Computational Physics Project / Gravity on a mesh

This project focuses on solving Poisson's equation on a uniform mesh. The problem you will solve here is *one piece* of a larger technique known as *particle-mesh* which is often used in the context of simulating the formation of structure in the universe and in other contexts.

In order to follow how structure forms due to gravity, the particle-mesh technique approximates the distribution of matter using a large set of particles in 3-dimensions. The simulation stars at some initial time step with some initial conditions for the particles. The density field is approximated by counting the number of particles in each cell of a rectangular mesh. This yields a 3D array containing the density field. Then, Poisson's equation is solved for the gravitational potential. The acceleration on each particle can then be calculated as the gradient of the potential. The velocity of the particle is updated, and its position is updated according to the velocity. Then the process is repeated with the new (slightly different) density field. The basics of this method were described thoroughly in a classic book by Hockney & Eastwood (1988), *Computer Simulation Using Particles*.

In this project, you will just perform and test the solution of Poisson's equation for a density field that you define. Then you will play with integrating orbits in this potential (without updating the potential). Chapter 19 of Landau has some information about this *but* in the gravitational case you cannot do exactly what it says there.

1. Defining a density field

Your first task will be to define a 3D density field. For this purpose, you should use a multivariate Gaussian distribution. That is, the density field will have the form:

$$\rho(\vec{x}) = \left(\frac{1}{\sqrt{2\pi}\det\mathbf{C}}\right)^{1/2} \exp\left(-\frac{(\vec{x}-\vec{x}_0)\cdot\mathbf{C}^{-1}\cdot(\vec{x}-\vec{x}_0)}{2}\right) \tag{1}$$

where $\mathbf{C}$ is the *covariance matrix* of the Gaussian and $\vec{x}_0$ is the location of the peak of the Gaussian. It is diagonal with equal elements for a spherically symmetric Gaussian.

- Write a routine to create a mesh with such a density field taking the covariance, center, and the size of the mesh as input. You should put this routine into a module that you can call.
- Then write a routine to plot slices of the mesh, plus any other sort of visualization that you want to explore, to show that the Gaussian is behaving as you expect in the following cases.
- Experiment with different axis ratios for the Gaussian by changing the diagonal elements.
- Experiment with using principal axes of the Gaussian not aligned with the mesh. Note you should *rotate* the covariance matrix elements; just making up random off-diagonal elements will not necessarily behave as you expect.

2. Solving Poisson's equation

Then you need to solve Poisson's equation:

$$\nabla^2 \phi(\vec{x}) = -4\pi\rho(\vec{x}) \tag{2}$$

You should use Fourier methods to perform the solution — i.e. use the fact that the Fourier transform of the above equation is:

$$k^2\tilde{\phi}(\vec{k}) = -4\pi\tilde{\rho}(\vec{k}) \tag{3}$$

A very similar case is described in the book. However, unlike the electrostatics cases described in the book, you do not have boundary conditions of ϕ at the boundary. Instead, you need to use a trick to treat the mass distribution on the mesh as isolated. James (1977) describes this method.

- Write a function in your module to solve Poisson's equation using this method.
- Test this method by making a relatively small spherical, Gaussian making sure that at large distances you get the Keplerian potential you expect.
- Test this method further by making a bigger spherical Gaussian and making sure that even where the density is high you get the potential you expect.
- Now explore a prolate or oblate Gaussian. Use a fairly extreme axis ratio (e.g. 0.1) and compare the density and the potential. Does the potential have the same axis ratio?

3. Bonus: integrating orbits in this potential

Once you have a potential calculated, you should be able to integrate orbits through it according to the ordinary differential equation:

$$\begin{aligned} \dot{\vec{v}} &= -\vec{\nabla}\phi \\ \dot{\vec{x}} &= \vec{v} \end{aligned} \tag{4}$$

You can use Runge-Kutta.

- Try starting a particle along a circular orbit with the right velocity in a spherical potential. Does it stay on a circular orbit?
- Try making the potential non-spherical. What happens? Under what conditions does the orbit stay spherical?