# 1 N-body systems

**Notation**

$N$-body system in which each body is represented by a point particle with position and velocity

$$\mathbf{r}_i = (r_{ix},\, r_{iy},\, r_{iz})^\top$$
$$\mathbf{v}_i = (v_{ix},\, v_{iy},\, v_{iz})^\top .$$

Positions and velocity vector

$$\mathbf{r}^{3N} = (\mathbf{r}_1,\, ...,\, \mathbf{r}_N)^\top \in \mathbb{R}^{3N}.$$
$$\mathbf{v}^{3N} = (\mathbf{v}_1,\, ...,\, \mathbf{v}_N)^\top \in \mathbb{R}^{3N}.$$

Mass matrix

$$\mathbf{M} = \mathrm{diag}(m_1,\, m_1,\, m_1,\, m_2,\, m_2,\, m_2,\, ...,\, m_N,\, m_N,\, m_N)$$

Momenta

$$\mathbf{p}^{3N} = \mathbf{M}\mathbf{v}^{3N}.$$

**Energy and force**

*Force field*: Force on particle at position $\mathbf{r} = (r_x,\, r_y,\, r_z)$

$$\mathbf{F}(\mathbf{r}) = (f_x(\mathbf{r}),\, f_y(\mathbf{r}),\, f_z(\mathbf{r}))^\top$$

*Conservative force* can be written as the negative gradient of a **potential**, $\Phi$

$$\mathbf{F}(\mathbf{r}) = -\nabla\Phi(\mathbf{r})$$

where $\Phi$ is a scalar *potential field*.

**Kinetic energy**: While the potential energy of the system is often based on a model, the kinetic energy of the system is well defined for an $N$-body system.

$$K(\mathbf{p}^{3N}) = \frac{1}{2}\sum_i m_i\,|\mathbf{v}_i|^2 = \frac{1}{2}(\mathbf{p}^{3N})^\top\mathbf{M}^{-1}\mathbf{p}^{3N}$$

where we have used the definition of the mass matrix $\mathbf{M}$ introduced above.

# 2 Dynamics and searching configurations

**Dynamics**:

Newton's second law

$$\mathbf{f} = \mathbf{M}\mathbf{a}$$

with

$$\mathbf{a} = \frac{d}{dt}\mathbf{v}$$
$$\mathbf{v} = \frac{d}{dt}\mathbf{r}$$

MD is the numerical integration of Newton's equations:

$$\mathbf{p}_i(t) = \mathbf{p}_i(0) + \int_{t'=0}^{t} \mathbf{f}_i(t')\,dt'$$
$$\mathbf{r}_i(t) = \mathbf{r}_i(0) + \mathbf{M}^{-1}\int_{t'=0}^{t} \mathbf{p}_i(t')\,dt'$$

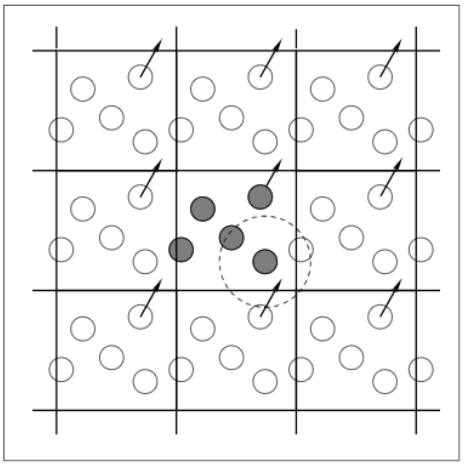
Different integration schemes have different numerical properties. E.g. explicit Euler, velocity verlet.

Closed system is characterized by the **conservation of total energy in time**, i.e.:

$$H(\mathbf{r}^{3N}, \mathbf{p}^{3N}) = \Phi(\mathbf{r}^{3N}) + K(\mathbf{p}^{3N})$$

Microcanonical ensemble.

