# Selected methods in $N$-body simulations

Aleksy Bałaziński

March 31, 2025

## 1 Introduction

The dominant force over large distances is the gravitational force. The force exerted on a body with mass $m_2$ at the point $\mathbf{x}_2$ by a body with mass $m_1$ located at $\mathbf{x}_1$ can be expressed by the relation

$$\mathbf{F} = -G\frac{m_1 m_2}{|\mathbf{x}_{21}|^3}\mathbf{x}_{21}$$

where $G$ is the gravitational constant $6.674 \times 10^{-11}\mathrm{m}^3\mathrm{kg}^{-1}\mathrm{s}^{-2}$ and $\mathbf{x}_{21} = \mathbf{x}_2 - \mathbf{x}_1$. Therefore, the evolution of a system of $N$ bodies is described by $N$ equations

$$\ddot{\mathbf{x}}_i = -G\sum_{j\neq i}\frac{m_j}{|\mathbf{x}_{ij}|^3}\mathbf{x}_{ij}. \tag{1}$$

for each $i = 1, \ldots, N$. Direct application of Equation 1 is the basis the so-called *particle-particle* method. The method is characterized by $O(N^2)$ time complexity (more precisely, it requires $(N-1)N/2$ operations if Newton's 3rd law is used in the computation). Assuming that 100ns are required to perform the floating-point operations under the summation symbol, $N = 30,000$, and 150 iterations, the simulation would take approximately 2 hours to complete. It is therefore evident that more efficient algorithms have to be put in place in order to make simulations of this scale feasible.

The *particle-mesh* (PM) technique, introduced around 1985 by Hockney and Eastwood, was an early improvement over the PP method. In the PM approach, the space is divided into a rectangular grid (or mesh) of cells. Each cell is assigned a portion of the mass of nearby particles, creating a density distribution $\rho(\mathbf{x})$. The relation between the density and gravitational potential $\phi$, in the form of Poisson's equation

$$\nabla^2\phi = 4\pi G\rho, \tag{2}$$

is then used to obtain the potential at each cell center. The gravitational field $\mathbf{g}$ can then be calculated as $\mathbf{g} = -\nabla\phi$. Since $\mathbf{g}$ equals the acceleration due to gravity, we get $\ddot{\mathbf{x}}_i = \mathbf{g}(\mathbf{x}_i)$.

The drawback of the PM method is its poor modeling of forces over short distances. Eastwood and Hockney proposed a remedy for this problem: the *particle-particle-particle-mesh* method (or $\mathrm{P}^3\mathrm{M}$ in short). In the $\mathrm{P}^3\mathrm{M}$ method, the force on the $i$-th particle is split into two components: *short-range* and *long-range* force. The long-range force is calculated using the PM method, whereas the short-range force can be found by direct summation of the forces due to nearby particles.

The computational complexity of the PM and $\mathrm{P}^3\mathrm{M}$ methods depends on the implementation of the potential solver used to calculate $\phi$ from Equation 2. For instance, if a fast Fourier transform is used, then the complexity of the PM algorithm is $O(N + N_g^3 \log N_g)$, where $N_g$ is the number of cells in a single dimension of the grid (note it is linear in $N$). For the $\mathrm{P}^3\mathrm{M}$ method, the worst-case scenario happens when all particles are clustered closely together, which causes the short-range $O(N^2)$ correction part to become dominant.

## 2 Particle-mesh method

The particle-mesh method can be described as the following sequence of four steps:

1. Assign masses to mesh-points,

2. Solve the field equation (Equation 2) on the mesh,

3. Calculate the field strength at mesh-points,
4. Find forces applied to individual particles by interpolation.

In this section, each of these steps will be described in more detail.

## 2.1 Mass assignment

The specifics of the procedure of assigning mass from particles to mesh-points depend on the density profile (or *shape*) associated with the particles. In general, the particles need not be represented as idealized dimensionless points; indeed, it is possible to construct a hierarchy of shapes, where each successive member covers a larger number of mesh-points and whose application leads to smaller numerical errors.

