An N-Body Code for Solving the Cosmological Vlasov-Poisson Equations in 2D

Elizabeth Berzin

May 5 2022

This report represents my own work in accordance with University regulations.

1 Introduction

Dark-matter can be modelled as a collisionless system, meaning that its evolution is described by the Vlasov-Poisson equations [1]:

$$\frac{\partial f}{\partial t} + \mathbf{v}\frac{\partial f}{\partial \mathbf{r}} - \frac{\partial \Phi}{\partial \mathbf{r}}\frac{\partial f}{\partial \mathbf{v}} = 0 \tag{1}$$

$$\nabla^2 \Phi = 4\pi G m \int f(\mathbf{r}, \mathbf{v}, t) d\mathbf{v} \tag{2}$$

In the above, $f(\mathbf{r}, \mathbf{v}, t)$ is the dark-matter distribution. The real space density is given in terms of the distribution function by

$$\rho = m \int f(\mathbf{r}, \mathbf{v}, t) d\mathbf{v} \tag{3}$$

In an N-body model, the collisionless fluid described by the above Vlasov-Poisson system is discretized into particles. Rather than summing individual forces between particles, we proceed with a particle-mesh technique [2], which computes gravitational forces using an interpolated density field.

Following this general framework, the simulation proceeds as follows:

1. Generate appropriate initial conditions.
2. Calculate the mass density of particles on a fine mesh.
3. Solve the Poisson equation under the above mass density to find the potential on the mesh.
4. Calculate the acceleration on the mesh, by taking the gradient of the potential.
5. Interpolate the acceleration on the mesh back to the particles.
6. Update particle velocity and position using the appropriate equations of motion.

2 Coordinates

In cosmological simulations, it is customary to use co-moving coordinates $\mathbf{x}$ instead of physical coordinates $\mathbf{r}$. These coordinate systems are related by the cosmological scale factor $a(t)$:

$$\mathbf{x} = \frac{\mathbf{r}}{a(t)}, \qquad \mathbf{p} = a(t)\mathbf{v}, \tag{4}$$

The evolution of the cosmological scale factor is given by the Friedman equation [3]:

$$\frac{\dot{a}}{a} = H_0\sqrt{\Omega_\Lambda + \Omega_m a^{-3} + (1-\Omega_k)a^{-2}}, \tag{5}$$

where H_0 is the Hubble constant, Ω_m is the matter density, Ω_Λ is the dark-energy density, and Ω_k is the spatial curvature density. Using this change of coordinates, the Poisson equation for an expanding universe can be written as

$$\frac{1}{a^2}\nabla^2\phi = 4\pi G\bar{\rho}\delta, \tag{6}$$

where $\delta + 1 = \rho/\bar{\rho}$ is the matter density contrast, $\bar{\rho}$ is the mean matter density, and ϕ is the corresponding peculiar gravitational potential. Similarly, Newton's second law can be rewritten as

$$\frac{\partial}{\partial t}(a\mathbf{v}) = -\nabla\phi, \tag{7}$$

where $\mathbf{v} = a\dot{\mathbf{x}}$ is the peculiar velocity. It is also convenient to introduce non-dimensional coordinates. Do to this, we use the following transformations:

$$\tilde{\mathbf{x}} = \frac{\mathbf{x}}{r_0}, \qquad \tilde{\mathbf{p}} = \frac{\mathbf{p}}{r_0 H_0}, \qquad \tilde{\phi} = \frac{\phi}{r_0^2 H_0^2}, \qquad \tilde{\rho} = a^3\frac{\rho}{\rho_0}, \tag{8}$$

where $\rho_0 = 3H_0^2\Omega_m/8\pi G$, and r_0 is an arbitrary length scale, which we take to be L. These transformations are adapted by those used in [4]. Under these transformations, the Poisson equation can be written as:

$$\nabla^2\tilde{\phi} = \frac{3\Omega_m}{2a}\tilde{\delta}, \tag{9}$$

where $\tilde{\delta} = \tilde{\rho} - 1$. The corresponding equations of motion, written using a as the temporal variable, are:

$$\dot{\tilde{\mathbf{x}}} = H_0\frac{\tilde{\mathbf{p}}}{\dot{a}a^2}, \qquad \dot{\tilde{\mathbf{p}}} = -H_0\frac{\nabla\tilde{\phi}}{\dot{a}} \tag{10}$$

