].

3. `21cmfast_get_many_params.ipynb`: parses 21cmFAST data and write `.ini` files for class

4. bin/makefile: `make class` to generate final spectra[3].

5. `class_collect_param.ipynb`: collect spectra from CLASS

4 results

Note that in 2 set of parameters, 1 of them got non-sensical result from 21cm-FAST, and another one request north of 192 GiB memory in class. Therefore the results below contains only 13 sets of parameters.

x_e as the parameters varies:

x_e

Below are the plots per spectrum, with slider across all parameters above. Except for the one labeled, the other parameters are left at the defaults.

TT

EE

BB

TE

dd

dT

dE

From the plots, these spectra, except perhaps dT, are not sensitive to these changes.

It is inconclusive however if the spectra actually is not sensitive to the changes of the parameters, or if the included z-range of 6–30 is not enough to capture the differences, due to the technical problems encountered mentioned above in 21cmFAST to CLASS.

---

[2]sequentially since it has ~20 GiB of scratch and ~12 GiB of RAM needed.
[3]sequentially since class is multithreaded, also it uses a ton of RAM, north of 128 GiB.

References

Blas D, Lesgourgues J, Tram T (2011) The Cosmic Linear Anisotropy Solving System (CLASS) II: Approximation schemes. arXivorg 034–034

Collaboration P, Aghanim N, Akrami Y et al (2018) Planck 2018 results. VI. Cosmological parameters. arXivorg arXiv:1807.06209

Park J, Mesinger A, Greig B, Gillet N (2018) Inferring the astrophysics of reionization and cosmic dawn from galaxy luminosity functions and the 21-cm signal. arXivorg 933–949