An infinite hierarchy of shapes with this property, as described by Hockney and Eastwood in [1], can be generated by successive convolutions with the "top-hat" function $\Pi$, defined as

$$\Pi(x) = \begin{cases} 1, & |x| < \frac{1}{2} \\ \frac{1}{2}, & |x| = 1 \\ 0, & \text{otherwise.} \end{cases}$$

The three most popular assignment schemes that hail from this family (and the ones implemented in our program) are the *nearest grid point* (NGP), *cloud in cell* (CIC), and *triangular shaped cloud* (TSC) schemes, with shapes $S$ given by

$$S_{\text{NGP}} = \delta(x), \quad S_{\text{CIC}} = \delta(x) * \frac{1}{H}\Pi\left(\frac{x}{H}\right) = \frac{1}{H}\Pi\left(\frac{x}{H}\right), \quad S_{\text{TSC}} = \frac{1}{H}\Pi\left(\frac{x}{H}\right) * \frac{1}{H}\Pi\left(\frac{x}{H}\right) = \frac{1}{H}\Lambda\left(\frac{x}{H}\right),$$

where $\Lambda$ is the triangle function

$$\Lambda(x) = \begin{cases} 1 - |x|, & |x| < 1 \\ 0, & \text{otherwise.} \end{cases}$$

In the one-dimensional case, the fraction of mass $W_p$ assigned to mesh-point $p$ from particle at position $x$ is given by

$$W(x - x_p) = W_p(x) = \int_{x_p - H/2}^{x_p + H/2} S(x' - x)dx'.$$

A simple rule for relating the assignment function $W$ defined above with shape $S$ can be found by noticing that

$$W(x) = \int_{-H/2}^{H/2} S(x' - x)dx' = \int_{-\infty}^{\infty} \Pi\left(\frac{x'}{H}\right) S(x' - x)dx' = \Pi\left(\frac{x}{H}\right) * S(x).$$

This implies that

$$W_{\text{NGP}}(x) = \Pi\left(\frac{x}{H}\right), \quad W_{\text{CIC}}(x) = \Lambda\left(\frac{x}{H}\right), \quad W_{\text{TSC}}(x) = \Pi\left(\frac{x}{H}\right) * \frac{1}{H}\Lambda\left(\frac{x}{H}\right) = (\Pi * \Lambda)\left(\frac{x}{H}\right). \tag{3}$$

Splitting the domain of integration in the expression for $W_{\text{TSC}}$ into five disjoint intervals shows that

$$(\Pi * \Lambda)(x) = \begin{cases} \frac{1}{8}(3 - 2|x|)^2, & \frac{1}{2} \le |x| < \frac{3}{2} \\ \frac{3}{4} - x^2, & |x| < \frac{1}{2} \\ 0, & \text{otherwise.} \end{cases}$$

Two- and three-dimensional versions of the assignment functions in Equation 3 are products of the assignment functions in each dimension. For example, in the three-dimensional the assignment function $W$ is defined as

$$W(\mathbf{x}) = W(x)W(y)W(z).$$

Hence, the mass assigned at mesh-point at $\mathbf{x_p}$ is

$$m(\mathbf{x_p}) = \sum_i m_i W_{\mathbf{p}}(\mathbf{x}_i),$$

or, in terms of density $\rho$,

$$\rho(\mathbf{x_p}) = \frac{1}{V}\sum_i m_i W_{\mathbf{p}}(\mathbf{x}_i), \tag{4}$$

where $V = H^3$ is the volume of a cell and $i$ indexes the particles.