**Periodic boundary conditions**:



**Stochastic dynamics**:

**Temperature** can be defined *via* the mean kinetic energy $K/N$, by:

$$T = \frac{2}{3k_B}\frac{K}{N}$$

Boltzmann distribution: for each pair $(\mathbf{r}^{3N}, \mathbf{p}^{3N})$ we have the following probability of occuring in the ensemble:

$$\mathbb{P}(\mathbf{r}^{3N}, \mathbf{p}^{3N}) \propto \exp\left(-\frac{H(\mathbf{r}^{3N}, \mathbf{p}^{3N})}{k_B T}\right).$$

Overdamped Langevin dynamics

$$\frac{d}{dt}\mathbf{r}^{3N} = \frac{1}{k_B T}\mathbf{D}\mathbf{f}(\mathbf{r}^{3N}) + \sqrt{2}\mathbf{D}^{\frac{1}{2}}\boldsymbol{\eta}(t)$$

where $\mathbf{D} = \mathrm{diag}(d_{1x}, d_{1y}, d_{1z}, ..., d_{Nx}, d_{Ny}, d_{Nz})$ is a diagonal matrix of diffusion coefficients.

**Monte Carlo simulation**

Density in configuration space

$$\mathbb{P}(\mathbf{r}^{3N}) \propto \exp\left(-\frac{\Phi(\mathbf{r}^{3N})}{k_B T}\right).$$

Metropolis Monte Carlo

1. Start with initial configuration $\mathbf{r}^{(0)}$ and set $k = 0$.
2. For $k = 1$, ...., $K$:
   (a) Sample random vector $\boldsymbol{\eta} \in \mathbb{R}^N$ with $\eta_i \sim \mathcal{N}(0, \sigma^2)$, i.e. an isotropic multivariate Gaussian distribution with variances $\sigma^2$.
   (b) Propose new configuration $\mathbf{r}' = \mathbf{r}^{(k)} + \boldsymbol{\eta}$.
   (c) Accept new configuration with probability

$$p_{\mathrm{acc}} = \min\left\{1, \exp\left(-\frac{\Phi(\mathbf{r}^{'}) - \Phi(\mathbf{r}^{(k)})}{k_B T}\right)\right\}$$

   If accepted, set $\mathbf{r}^{(k+1)} = \mathbf{r}^{'}$, otherwise $\mathbf{r}^{(k+1)} = \mathbf{r}^{(k)}$.

**Optimization**

Local search for low-energy configurations

1. Start with initial configuration $\mathbf{r}^{(0)}$ and set $k = 0$.
2. Increment $k = 1, ..., K$ until convergence criterion met:
   (a) Move configuration $\mathbf{r}^{(k)} \to \mathbf{r}^{(k+1)}$ in such a way that $\Phi(\mathbf{r}^{(k+1)}) \leq \Phi(\mathbf{r}^{(k)})$.
3. Return $\mathbf{r}^{(K)}$, $\Phi(\mathbf{r}^{(K)})$.

**Computational cost**

$N^2$ interaction energies or forces for $N$ particles

# 3 Forces

**Electrostatics**

Poisson equation:

$$\nabla^2 \Phi(\mathbf{r}) = -\frac{1}{\epsilon_0}\rho(\mathbf{r})$$

Gaussian charge density:

$$\rho_\sigma^G(\mathbf{r}) = \frac{q}{\sigma^3(2\pi)^{3/2}} \exp\left(-\frac{|\mathbf{r}|^2}{2\sigma^2}\right)$$
$$\Phi_\sigma(\mathbf{r}) = \frac{q}{4\pi\epsilon_0|\mathbf{r}|}\text{erf}\left(\frac{|\mathbf{r}|}{\sqrt{2}\sigma}\right)$$

Point charge:

$$\rho_i(\mathbf{r}) = q_i\delta(\mathbf{r} - \mathbf{r}_i)$$

Potential and force field at position $\mathbf{r}$ due to charge $q_i$ at $\mathbf{r}_i$:

$$\Phi_i(\mathbf{r}) = \frac{q_i}{4\pi\epsilon_0|\mathbf{r} - \mathbf{r}_i|}$$
$$\mathbf{F}_i(\mathbf{r}) = -\frac{q_i}{4\pi\epsilon_0 |\mathbf{r} - \mathbf{r}_i|^2}\frac{\mathbf{r} - \mathbf{r}_i}{|\mathbf{r} - \mathbf{r}_i|}$$