3 Force Calculation

To perform the mass assignment, we use the cloud-in-cell method, where the fraction of mass assigned to a given cell is weighted by the distance between a particle and its neighboring cells. For a given position $\tilde{\mathbf{x}}$, the particle coordinate in grid units is given by $\tilde{\mathbf{p}}_f = \tilde{\mathbf{x}}/h$, where $h = L/N_g$. The index of a cell is then given by $\lfloor\tilde{\mathbf{p}}_f\rfloor$. To quantify the degree of overlap, we define $\mathbf{p}_c = \tilde{\mathbf{p}}_f - \lfloor\tilde{\mathbf{p}}_f\rfloor$, which is a number between 0 and 1. In two dimensions, the density assigned to cell (i, j) is given by:

$$\rho_{pq} = \frac{1}{h^2}\sum_{i=1}^{N} m_i W_{pq}(\tilde{\mathbf{x}}_i), \tag{11}$$

where for a given particle i in cell (p,q):

$$W_{p,q} = (1 - p_c)(1 - q_c) \tag{12}$$
$$W_{p+1,q} = p_c(1 - q_c) \tag{13}$$
$$W_{p,q+1} = (1 - p_c)q_c \tag{14}$$
$$W_{p+1,q+1} = p_c q_c \tag{15}$$

The solution to the Poisson equation using the calculated density field will then be performed using a multigrid method, following the implementation given in Lecture 11. Under this implementation, a Jacobi solver is used as the base iteration. Once multigrid converges on a potential ϕ, we calculate the acceleration on the mesh using a fourth-order central finite difference scheme:

$$\tilde{\phi}'(x_i) = \frac{-\tilde{\phi}_{i-2} + 8\tilde{\phi}_{i-1} - 8\tilde{\phi}_{i+1} + \tilde{\phi}_{i+2}}{12h}. \tag{16}$$

4 Time Integration

The acceleration defined on the mesh is first interpolated back to the particles using an inverse cloud-in-cell scheme. Because this system is represented by a separable Hamiltonian, we use the symplectic second-order leapfrog scheme. Under this scheme, velocities and positions are staggered in time by a half-step. In order words, at a particular time $a(t_n)$, we have momenta $\tilde{\mathbf{p}}_{n-1/2}$ and positions $\tilde{\mathbf{x}}_n$. The update proceeds as:

$$\tilde{\mathbf{p}}_{n+1/2} = \tilde{\mathbf{p}}_{n-1/2} - H_0 \frac{\nabla\tilde{\phi}_n}{\dot{a}_n}\Delta a, \tag{17}$$
$$\tilde{\mathbf{x}}_{n+1} = \tilde{\mathbf{x}}_n + H_0 \frac{\tilde{\mathbf{p}}_{n+1/2}}{\dot{a}_{n+1/2}a^2_{n+1/2}} \frac{\Delta a}{2}. \tag{18}$$

5 Validation

To test the functionality of the code, we perform two simple tests to validate the force calculation and time integration. To validate the force calculation, we place a particle of unit mass in the center of the domain, and randomly distribute massless test particles. We then compute the radial and tangential forces acting on each particle. The results of the force calculation test are shown in Figure 1. We see that the radial force follows the expected $1/r$ behavior in the regime unaffected by periodic boundary conditions. We also see the expected peak at $r = 1/r_0$. These results indicate that the mass distribution, multigrid solver, and gradient calculation are operating as expected.

To evaluate the time integration portion of the code, we simulate a circular orbit about a particle of unit mass, for a particle with initial velocity in the x direction. The results of this for 61 revolutions, with $\Delta t = 0.01$, are shown in Figure 2. While we see slight precession as expected from the leapfrog scheme, we find that the orbit is closed and is in close agreement with the predicted circular shape, confirming that the time integration step proceeds appropriately.

6 Cosmological Simulations

Initial conditions are found using the `MUSIC` software [5], and are re-scaled into our coordinates using the transformations described in Equation 8. We use an Einstein-de Sitter (EdS) cosmology. The results of an example simulation for $N = 256^2$ particles at four different snapshots are shown in Figure 3.