Obviously, Equation 4 is not suitable for direct application in the actual algorithm. Instead, we iterate over all particles, identify the parent cell of each particle (and its neighborhood) and update $\rho(\mathbf{x_p})$. This process is illustrated in Algorithm 1. The set $C_S(\mathbf{x})$ of cells that have to be considered while

**Algorithm 1** Density assignment algorithm

```
for each particle i do
    for each cell q in C_S(x_i) do
        ρ(x_q) ← ρ(x_q) + m_i W(x_i − x_q)/V
    end for
end for
```

assigning density, depends on the shape $S$ of the particle. Specifically, we have $C_{\text{NGP}}(\mathbf{x}) = \{[\mathbf{x}/H]\}$, $C_{\text{CIC}}(\mathbf{x}) = \{\lfloor \mathbf{x}/H \rfloor + \mathbf{t} \mid t_i = 0, 1\}$, and $C_{\text{TSC}}(\mathbf{x}) = \{[\mathbf{x}/H] + \mathbf{t} \mid t_i = -1, 0, 1\}$. It follows that $|C_{\text{NGP}}(\mathbf{x})| = 1$, $|C_{\text{CIC}}(\mathbf{x})| = 8$, and $|C_{\text{TSC}}(\mathbf{x})| = 27$ which illustrates the increasing computational cost resulting from using higher-order assignment schemes.

### 2.2 Solving the field equation

The Poisson equation (Equation 2) can be restated in integral form

$$\phi(\mathbf{x}) = \int G(\mathbf{x} - \mathbf{x}')\rho(\mathbf{x}')dV',$$

which has the following discrete analogue

$$\phi(\mathbf{x_p}) = V \sum_{\mathbf{p}'} G(\mathbf{x_p} - \mathbf{x_{p'}})\rho(\mathbf{x_{p'}}), \tag{5}$$

where $G$ is the Green's function (potential due to unit mass). The right-hand side of Equation 5 is a convolution sum that runs over a finite set of mesh-points. If we assume periodic boundary conditions, we can apply the discrete Fourier transform to both sides and use the convolution theorem to conclude that[1]

$$\hat{\phi}(\mathbf{k}) = \hat{G}(\mathbf{k})\hat{\phi}(\mathbf{k}). \tag{6}$$

An approximation to $\hat{G}$ can be found using a discretized version of the Laplacian in Equation 5. Specifically, for a 7-point stencil,

$$\begin{aligned} 4\pi G\rho(\mathbf{x}_{ijk}) = & \frac{\phi(\mathbf{x}_{i-1,j,k}) - 2\phi(\mathbf{x}_{ijk}) + \phi(\mathbf{x}_{i+1,j,k})}{H^2} \\ & + \frac{\phi(\mathbf{x}_{i,j-1,k}) - 2\phi(\mathbf{x}_{ijk}) + \phi(\mathbf{x}_{i,j+1,k})}{H^2}. \\ & + \frac{\phi(\mathbf{x}_{i,j,k-1}) - 2\phi(\mathbf{x}_{ijk}) + \phi(\mathbf{x}_{i,j,k+1})}{H^2}. \end{aligned}$$

After Fourier-transforming both sides, applying the shift theorem, and simplifying using the Euler's formula, we arrive at an expression for $\hat{\phi}$,

$$\hat{\phi}(\mathbf{k}) = \underbrace{-4\pi G \frac{(H/2)^2}{\sin^2\left(\frac{Hk_1}{2}\right) + \sin^2\left(\frac{Hk_2}{2}\right) + \sin^2\left(\frac{Hk_3}{2}\right)}}_{\hat{G}(\mathbf{k})} \hat{\rho}(\mathbf{k}),$$

[1] In this work, the Hockney & Eastwood definition of DFT is used, i.e.

$$D(x_p) = \frac{1}{L} \sum_{l=0}^{N-1} \hat{D}(k)e^{ikx_p}, \quad \hat{D}(k) = H \sum_{p=0}^{N-1} D(x_p)e^{-ikx_p},$$

where $x_p = pH$. The conversion between this form and another popular definition,

$$\widetilde{D_H}(k) = \sum_{p=0}^{N-1} D_H(p)e^{-i2\pi kp/N},$$

is given by

$$\widetilde{D_H}(k) = \frac{1}{H}\hat{D}\left(\frac{2\pi}{NH}k\right),$$

where $D_H(p) = pH$.

where $\hat{G}$ can be identified by comparison with Equation 6. In the implementation, values of $\hat{G}$ should be computed only once and saved for future look up.