Interaction potential and force:

$$e_{ij}(\mathbf{r}^{3N}) = q_i\Phi_j(\mathbf{r}_i) = \frac{q_iq_j}{4\pi\epsilon_0|\mathbf{r}_j - \mathbf{r}_i|}$$
$$\mathbf{f}_{ij}(\mathbf{r}^{3N}) = q_i\mathbf{F}_j(\mathbf{r}_i) = -\frac{q_iq_j}{4\pi\epsilon_0 |\mathbf{r}_j - \mathbf{r}_i|^2}\frac{\mathbf{r}_j - \mathbf{r}_i}{|\mathbf{r}_j - \mathbf{r}_i|}$$

**Gravity**

Force and potential

$$e_{ij}(\mathbf{r}^{3N}) = G\frac{m_im_j}{|\mathbf{r}_j - \mathbf{r}_i|}$$
$$\mathbf{f}_{ij}(\mathbf{r}^{3N}) = -G\frac{m_im_j}{|\mathbf{r}_j - \mathbf{r}_i|^2}\frac{\mathbf{r}_j - \mathbf{r}_i}{|\mathbf{r}_j - \mathbf{r}_i|}$$

**Lennard-Jones**

Lennard-Jones

$$e_{ij}(\mathbf{r}^{3N}) = 4\epsilon_{ij}\left[\left(\frac{\sigma_{ij}}{|\mathbf{r}_i - \mathbf{r}_j|}\right)^{12} - \left(\frac{\sigma_{ij}}{|\mathbf{r}_i - \mathbf{r}_j|}\right)^{6}\right]$$

# 4 Particle-particle methods

Suppose that we have a system with $N$ particles and the interaction energy between particles $i$ and $j$ is given by $e_{ij}(r_{ij})$, where $r_{ij} = |\mathbf{r}_i - \mathbf{r}_j|$ is the interparticle distance. The direct approach to compute interactions is:

$E \leftarrow 0$

`For` $i = 1, ..., N-1$:

  $E_i \leftarrow 0$

  `For` $j = i+1, ..., N$:

    $E_i \leftarrow E_i + e_{ij}(r_{ij})$

  $E \leftarrow E + E_i$

This algorithm has obviously a computational complexity of $\mathcal{O}(N^2)$, with $\frac{N(N-1)}{2}$ interactions to be computed. For large $N$ this approach becomes too expensive.

**Cutoff**:

- Simple cutoff
- Shift method avoids discontinuity in energy
- Switch method avoids discontinuity in energy and force

When using a cutoff scheme, each particle has a neighborhood volume of $v = \frac{4}{3}\pi r_c^3$ and thus on average $n = \frac{4}{3}\pi r_c^3 \rho$ neighbors. By convention $\rho$ is a constant, i.e. independent of $N$. Each particle interaction only needs to be computed once, thus once we have the neighborhoods, we have to compute $N\frac{n-1}{2}$ interactions, and the complexity has dropped to $\mathcal{O}(N)$.

**Errors**: General interaction law

$$e(r_{ij}) = \frac{a}{r_{ij}^p}$$

Evaluate the absolute energy error $e_R$with cutoff $R$:

1. Long-ranged interactions $p < 3$: With $R = \alpha R_{max}$:

   $$e_R = \frac{4\pi\rho a}{p-3} R_{max}^{3-p} \left[\alpha^{3-p} - 1\right]$$

   has system-size dependent error up to all $\alpha < 1$ and $\lim_{R_{max}\to\infty} e_R = \infty \quad \forall R < \infty$.

