Computational Astrophysics (PHYS11037)

Cosmological *N*-body simulation I

A. Introduction

The purpose of this exercise is to compute the evolution of cosmological structures using a Particle-Mesh *N*-body code and analyze the results.

For the exercise, you will:

1. Set up and run an *N*-body code for a Cold Dark Matter cosmological model
2. Plot images of the density field of the dark matter particles
3. Explore the distribution of the dark matter density fluctuations
4. Explore the evolution of dark matter density fluctuations

The assignment write up should include:

1. A Methods section listing the required commands to go from an *N*-body run to getting the dark matter density distribution, producing plots and answering the questions.
2. A Results section of your principal results, with answers to the questions below along with figures and possibly tables.

B. Running and analysing the simulations

The code used is Enzo (including an initial conditions routine inits), a standard tool for cosmological simulations. (See http://enzo-project.org.) The code is actually a combined *N*-body and hydrodynamics grid code, but for this project only its *N*-body functionality will be used.

The python-based analysis package yt that will be used to analyse the output from Enzo. (See http://yt-project.org.) Any python commands required will be provided. It is also possible to run yt through a jupyter-notebook interface, which many students prefer. Unfortunately, when working remotely the jupyter-notebook interface is too slow to be practical. If working on a CPLab system remotely, direct python commands (as described) below will be required. Alternatively, the required data from an Enzo simulation may be downloaded onto your own device if working remotely, however the files are large and may take a while.

It is advised that all students be prepared to use python commands directly on the CPLab systems, as this will be necessary for the Assessment in this module.

The packages Enzo, inits and yt are installed on the CPLab systems. You should have an account on the CPLab systems for the assignment. To log in remotely, do:

> ssh UUN@student.ph.ed.ac.uk, where 'UUN' is your Universal User Name. This logs you into the CPLab gateway. Once logged into the gateway, you need to access one of the workstations through ssh:
> ssh-to-cplab.
Do not run your computations on the gateway system.

0a. Before you do anything, you need to log into your account and create a working directory for your runs. To accommodate the large data files that will be produced, an ifa directory has been created on the CPLab system: /storage/teaching/2021-22/compastro.

Move into the directory:
> cd /storage/teaching/2021-22/compastro

A subdirectory using your student number (of the form snnnnnnn) should already exist. If not, please let the TA or the Course Organiser know and we'll arrange it.

Move into your subdirectory for all your work for the PM assignments and the assessment:
>cd snnnnnnn

Make a subdirectory to hold your PM simulations, for example:
> mkdir PM
then move into it:
> cd PM

0b. You'll need to answer some questions in this exercise. Open up a document to make a report with your answers. (You may use Libre Office: > loffice , or some other resource if you prefer.) **Please save the report as a PDF file.** You may upload the completed report to Learn to receive feedback.

If using the jupyter notebook, you may convert to PDF using the brower: on the far right of the browser menu, click on the symbol with 3 parallel horizontal bars to open a menu. Use the 'Print' option to save to a PDF file.

Be sure to save your results frequently in case the notebook fails for any reason.

Now you're ready to start a simulation.

1. First, you'll need to set up initial conditions for a cosmological *N*-body run. These are the original particle positions and velocities for Enzo to evolve using its *N*-body integrator.

You will first use a tool called inits, which is executed on a parameter file

defining the problem. Then Enzo will use the output from inits along with a second parameter file (where the name of the output files from inits are some of the parameters) to perform the run. The possible parameters are listed at: http://enzo.readthedocs.org/en/latest/user_guide/CosmologicalInitialConditions.html

As you can see, the list is rather long. The Enzo team, however, has made it easier by providing some sample parameter files for inits and Enzo on their website. The inits parameter file for a cosmological *N*-body run may be downloaded from the Learn page for the course in the folder */Course Materials/III. Particle-Mesh method/Practice workshop assignments/Supplementary PM files/Enzo scripts*. They are also available in the CPLab directory /storage/teaching/2021-22/compastro/SupplementaryFiles/enzo_scripts.

You will need the file: dmonly.inits
The matching Enzo parameter file is: dmonly_unigrid.enzo
You need to download both of these into the work directory (/storage/teaching/2021-22/compastro/snnnnnnn/PM) for the *N*-body results.

2. Run inits to generate the initial conditions:

> inits.exe -d dmonly.inits

3. You'll need to edit the Enzo parameter file dmonly_unigrid.enzo to tell it where your work directory is. Using your preferred editor, look for the variable "GlobalDir". By default it was set to "/Users/bwoshea/Desktop/testrun". Change this to your work directoy (/storage/teaching/2021-22/compastro/snnnnnnn/PM) (and save the file).