### 2.3 Field strength calculation

The strength $\mathbf{g}$ of the gravitational field at mesh-point $\mathbf{x_p}$ can be approximated using a central difference. Our implementation currently supports two types of finite differences, described below.

The two-point finite difference operator $\mathbf{D}$, whose $x$ component is given by

$$D_x(\phi)(\mathbf{x_p}) = -\frac{\phi(\mathbf{x}_{i+1,j,k}) - \phi(\mathbf{x}_{i-1,j,k})}{2H}$$

(and analogously for the $y$ and $z$ components), is second order accurate.

The fourth-order accurate finite-difference is given by

$$D_x(\phi)(\mathbf{x_p}) = -\alpha\frac{\phi(\mathbf{x}_{i+1,j,k}) - \phi(\mathbf{x}_{i-1,j,k})}{2H} - (1-\alpha)\frac{\phi(\mathbf{x}_{i+2,j,k}) - \phi(\mathbf{x}_{i-2,j,k})}{4H},$$

where $\alpha = 4/3$.

### 2.4 Interpolation

The value of the field strength $\mathbf{g}(\mathbf{x})$ at the position particle's position $\mathbf{x}$ is calculated by interpolating the values of $\mathbf{g}$ from the neighboring mesh-points. Formally,

$$\mathbf{g}(\mathbf{x}) = \sum_{\mathbf{p}} W(\mathbf{x} - \mathbf{x_p})\mathbf{g}(\mathbf{x_p}).$$

In practice, there is no need to sum over all mesh-points. Instead we use an algorithm analogous to Algorithm 1 to only include the cells with non-zero contribution to the sum. The method is illustrated in Algorithm 2. It is important to note that in order to retain correct physical behavior, the interpolation

**Algorithm 2** Field strength interpolation

**for each** particle $i$ **do**
  **for each** cell $\mathbf{q}$ in $C_S(\mathbf{x}_i)$ **do**
    $\mathbf{g}(\mathbf{x}_i) \leftarrow \sum_{\mathbf{q}} W(\mathbf{x}_i - \mathbf{x_q})\mathbf{g}(\mathbf{x_p})$
  **end for**
**end for**

and mass assignment schemes must use the same shape to represent the particles.

## 3 Galaxy model

The model of a galaxy used as a test bed for the implementation is a simple one. The galaxy is assumed to be composed of only two parts: a thin disk and a central bulge. The disk comprises a number of particles, each representing some number of stars. The central bulge is simulated as a fixed external gravitational field.

### 3.1 Disk

The disk particles are sampled from a radial distribution

$$p(r) = \frac{3}{\pi R_D^2}\left(1 - \frac{r}{R_D}\right),$$

where $R_D$ is the radius of the disk and $r \le R_D$. The cumulative distribution function is therefore

$$F(r,\phi) = \int_0^{\phi}\int_0^{r} p(r')r'dr'd\phi' = \frac{\phi}{2\pi R_D^3}(3R_D r^2 - 2r^3)$$

and the marginal CDFs are

$$F_R(r) = F(r, 2\pi) = \frac{1}{R_D^3}(3R_D r^2 - 2r^3) \quad \text{and} \quad F_\Phi(\phi) = F(R_D, \phi) = \frac{\phi}{2\pi}.$$

Now we use inverse transform sampling to generate initial positions $(r, \phi)$ for the particles, i.e. $\phi = 2\pi u$ and $r$ is given implicitly by $h(r) \equiv 2r^3 - 3R_D r^2 + uR_D^3 = 0$ with $u \sim U(0,1)$. Straightforward calculation shows that $dh/dr < 0$ for $0 < r < R_D$ and $h(0)h(R_D) < 0$ implying that $h$ has exactly one zero between 0 and $R_D$ (which can be found for example using Newton's method).

