Prepared for submission to JCAP

Manual for NuGrid-Gadget (0.9)

Zhichao Zeng,[a,b,1]

[a]Department of Physics, the Chinese University of Hong Kong, Sha Tin, NT, Hong Kong
[b]Department of Physics, the Ohio State University, Columbus, Ohio 43210, USA

E-mail: zeng.408@osu.edu

Abstract. It is a code for grid-based N-body cosmological simulation with massive (and degenerate) neutrinos involved. Upto version 0.9, the simulation can be setup separately by preparing CAMB, 2LPTic and Gadget2, but not automatically connected. I will try to make it smarter in the next update.

[1]Corresponding author.

Contents

1 CAMB

`CAMB` is a common cosmo code for generating initial power spectrum (for different matter species). By default it is able to include massive neutrinos and Terry Yeung extended it to allow for degenerate neutrinos with the asymmetry parameter ξ. I modified the Jan-2017 version of the original `CAMB` so that it follows Terry's work.

How to use: 1. cd to the example directory `cmbonly`

2. ./camb m002-xi3-025.ini (m002 means m_ν the averaged neutrino mass is 0.02 eV, ξ_3 for the 3rd mass eigenstate is 0.25, -025-low.ini means it includes the -1σ of the refitted cosmological parameters)

What you need to change in a .ini file if needed (from top to bottom):

xi_nu3 (input variable), output_root (directory for output and prefix), Ω_b, Ω_c, Ω_Λ and Ω_ν

(**please note** that here Ω_ν is $\Omega_\nu(m_\nu, \xi = 0)$; the logic here is that we input the Ω_ν for massive only, and then add back the ξ_3 so that it becomes degenerate as well; you can check the returned value of m_ν in the output screen of `CAMB`);

nu_mass_eigenstates, massive_neutrinos and nu_mass_fractions are related to hierarchy problem and you may change it if needed;

scalar_amp(1) and scalar_spectral_index(1) are A_s and n_s, also refitted from `CosmoMC`;

transfer_kmax and transfer_k_per_logint sets the output spacing of k and P_k; transfer_redshift sets the redshift you want to output the P_k and transfer_matterpower is the name of the output file (added by the prefix mentioned above);

transfer_power_var specifies the species of which the power spectrum is output: 6 for neutrinos only, 7 for all matter and 8 for CDM+b;

Note1: above is for massive neutrino cosmology, if you want the CDM fiducial model, please DO NOT use this code but the original `CAMB` instead, as I've also included in this package. The result is incorrect if you input $\Omega_\nu = 0$ but use this modified version of `CAMB`.

Note2: after the successful output, open the .txt file (by default in the 2LPTic folder) and you will find the first line is started by #. Make sure you remove this line, otherwise error would occur in the next step (Shihong has improved this but I haven't tested yet so maybe we can implement it in the next update..)

Note3: for now the default output directory of pk.txt is set to be in the folder of 2LPTic, but maybe changed later

'./camb xxx.ini' to run CAMB

2 2LPTic

`2LPTic` is an initial condition generator for a cosmological Gadget2 simulation. The basic idea is that, given an initial Pk, what the perturbed initial configuration could be, and apply it to the particle position. This requires the perturbation in terms of particle instead of field, where we need the Lagrangian theory now (see Chap 1.2.1 of my thesis).

/de2_deduction/2LPTic is for one-kind particle simulation and /2-compo-nbody/2LPTic is for simulations with two kinds of particles.

- **set the particle number** : Better all of **Nmesh**, **Nsample** and **GlassTilefac** (most direct to particle number) are changed together.

- **Box** : in unit of kpc/h.

- **GlassFile** : This is the unperturbed structure, which in principle should be homogenous and isotropic. I have enclosed a glassfile **64-glass** made by Shihong Liao, which should be a pretty good one.

- **Cosmo params** : **OmegaBayon** $= \Omega_b$ in CAMB; **Omega** $= \Omega_b + \Omega_c$ in CAMB; **OmegaLambda** $= \Omega_\Lambda + \Omega_\nu$ in CAMB (so we turn on the 'deductfromDE' in later part of the .param); $H_0 = H_0$ in CAMB.

- **WhichSpectrum** : when it is 2, it simply uses the designated input pk in **FileWithInputSpectrum**. No need to worry about the modifed H or n_s.

