ASTR 660 Homework 1

Deadline: start of class on 10/17

There are two questions. Each question is worth 50 marks.

Please upload answers to your GitHub repository as `homework1.pdf`**.**

You can use whatever tool you want to produce the answers, but they must be neat and readable. Equations, units etc. must be properly formatted and preferably look like LaTeX output, not MS equation editor. Some examples:

Good:

The Hubble constant is $H_0 = 70\,\mathrm{km\,s^{-1}\,Mpc^{-1}}$. The mass of the Sun is $M = 1\,\mathrm{M_\odot}$. The following is true:

$$\frac{1}{\sin^2\theta + \cos^2\theta} = 1$$

Bad:

The Hubble constant is H0 = 70km/s/Mpc. The mass of the Sun is M=1.0M_sun. The following is true:

$$\frac{1}{\sin\square^2\,\theta + cos^2\theta} = 1$$

Plots

One of your most important objectives as a computational astrophysicist is to convince other people that your work is of good quality. Plots make a very strong impression.

It is worth investing time and effort in making publication quality plots, even for your homework. Publication quality means your plots look like those you see in papers. Consider especially:

- Clearly labelled axes (learn to use the fontsize argument!)
- Use LaTeX to format the axis and tick labels.
- Appropriate choice of tick spacing (don't use one tick per axis!)
- Titles are not strictly needed: sometimes they help, but mostly we use **captions** to explain figures.
- Make choices about colours and thicknesses for lines with the aim of helping the reader notice the most important thing about the plot quickly.
- Where possible, avoid screenshots, and embed plots as PDFs rather than PNGs, which don't scale well.

Code

There is no need to embed long code in your answers; give references to the code in your GitHub repository. Jupyter notebooks are fine for homework answers.

Question 1

You are planning to run a cosmological N-body simulation of a periodic cubic volume with a comoving side-length 100 Mpc, using a Tree-PM code. Let's call this the "NTHU Box".

Note: "Comoving" means you can ignore the expansion of the Universe in all calculations in this question; the expansion is 'factored out' by the code. In this case, any parameters that change with the expansion can be taken to have their present-day values.

The particles in your simulation are tracers of the density field of gravitating matter in the Universe. Each particle has a position, a velocity and a unique ID number. You want your simulation to resolve bound objects of total mass $\sim 10^6\,\mathrm{M}_\odot$ with 500 particles.

1. In the cosmology you're using, the mean (comoving) density of the Universe is $\sim 1.36 \times 10^{11}\,\mathrm{M}_\odot\,\mathrm{Mpc}^{-3}$. How many particles do you need in the simulation? Express your answer as the closest cube of a power of 2 (for example, $4^3, 16^3, 128^3 \ldots$ for historical reasons, that is the standard). Comment on whether this is a technically achievable simulation.

2. What, roughly, would be an appropriate gravitational softening length for this simulation? Justify your choice. (You do not have to read Power et al. 2003, but you can if you want[1]).

3. Suggest appropriate numerical representations (data types) for the particle properties. For your chosen representations, calculate the approximate memory required to store the particles (called the *particle load*). What else needs to be accounted for in calculating the total memory needed for the simulation?

   Assume the simulation will be run on hardware identical to the CPU nodes in the CICA cluster. Approximately how many such nodes would you need?

4. In a cosmological simulation, the initial distribution of matter is extremely smooth, whereas the final distribution is highly clustered. Qualitatively, how will the time required for the gravitational force calculation evolve as the simulation progresses? Why?

Skim-read the description of the Aquarius Aq-A-1 simulation in Springel et al. (2008), MNRAS 391, 4, pp. 1685-1711: https://doi.org/10.1111/j.1365-2966.2008.14066.x.

5. How does the mass and spatial resolution of the Aq-A-1 simulation compare with the "NTHU Box" simulation? Could Aq-A-1 be run on the CICA cluster?

6. Aquarius is a cosmological simulation run with Gadget-3 (also a Tree-PM code). What is the main difference between Aquarius and the "NTHU Box"? How did that make it possible to run Aquarius more than 10 years ago?

---

[1] https://ui.adsabs.harvard.edu/abs/2003MNRAS.338...14P/abstract

Question 2

The purpose of this question is to experiment with solving an ODE system from a real paper using the machinery in Scipy.

Cole et al. 2000 (MNRAS 319, 1, pp. 168-204) [https://ui.adsabs.harvard.edu/abs/2000MNRAS.319..168C/abstract] describes the GALFORM **semi-analytic model** of galaxy formation. **This paper is a classic, but do not worry about reading the whole paper in detail.**

Fundamentally, this model is a system of ODEs that describe the rate of flow of baryonic mass (ordinary matter, mostly hydrogen atoms) between different, broadly-defined "phases" within the gravitational potential of a dark matter halo. In this model, the phases are "hot gas" (in hydrostatic equilibrium in the halo), "cold gas" (in the galactic disk) and "stars". These phases can exchange mass: for example, hot gas can radiate energy and condense, cold gas can collapse into stars, and supernovae can heat up the cold gas. This is called the **baryon cycle**.

This is described in section 4.2 of the paper. Read that section and look at equations (4.6), (4.7) and (4.8); you can ignore the equations (4.9)-(4.11) which describe the metallicity of the gas. You can ignore all the other stuff that goes on around this ODE system to set its initial conditions (dark matter halo growth, galaxy sizes, galaxy mergers etc.). Just concentrate on those ODEs.

In this case, the system has an analytic solution under certain assumptions (see Appendix B). The analytic model is used to solve for the star formation rate of the system on timescales when the cooling rate of gas from the hydrostatic halo can be approximated as constant.

[30 marks] Using the extra information below and one of the ODE solvers in Scipy, write and validate code that solves these three equations numerically. Demonstrate that your solution matches the analytic solution in Appendix B for a range of different parameter choices.

The parameters to adjust are are $\dot{M}_{\rm cool}$, $\alpha_{\rm hot}$, $V_{\rm hot}$ (*in the complete model,* $\dot{M}_{\rm cool}$ *is not free — it is a function of the mass and mass accretion rate of the dark matter halo, which determine the mass available for cooling, and the radiative physics of the hot gas, i.e. the rate at which the hot gas loses thermal energy and flows towards the galaxy*).

Use equations (4.14) and (4.15) to calculate $\tau_\star$ and β. Assume $R = 0.49$, $V_{\rm disc} = 220\,{\rm km\,s^{-1}}$ and $r_{\rm disc} = 3\,{\rm kpc}$ (i.e. a Milky Way-like galaxy). For simplicity, fix $\epsilon_\star = 0.01$ and $\alpha_\star = 0$. For initial conditions, assume $M_{\rm cold} = 1 \times 10^{10}\,{\rm M_\odot}$. You should read the paper and try to understand what these quantities mean.

For the free parameters, you can skim through the paper to find physically reasonable values to explore. You will probably need to understand what the parameters are supposed to mean and use some common-sense astrophysical intuition.

Provide your code and plots via your GitHub repository in a folder named "homework1".

[20 marks] Explore the solution under conditions where $\dot{M}_{\rm cool}$ is not constant and the total mass of gas available in the hot reservoir is finite (i.e. it can run out, but can also be replenished by transfer from cold to hot). Describe any technical problems you encounter.

This last part could be quite challenging! Experiment and see what you find. Points will be awarded generously for effort, clear presentation, plots and creativity.