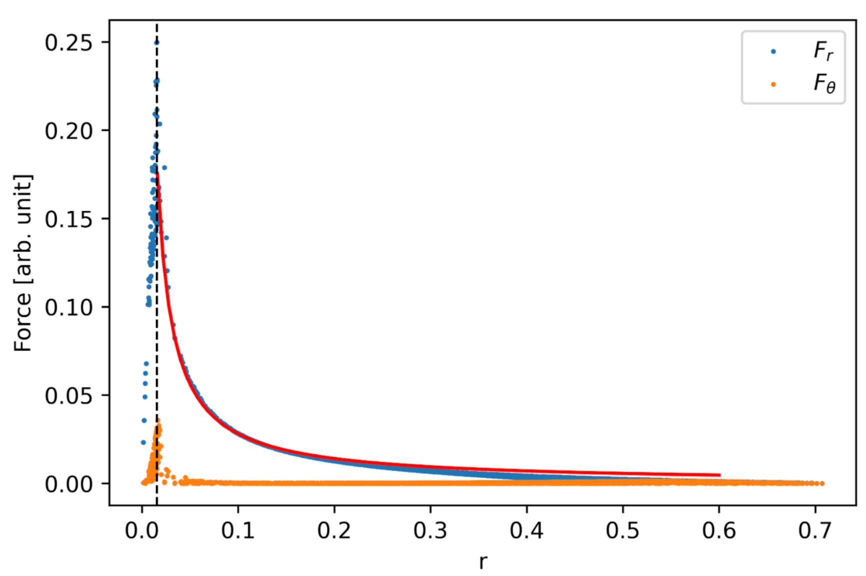

Figure 1: Radial and tangential forces evaluated at the positions of 64^2 randomly distributed massless particles, from a particle of unit mass located at the center of the domain. Multigrid was performed with 6 refinement levels. Dashed black line shows $r_0 = 1/64$, and red line shows fit to $F \propto 1/r$.

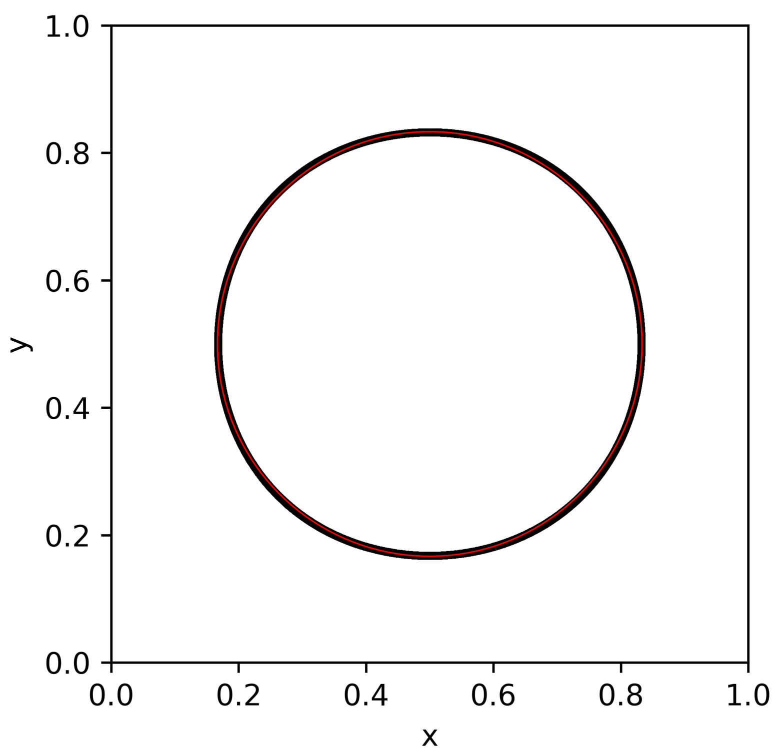

Figure 2: An example of a circular orbit used to test time integration, for a total of 61 revolutions. Red line shows the expected behavior.