- **ReNormalizeInputSpectrum** : set it to 0 because we don't need to normalize it.

- **Seed** : change this to any value means another realization of a given cosmology. We actually change it quite often.

- **mass_1 and xi_3** : set it according to your input.

- **Others** just leave as they are; for some I'll hide them inside the code so the user does not need to know about it.

For a two-kind-particle simulation:

- **mass_nu_expan** when it is 0, no neutrino thermal velocity is included for type 2, so it's CDM+CDM; when it is 1, the thermal speed is turned on and it's ν + CDM.

- **mass_nu_frstr** : controls the mass ratio of these two particles.

Before running: always open ratio.py in the 2LPTic directory, modify the name of the CDM and neutrino power spectrum files, and calculated the ratio accordingly. The generated ratioxxxxx.txt should be together with ./Gadget2.

Makefile of 2LPTic should be accordingly modified–mainly to customize the position of libraries.

'make'
'./2LPTic xxx.param' to run the IC generator

3 Gadget2

Similar to 2LPTic, /de2_deduction/Gadget2 is for one-kind particle simulation and /2-compo-nbody/Gadget2 is for simulations with two kinds of particles.

Following are the options in a .param file:

Directory and filenames : the first three are input while the 'nu_pk_txt' is the output of neutrino power spectrum (in form of ratio to CDM). 'InitConFIle' is just the ics output of 2LPTic, please set the file name accordingly. 'OutputDir' is the directory of output snapshots. 'ratio_nu_cdm' is the output of ratio.py, which is the initial power spectrum ratio of neutrino and cdm, we use it as an input at the first PM calculation. My suggestion is that put all of these in the same directory. Note: if the simulation is successful, you will see the same number of files nu_pk_i.txt as the number of snapshots. This is because we output the neutrino every time we output a snapshot.

TimeLimitCPU : The max time to run this simulation, so if you need a long run, make sure this is large enough. Otherwise you may need to make use of the restart functions.

OutputListFileName : when you set 'OutputListOn', in this file the a at which you want outputs are stored. Example is included.

Ω : same as those of 2LPTic, make sure $\Omega_0 + \Omega_\Lambda = 1$, and $\Omega_\Lambda = \Omega_\Lambda + \Omega_\nu$.

Boxsize : same as ics.

TreeAllocFactor : you need to increase this (to 2 in my case) when you are doing a particle-based neutrino simulation. This is related to how the tree algorithm is further distributed.

Softening : in my case I set the softening length to be 1/20 of the average distance of particles. Only the Halo softening is relevant in a grid-based simulation, while only Halo and Disk relevant in a particle-based simulation (that is to say, the additional neutrino particle is stored in the 'disk' category). For now I just scale the softening length according to the mass of different particles, but this may need further investigation.

neutrino_scheme : 4.0 when you need a grid-based correction in the PM calculation. Anything else (better 0) when you don't need it. This is the main option to turn on/off this neutrinio patch.

xi_3 and mass_1 : input neutrino parameters of your own choice.

expan_on : keep it 1 if you need to include the modified Hubble expansion rate.

mass_hierarchy : 2 for same mass (no hierarchy, which is not true but doesn't really matter). If you want to try on inverted/normal hierarchy, use 1 or 3.

lepton_asymmetry : 1 if you use the constraining equation of xi_i, 0 if you just keep them the same value.

'make'
'mpirun -np 2^n ./Gadget2 xxx.param ' to run Gadget2

4 Readsnap

This is a simply code for calculating power spectrum (or you can modify it to do whatever you want, as the first part is to transform the binary information in snapshot shots to human-readable data.)

Similarly, there're two sets, nonu-readsnap for CDM only simulation and particle-based neutrinos (i.e. you don't need the extra input of neutrino power spectrum). And readsnap is for grid-based neutrino simulations.

I have set different z-read.param for the same number of output snapshots, please modify the corresponding output number in the .param. It will also look for the ratio of pk txt automatically.

'make'
'./Readxxx aaa.param' to run the pk reader

5 Final notes

I will suggest the users to write a script to connect the CAMB, 2lpt and Gadget2 in a sequence. I would make this an internal feature in the next update of this set of code.

A sample script is also attached without much editions...please find what you might need inside

You may also find some options in .param files that are not mentioned in this manual, part of which are from the original code and are not very relevant, while others are just wastes and I'll remove them next time.

References