Strength of the gravitational field $\mathbf{g}$ at point $\mathbf{x}_0$ is

$$\mathbf{g}_D = G \int_0^{2\pi} \int_0^{R_D} \sigma(r) \frac{\mathbf{x} - \mathbf{x}_0}{|\mathbf{x} - \mathbf{x}_0|^3} r dr d\phi,$$

where $\sigma(r) = \sigma_0(1 - r/R_D)$ describes the density profile of the disk for $r \leq R_D$. If $M_D$ is the total mass of the disk, then $\sigma_0 = 3M_D/(\pi R_D^2)$. By symmetry, the point $\mathbf{x}_0$ may be chosen to lie on the $x$-axis, i.e. $\mathbf{x}_0 = (-x_0, 0)$, so that $\mathbf{x} - \mathbf{x}_0 = (x_0 + r\cos\phi, r\sin\phi)$. Letting $\bar{r} = r/R_D$ and $\bar{x}_0 = x_0/R_D$, the integral becomes

$$\mathbf{g} = G\sigma_0 \int_0^{2\pi} \int_0^1 (1 - \bar{r}) \frac{(\bar{x}_0 + \bar{r}\cos\phi, \bar{r}\sin\phi)}{|(\bar{x}_0 + \bar{r}\cos\phi, \bar{r}\sin\phi)|^3} \bar{r} d\bar{r} d\phi.$$

By symmetry $g_{D,y} = 0$ and thus the radial component of the field $\mathbf{g}_D$ at distance $\bar{x}_0$ from the center is

$$g_{D,r} = -G\sigma_0 \int_0^{2\pi} \int_0^1 (1 - \bar{r}) \frac{\bar{x}_0 + \bar{r}\cos\phi}{(\bar{x}_0^2 + \bar{r}^2 + 2\bar{x}_0\bar{r}\cos\phi)^{3/2}} \bar{r} d\bar{r} d\phi. \tag{7}$$

If the disk had constant density, $\mathbf{g}_D$ could be expressed in terms of elliptic integrals [2]. However, to the best of the author's knowledge, the integral in Equation 7 cannot be further simplified. For this reason, a crude approximation with a quadratic was used: $g_{D,r} \approx a(r - h)^2 + k$, where the values $k = 2.5$ and $h = 0.66$ (the maximum of $g_{D,r}$ and the argument thereof) were estimated based on the graph of $g_{D,r}$ (see Figure 1). The value of $a = -k/h^2$ can be found by setting $g_{D,r}(0) = 0$ in the approximate formula.

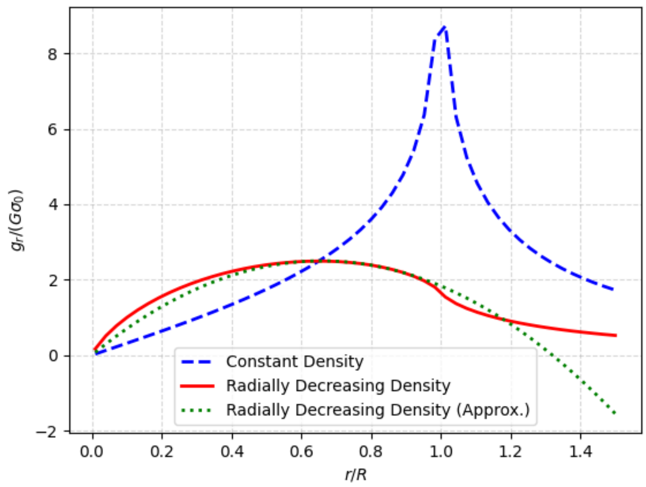


Figure 1: Radial component of the field strength due to a disk

## 3.2 Central bulge

The density profile of the central bulge is analogous to the one used for the disk, save for the fact it is 3-dimensional, i.e.