2. Short-ranged interactions $p > 3$ with $R_{max} \to \infty$:

   $$e_R = \frac{4\pi\rho a}{p-3} \frac{1}{R^{p-3}}$$

   has a proper limit with $\lim_{R\to\infty} e_R = 0$ and decay is with $R^{-1}$ or faster.

| long-ranged | short-ranged |
|---|---|
| Coulomb interaction | Dipole-Dipole and higher multipole interactions |
| Charge-dipole interaction | Debye-Hückel (screened electrostatics) |
| Gravity | Van der Waals / Lennard Jones |

# 5 Particle-field methods

Poisson equation

$$\nabla^2 \Phi(\mathbf{r}) = -\frac{\rho(\mathbf{r})}{\epsilon_0}$$

Discretizing the Laplace operator, e.g. in 1D:

$$\nabla_1^2 f_i = \frac{f_{i-1} + f_{i+1} - 2f_i}{h}$$

For example, let us consider the 1D case with the $n+2$ grid points.

$$\phi_0 \quad \phi_1 \quad \cdots \quad \phi_i \quad \cdots \quad \phi_n \quad \phi_{n+1}$$

Now we use Dirichlet boundary conditions, i.e. we set the potential to the pre-defined values $\phi_0$ and $\phi_{n+1}$ on the boundaries. For all intermediate points, the equation

$$\phi_{i-1} + \phi_{i+1} - 2\phi_i = -h\frac{\rho}{\epsilon_0}$$

must hold. Defining the matrix $\mathbf{A} \in \mathbb{R}^{n\times n}$ and vector $\mathbf{b} \in \mathbb{R}^n$ on the points $1...n$,

$$\mathbf{A} = \begin{bmatrix} -2 & 1 & & & \\ 1 & -2 & 1 & & \\ & \ddots & \ddots & \ddots & \\ & & 1 & -2 & 1 \\ & & & 1 & -2 \end{bmatrix}$$

$$b_i = \begin{cases} -\frac{h}{\epsilon_0}\rho_i - \phi_0 & i = 1 \\ -\frac{h}{\epsilon_0}\rho_i - \phi_{n+1} & i = n \\ -\frac{h}{\epsilon_0}\rho_i & \text{else} \end{cases}$$

we solve $\mathbf{Ax} = \mathbf{b}$.

**Complexity**

Direct Gauss algorithm: $\mathcal{O}(n^3)$, where $n$ is the number of discretization points.

Sparse iterative methods:

- Jacobi method $\mathcal{O}(L^2)$ sparse matrix-vector multiplications, where $L$ is the length in each dimension
- Gauss-Seidel method $\mathcal{O}(L^2)$ sparse matrix-vector multiplications
- Successive over-relaxation method $\mathcal{O}(L)$ sparse matrix-vector multiplications

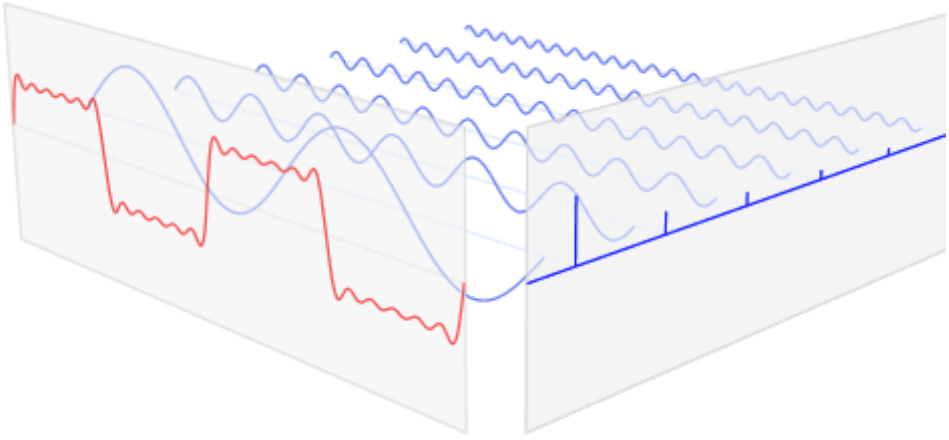
# 6 Fourier transform

Functional transform:

$$x(t) \quad \underset{\text{inverse Fourier transform}}{\overset{\text{Fourier transform}}{\rightleftarrows}} \quad X(\omega)$$