4. The Enzo parameter file also instructs Enzo to output a large number of data dumps that aren't required for this project. (You'll only need the redshift dumps.) Since the data dumps take up considerable diskspace, we don't want Enzo to make them. In the dmonly_unigrid.enzo file, delete the lines:

DataDumpName = DD
DataDumpDir = DD
dtDataDump = 2.0

(and save the file).

5. Now you can run the enzo executable file, which performs the *N*-body simulation using the parameter file dmonly_unigrid.enzo.

To run the enzo executable on the file, do:

> enzo.exe -d dmonly_unigrid.enzo > dmonly_unigrid.log &

The '&' puts the job into the background. The screen output is saved to dmonly_unigrid.log.

Congratulations. You are now performing an *N*-body simulation, and generating piles of data files. Particular redshift dumps are in the RDnnnn files. You can see which redshifts they correspond to from the amr.out file: for example, CosmologyOutputRedshift[5] = 1 means RD0005 corresponds to z = 1, or an expansion factor for the universe of a = 1/ (1 + z) = 0.5, ie, when the universe was half its current size.

The full run will take a while (about 20 minutes).

6. While the code is running, open a new terminal window and have a look at dmonly_unigrid.enzo. Answer the following:

Q1

i. What are the cosmological parameters used?
ii. How big is the simulation box on a side?
iii. How many grid zones are used?
iv. What is the initial redshift used? (This sets the initial time of the simulation.)

7. In addition to producing the initial conditions, the inits code also produced two files PowerSpectrum_z=60.out and PowerSpectrum_z=0.out. These are the theoretical power spectra of the dark matter density fluctuations at z = 60 and z = 0, respectively. Four columns are provided. The first is the comoving wavenumber k (in h/ Mpc). The following three columns are identical. This is because only one form of matter is present in the simulation, the dark matter. (The other columns are just copies.) Answer the following:

Q2

i. What range in wavenumbers should the simulation volume cover?
ii. Explain why the values of the power spectrum are much larger at z = 0 compared with z = 60. What is the approximate ratio?

8. To analyse the simulation results you will use yt. You might like to see what the simulation looks like. The yt online manual is at: http://yt-project.org/docs/3.0/ It is based on python scripts. A script not in the yt library that will be useful computes mass densities as overdensities, the ratio of the density to the mean density. A dark matter overdensity callback has been created for this.

From the Learn page for the course, download the file *Course Materials/III. Particle-*

Mesh method/Practice workshop assignments/Supplementary PM files/yt scripts/dm_overdensity.py
into your working directory. The file is also available in the CPLab directory /storage/teaching/2021-22/compastro/SupplementaryFiles/yt_python_scripts.

Since yt runs from python, you'll first have to start up python. You may do this either directly, or if working on a CPLab desktop, via a jupyter-notebook. The jupyter-notebook has the advantage of readily visualizing plots, and the notebook may be saved for future work.

To start-up Python directly, do:
> python

If you're in python, it should return the prompt:
>>>

To use the jupyter-notebook interface instead, do:
> jupyter-notebook

and activate Python 3 (click on the 'New' tab for a dropdown menu).

Next import the yt modules:
>>> import yt
[do 'shift-return' to exectute each line if using the jupyter-notebook]

and the units definition module:
>>> from yt.units import *

Import the “overdensity” field (be sure to include the underscores):
>>> from dm_overdensity import _overdensity

Now load the data file:
>>> ds0 = yt.load("RD0000/RD0000")

It should return some information about the file to screen.

Add the “dm_overdensity” field to the data set:
>>> ds0.add_field(("gas",'dm_overdensity'),function=_overdensity)

Q3:
i. How old is the universe for this data dump in Gyr (units of 10^9 yrs)? Explain what the internal code unit of time used is. [Hint: Look at http://enzo.readthedocs.io/en/latest/reference/EnzoInternalUnits.html and

http://yt-project.org/docs/3.0/analyzing/units/comoving_units_and_code_units.html]

ii. Why aren't cgs units used?

9. Now we'll make images of the dark matter density field.

In the Table of Contents, click on "Visualizing Data/How to Make Plots".
An example is given under "Slice Plots" for visualizing a slice through
the z-axis at z = 0.5 (in units of the box side), centred at (x,y,z)=(0.5,0.5,0.5),
that's 20 kpc on a side. Let's try it on "RD0000," which are the initial conditions.