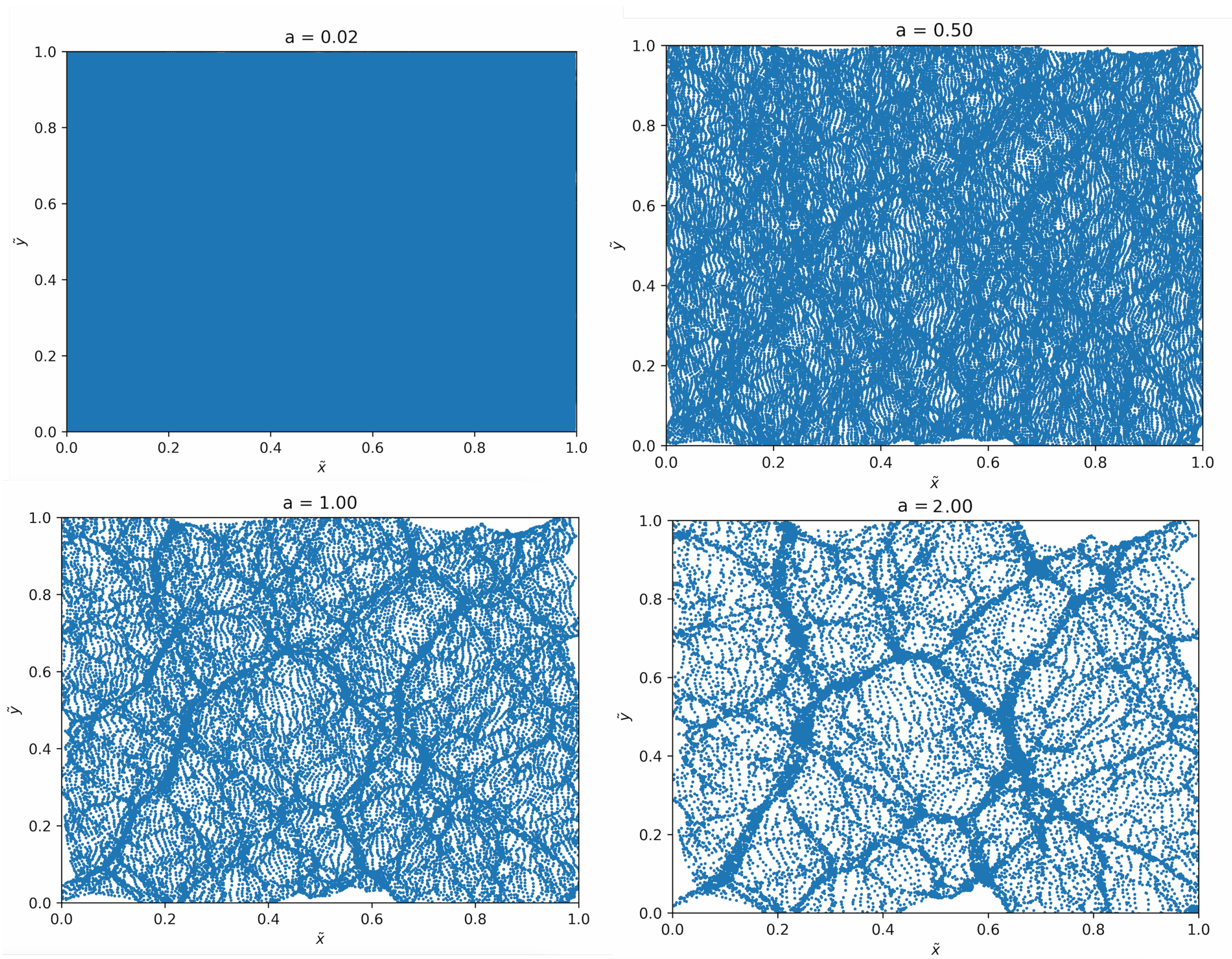

Figure 3: Particle positions at four different snapshots using an EdS cosmology.

References

[1] C. Rampf, *Cosmological Vlasov–Poisson equations for dark matter: Recent developments and connections to selected plasma problems*, Rev. Mod. Plasma Phys. **5**, 10 (2021), 2110.06265.

[2] V. Springel, *High Performance Computing and Numerical Modelling*, Saas-Fee Advanced Course **43**, 251 (2016), 1412.5187.

[3] B. Ryden, *Introduction to Cosmology* (Cambridge University Press, 2017).

[4] A. Klypin and J. Holtzman, PMCode: Particle-Mesh Code for Cosmological Simulations, Astrophysics Source Code Library, 1999, 9909.001.

[5] O. Hahn and T. Abel, *Multi-scale initial conditions for cosmological simulations*, Monthly Notices of the Royal Astronomical Society **415**, 2101 (2011).