**Fourier series**

Periodic signal - discrete spectrum



$$X(\omega) = \frac{1}{T}\int_{-T/2}^{T/2} x(t)\mathrm{e}^{-i\omega t}\,\mathrm{d}t$$
$$x(t) = \sum_{\omega=-\infty}^{\infty} X(\omega)\mathrm{e}^{i\omega t}.$$

with $\omega = \frac{2\pi n}{T}$

**Fourier transform**

Nonperiodic signal - continuous spectrum (Take limit $T \to \infty$ in Fourier series):

$$X(\omega) = \int_{-\infty}^{\infty} x(t)\mathrm{e}^{-i\omega t}\,\mathrm{d}t$$
$$x(t) = \int_{-\infty}^{\infty} X(\omega)\mathrm{e}^{i\omega t}\mathrm{d}\omega.$$

Important properties:

1. **Linearity**: For any complex numbers $a$ and $b$, $\mathcal{F}\{ax(t) + by(t)\} = a\mathcal{F}\{x(t)\} + b\mathcal{F}\{y(t)\}$.
2. **Translation/Time-Shifting**: For any real number $\tau$, $\mathcal{F}\{x(t-\tau)\} = \mathrm{e}^{-i\omega\tau}\mathcal{F}\{x(t)\}$.
3. **Differentiation**: $\mathcal{F}\left\{\frac{\mathrm{d}^n}{(\mathrm{d}t)^n}x(t)\right\} = (i\omega)^n\mathcal{F}\{x(t)\}$. By applying the Fourier transform and using these formulas, some ordinary differential equations can be transformed into algebraic equations, which are much easier to solve.

**Discrete Fourier transform**

Discrete signal - discrete spectrum

$$X_k = \sum_{n=0}^{N-1} x_n \mathrm{e}^{-i2\pi\frac{nk}{N}}.$$
$$x_n = \frac{1}{N}\sum_{k=0}^{N-1} X_k \mathrm{e}^{i2\pi\frac{nk}{N}}$$

The multidimensional DFT can be computed by the composition of a sequence of one-dimensional DFTs along each dimension.

**Fast Fourier transform**

FFT(**x**):

1. Divide the signal $\mathbf{x} = (x_0, ..., x_{N-1})$ into two subsignals **a** and **b** of lengths $N/2$ as:

$$\begin{aligned} a_n &= x_{2n} \qquad n = 0, 1, ..., \frac{N}{2} - 1 \\ b_n &= x_{2n+1} \quad n = 0, 1, ..., \frac{N}{2} - 1 \end{aligned}$$

   Thus, **a** contains the even and **b** contains the odd time-indexes of the original signal.

2. Compute their DFT's as:

$$\begin{aligned} A_k &= FFT(a_n) \\ B_k &= FFT(b_n) \end{aligned}$$

3. Return:

$$X_k = \begin{cases} A_k + (W_N)^k B_k & k = 0, ..., \frac{N}{2} - 1 \\ A_{k-\frac{N}{2}} - (W_N)^k B_k & k = \frac{N}{2}, ..., N - 1 \end{cases}$$

with $W_N = \mathrm{e}^{-i2\pi\frac{1}{N}}$.

For each value of $k$, we have on the order of $\log_2$ levels, and in each level 2/2/2 additions/products/powers for each fragment. The number of data points treated at each level is still $N$, so the computational complexity per level is $\mathcal{O}(N)$. The overall complexity of the FFT is thus $\mathcal{O}(N \log_2 N)$ rather than $\mathcal{O}(N^2)$.

**Solving the Poisson equation with DFT/FFT**

1. Transform charge density to Fourier space: $P(\mathbf{k}) = \mathcal{F}\{\rho(\mathbf{r})\}$
2. Compute Fourier-space potential (using linearity and transform of differential): $\Phi(\mathbf{k}) = \frac{P(\mathbf{k})}{k^2\epsilon_0}$.
3. Transform back to real space: $\phi(\mathbf{r}) = \mathcal{F}^{-1}\{\Phi(\mathbf{k})\}$