$$\rho(r) = \begin{cases} \rho_0 \left(1 - \frac{r}{R_B}\right), & r \leq R_B \\ 0, & \text{otherwise,} \end{cases}$$

where $R_B$ is the radius of the bulge. If we let $M_B$ be the mass of the bulge, then $\rho_0 = 3M_B/(\pi R_B^3)$. Application of Gauss's law shows that we have

$$g_{B,r} = -GM_B \times \begin{cases} \frac{r}{R_B^3} r \left(4 - \frac{3r}{R_B}\right), & r \leq R_B \\ \frac{1}{r^2}, & \text{otherwise.} \end{cases}$$

### 3.3 Initial conditions

The total field $\mathbf{g} = \mathbf{g}_D + \mathbf{g}_B$ is used to find initial velocities for the particles with initial positions $(x, y, 0)$. The formula for the centripetal force yields

$$\frac{v^2}{r} = -g_r$$

and thus

$$\mathbf{v} = \left(-v\frac{y}{r}, v\frac{x}{r}, 0\right)$$

with $v = \sqrt{-rg_r}$ for counter-clockwise rotation.

## 4 Particle-particle particle-mesh method

The $P^3M$ algorithm is a hybrid method: forces between distant particles are calculated using the PM method, whereas for particles lying closely together the PP method is used. The total force applied to particle $i$ is

$$\mathbf{F}_i^{\mathrm{SR}} + \mathbf{F}_i = \sum_{j \neq i} (\mathbf{f}_{ij}^{\mathrm{tot}} - \mathbf{R}_{ij}) + \mathbf{F}_i, \tag{8}$$

where $\mathbf{F}_i \approx \sum_{j\neq i} \mathbf{R}_{ij}$ is the force computed using the PM method and $\mathbf{R}_{ij} = \mathbf{R}(\mathbf{x}_i - \mathbf{x}_j)$ is a prescribed *reference force*. The reference force is defined as the force between two particle-clouds, i.e. each particle is represented by a sphere with diameter $a$ and a given density profile. The two examples of reference forces described in [1] are

$$R(r) = G \times \begin{cases} \frac{1}{35a^2}(224\xi - 224\xi^3 + 70\xi^4 + 48\xi^5 - 21\xi^6), & 0 \leq \xi \leq 1 \\ \frac{1}{35a^2}(12/\xi^2 - 224 + 896\xi - 840\xi^2 + 224\xi^3 + 70\xi^4 - 48\xi^5 + 7\xi^6), & 1 < \xi \leq 2 \\ \frac{1}{r^2}, & \xi > 2 \end{cases}$$

where $\xi = 2r/a$ for a sphere with uniformly decreasing density and

$$R(r) = G \times \begin{cases} \frac{1}{a^2}(8r/a - 9r^2/a^2 + 2r^4/a^4), & r < a \\ \frac{1}{r^2}, \end{cases}$$

for a solid sphere.

### 4.1 Optimal Green's function

As it is apparent from Equation 8, the method's validity depends on how well the reference force is approximated by the mesh force. The average deviation between the two forces can be minimized by a suitable choice of the Green's function. The details of the derivation are highly nontrivial and the can be found in [1]; in this work we restrict ourselves to presenting the results (essential to the implementation) obtained therein.

The optimal influence function $\hat{G}$ is given by

$$\hat{G}(\mathbf{k}) = \frac{\hat{\mathbf{D}}(\mathbf{k}) \cdot \sum_{\mathbf{n}} \hat{U}^2(\mathbf{k_n})\hat{\mathbf{R}}(\mathbf{k_n})}{|\hat{\mathbf{D}}|^2 \left[\sum_{\mathbf{n}} \hat{U}^2(\mathbf{k_n})\right]^2}.$$

## References


[1] R. W. Hockney and J. W. Eastwood. *Computer Simulation Using Particles.* CRC Press, 1st edition, 1988.

[2] J. Weiss. Certain aspects of the gravitational field of a disk. *Applied Mathematics*, 9:1360–1377, 2018.