The dark matter density field is defined from the dark matter particle positions through CIC smoothing. The variable to be used will be referred to as 'all_cic', or sometimes as '('deposit', 'all_cic')'. A listing of defined data fields is at http://yt-project.org/docs/3.0/reference/field_list.html#enzo-specific-fields . You can list them all:
>>> ds0.derived_field_list

Make a slice of the dark matter density field:
>>> slc0 = yt.SlicePlot(ds0, 'z', 'all_cic', center=[0.5,0.5,0.5],width= (20,'kpc'))

If you're running jupyter-notebook, you may visualize it using:
>>> slc0.show()

Whether using jupyter-notebook or python directly, you may save the file in png format:
>>> slc0.save('densitysliceplot0.png')

[Outside python, you can convert it to a PostScript file using the tools:
pngtopnm and pnmtops, eg, pngtopnm file.png > file.pnm and
pnmtops file.pnm > file.ps. You may also visualize it on your workstation using
> display densitysliceplot0.png &]

It won't look like much because the region was too small. The cell size is in units of proper kpc. Have a look at the original Enzo parameter file to see how wide a single grid zone should be:

Q4: How wide is the grid zone? Work out how wide it should be. [Hint: the box size in dmonly_unigrid.enzo expressed distances as comoving in units of Mpc/h, where h = H0/ (100 km/s/Mpc) and H0 is the Hubble constant. But the plot expresses distances in proper kpc at the redshift plotted.]

Remake the plot for the full simulation volume. (Read the description for SlicePlot on the webpage "Visualizing Data/How to Make Plots" if unclear how to do this.)
[Detailed information on SlicePlot is at: SlicePlot api]

10. Make a projection plot of the dark matter density field:
>>> prj0 = yt.ProjectionPlot(ds0, 'z', 'all_cic', center=[0.5,0.5,0.5],width= (20,'kpc'))

You can save the file in png format:
>>> prj0.save('densityprojectionplot0.png')

Q5:
i. Qualitatively compare the image with the corresponding slice plot.
ii. Explain the difference in units on the density. What range would you expect based on the colour bar for the density in the slice plot? Note: 1 kpc ≈ 3.0856e21 cm.

11. Shift the sliceplot for the full box along the x direction, eg, centre it at x = 0.6

Q6:
i. Compare the image with the previous, centred at x=0.5 (Note that a region of 'ghost cells' may appear, which may be disregarded.)
ii. Explain the similarities and differences in terms of the PM algorithm.

12. By now the simulation should have completed. Make sliceplots of the dark matter *overdensity* at z = 4, z = 1 and z = 0 for the full width of the simulation volume similar to the one for z = 60. To make a plot of the dark matter overdensity, replace the field 'all_cic' by 'dm_overdensity' in the yt.SlicePlot command argument. It will be helpful to set a common colour bar. For example, to fix the colour bar for an overdensity range 0.0001 to 200 for a slice plot of the overdensity named slc1, do:

>>>yt.AxisAlignedSlicePlot.set_zlim(slc1,field='dm_overdensity',zmin=0.0001,zmax=200)

before saving the plot.

Q7: Qualitatively compare the images of the overdensity at z = 60, z =4, z= 1 and z = 0. Explain their similarities and differences.

13. Now make a slice at z = 1 that's 8 x 8 cells.
Some cells look empty (white).

14. Plot the particles in the slice using, for appropriate values of W and D:

>>>pp1=yt.ParticleProjectionPlot(ds1,'z',center=[0.5,0.5,0.5],width=(W,'Mpc'),depth=(D,'kpc'))

Q8:
i. What values of W and D correspond to the 8 x 8 cell slice plot? Adjust D to include two layers of cell and remake the particle plot.

ii. Explain why the empty cells show zero density. Why do some cells show a non-zero density when there appear to be no particles in them?
15. To finish up a session, quit python:

```
>>> quit()
```

16. Upload your report to Learn, PM Assignment 1, to receive feedback. Be sure also to keep your results since you may find them helpful for the assessment.

C. References

Some useful online manual pages:

Enzo:

General Enzo documention:
http://enzo.readthedocs.io/

How to run a cosmology simulation:
Enzo cosmology simulation instructions

Additional scripts for a variety of problems are at:
Sample Enzo test problem scripts

yt: http://yt-project.org/doc/
To plot the density field:
http://yt-project.org/doc/visualizing/plots.html
http://yt-project.org/doc/cookbook/simple_plots.html

python basics: http://www.afterhoursprogramming.com/tutorial/Python/Introduction/

linux/unix basics: https://www.pcsuggest.com/basic-linux-commands/

Some useful literature:

1. Davis M., Efstathiou G., Frenk C. S., White, S. D. M. (1985) ApJ 292:371
“The evolution of large-scale structure in a universe dominated by cold dark matter”
(http://articles.adsabs.harvard.edu//full/1985ApJ...292..371D)

2. Efstathiou G., Davis M., White S.D.M., Frenk C.S. (1985) ApJS 57:241
"Numerical techniques for large cosmological N-body simulations"
(http://adsabs.harvard.edu/abs/1985ApJS...57..241E)