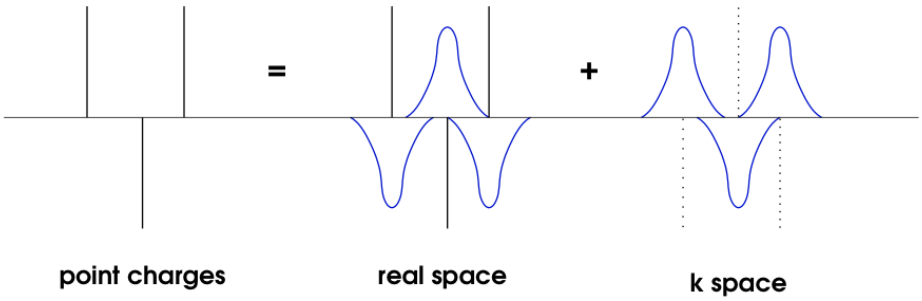
Assuming that we assign $m$ lattice points to each of $N$ charges, this results in a complexity of $Nm \log Nm = m(N \log N + N \log m)$ which has formal complexity $N \log N$ with a possibly large pre-factor $m$.

# 7 Ewald summation

Aim: compute electrostatic interactions in a periodic box. For long-ranged potentials such as electrostatics, direct particle-particle interactions cannot be cut off.

Idea of Ewald summation: Split charge density into a short-ranged and a long-ranged part

$$\rho_i(\mathbf{r}) = \underbrace{q_i\delta(\mathbf{r}-\mathbf{r}_i) - q_i G_\sigma(\mathbf{r}-\mathbf{r}_i)}_{\rho_i^S(\mathbf{r})} + \underbrace{q_i G_\sigma(\mathbf{r}-\mathbf{r}_i)}_{\rho_i^L(\mathbf{r})}$$




**Short range interaction energies**

We therefore compute the corresponding electrostatic energy directly between pairs of charges, and will later truncate this term at a certain cutoff radius:

$$E^S = \frac{1}{8\pi\epsilon_0} \sum_{\mathbf{n}} \sum_{i=1}^{N} \sum_{j=1}^{N} {}^* \frac{q_i q_j}{|\mathbf{r}_i - \mathbf{r}_j + \mathbf{n}L|} \mathrm{erfc}\left(\frac{|\mathbf{r}_i - \mathbf{r}_j + \mathbf{n}L|}{\sqrt{2}\sigma}\right) \tag{1}$$

Here the star $\sum^*$ indicates that the term $i = j$ is omitted in the primary cell in order to avoid interaction of the charge with itself.

**Long range interaction energy**

$$E^L = \frac{1}{2V\epsilon_0} \sum_{\mathbf{k}} |S(\mathbf{k})|^2 \frac{\mathrm{e}^{-\sigma^2 k^2/2}}{k^2}$$

with

$$S(\mathbf{k}) = \sum_{j=1}^{N} q_j \mathrm{e}^{-i\langle \mathbf{k}, \mathbf{r}_j \rangle}$$

**Long range self energy**

$$E^{\text{self}} = \frac{1}{2\epsilon_0 \sigma (2\pi)^{3/2}} \sum_{i=1}^{N} q_i^2$$

**Total energy**

$$E = E^S + E^L - E^{\text{self}}$$

**Numerical error and computational complexity**

Set the cutoff error in real space and in $k$-space to $\mathrm{e}^{-p}$. Fix the particle density $N/V$, the accuracy requirement $p$, and then minimize the execution time with respect to $R$, we find that:

$$R_{opt} = \left(\frac{p}{\pi}\right)^{1/2} \left(\frac{t_k}{t_r}\right)^{1/6} \frac{L}{N^{1/6}}$$

where $L$ is the box length, and $t_k$, $t_r$ are the times needed for one direct space interaction and for one $\mathbf{k}$ vector computation, respectively. The corresponding optimal time is:

$$T_{opt} = \frac{4\pi}{3} \left(\frac{p}{\pi}\right)^{3/2} \sqrt{t_r t_k} N^{3/2}.$$

**Particle-mesh Ewald**

1. Map the charges on a periodic lattice.
2. Solve the Fourier transform on the periodic lattice using FFT.
3. Map the potential back to the charges.

The computational complexity of PME is $N \log N$ (see Poisson solution, above)