Molecular Dynamics Simulations for Lennard-Jones gas.

Laura Bukonte - laura.bukonte@helsinki.fi

Introduction

Molecular dynamics is a simulation technique, which describes atomic trajectories by numerically solving Newton's equations of motion

$$\ddot{\mathbf{r}}_i = \mathbf{f}_i/m_i, \;\; i = 1, \ldots, N \tag{1}$$

for all atoms i in the system in a step-by-step fashion. $\mathbf{r}_i$ is the position of the i:th atom, m_i is its mass and $\mathbf{f}_i$ the force acting upon it. The forces between the atoms are described by the potential energy

$$\mathbf{f}_i = -\nabla_i U\left(\{\mathbf{r}_n\}\right), \tag{2}$$

where $U(\{\mathbf{r}_n\})$ contains the potential energy of all of the interactions of the system.

When the equations of motion (equation (1)) are solved, the total energy of the system

$$E = E_{\text{pot}} + E_{\text{kin}} \tag{3}$$

$$E_{\text{pot}} = \sum_{i=1}^{N} \sum_{j<i} U\left(\mathbf{r}_{ij}\right) \tag{4}$$

$$E_{\text{kin}} = \frac{1}{2} \sum_{i} m_i v_i^2 \tag{5}$$

is conserved. If the total number of atoms N and the volume V are also kept constant, then the MD simulations are said to be performed in the microcanonical (NVE) ensemble.

In practice, the system is often in a heat bath. The temperature is kept constant (instead of the energy) by using a thermostat, which simulates heat exchange between the system and the heat bath. This corresponds to the canonical (NVT) ensemble, in which the number of particles, volume, and temperature are fixed.

Methods

Initialization

Initial coordinates

The simulation is initialized by distributing N atoms randomly in the system – a box of length L in each direction centered at the origin.

However, as this may lead to atoms being unphysically close to each other, molecular dynamics simulations are always initialized by minimizing the energy of the starting configuration.

You will need to start by writing a minimization algorithm. As forces are easily available for the potentials in use, you can use, *e.g.*, the conjugate gradient or the Broyden-Fletcher-Goldfarb-Shanno (BFGS) algorithm, which are available in the GNU Scientific Library, see `http://www.gnu.org/software/gsl/manual/html_node/Multimin-Examples.html`.

Initial velocities

The atoms with are given an initial velocity at the start of the simulation according to Maxwell-Boltzmann statistics, following a Gaussian distribution. To generate the initial velocity for an atom in the NVT ensemble

1. generate three random numbers from a Gaussian distribution with zero mean $\mu = 0$ and unit variance $\sigma = 1$
2. multiply the numbers by $\sqrt{k_B T/m_i}$ to get the proper distribution of velocities
3. use the random numbers as the x, y and z components of the velocity.

To generate Gaussian distributed random numbers you can use, e.g., the Box-Muller algorithm. Generate two random numbers u_1 and u_2 uniformly distributed on $[0, 1)$ and calculate

$$\begin{aligned}\phi =& 2\pi u_1 \\ r =& \sqrt{-2 \log u_2}\end{aligned}$$

and then obtain two Gaussianly distributed random numbers g_1 and g_2 as

$$\begin{aligned}g_1 =& r \cos \phi \\ g_2 =& r \sin \phi\end{aligned}$$

Finally, set the center-of-mass velocity

$$\mathbf{V} = \frac{\sum_i m_i \mathbf{v}_i}{\sum_j m_j} \tag{6}$$

to zero by substracting the residual

$$\mathbf{v}_i \to \mathbf{v}_i - \frac{m_i}{\sum_j m_j} \mathbf{V}.$$

For the NVE case once you have generated the positions for the atoms, you can compute the necessary amount of kinetic energy to give the atoms as $E_{\text{kin}} = E - E_{\text{pot}}$. You can then distribute this energy in a random fashion to the atoms, e.g. using the following algorithm:

1. Generate a random number u_i uniformly distributed in $[0, 1)$ for every atom $i = 1..N$.

2. Calculate the sum of the random numbers and normalize u_i with

$$u_i \to \frac{u_i}{\sum_j u_j}$$

3. The i:th atom is now given a u_i:th part of the total kinetic energy, so its speed will be

$$v = \sqrt{\frac{2 u_i E_{\text{kin}}}{m}}.$$

4. Generate three random numbers v_i for the atom, distributed uniformly between $[-1, 1)$[1], and normalize them as

$$v_i \to \frac{v_i}{\sqrt{\sum_j v_j^2}}$$

after which the velocity for the i:th atom can be set as $\mathbf{v} = (v_1, v_2, v_3)\, v$.

Solution of the equation of motion

The equations of motion are solved by using numerical mathods. In this project, you will implement the velocity Verlet algorithm to integrate the equations of motions for all the particles in the system, computing positions, velocities $\mathbf{v}_i(t)$ and accelerations $\mathbf{a}_i(t)$ at each time step Δt:

$$\begin{aligned}\mathbf{r}_i\left(t + \Delta t\right) =& \mathbf{r}_i\left(t\right) + \mathbf{v}_i\left(t\right) \Delta t + \frac{1}{2} \mathbf{a}_i\left(t\right) \Delta t^2 \\ \mathbf{v}_i\left(t + \Delta t\right) =& \mathbf{v}_i\left(t\right) + \frac{\mathbf{a}_i\left(t\right) + \mathbf{a}_i\left(t + \Delta t\right)}{2} \Delta t\end{aligned}$$

Positions, velocities and accelerations are denoted by $\mathbf{r}_i(t)$, $\mathbf{v}_i(t)$ and $\mathbf{a}_i(t)$, respectively. The (instantaneous) acceleration at each time step is given by (1) as $\mathbf{a}_i(t) = \mathbf{f}_i(t)/m_i$, and the force $\mathbf{f}_i$ is calculated from the interaction potential, using (2).

Periodic boundary conditions are applied in the simulation. For example, if a particle passes through the wall at $x = L/2$, it emerges at $x = -L/2$. Also, the periodic boundary conditions affect the calculations of distances: the distance of two atoms

$$\mathbf{r}_{ij} = \mathbf{r}_i - \mathbf{r}_j$$

is limited by

$$\begin{aligned}-\frac{L}{2} \leq x_{ij} <& \frac{L}{2}, \\ -\frac{L}{2} \leq y_{ij} <& \frac{L}{2}, \\ -\frac{L}{2} \leq z_{ij} <& \frac{L}{2}.\end{aligned}$$

[1] $v_i = 2w_i - 1$, if w_i is uniformly distributed in $[0, 1)$.

Thermostat

To be able to simulate the NVT ensemble, implement the Berendsen thermostat. The thermostat rescales all the velocities $\mathbf{v}_i$ in the system at every time step with a factor

$$\lambda\left(t\right) = \sqrt{1 + \frac{\Delta t}{\tau_T}\left(\frac{T_0}{T\left(t\right)} - 1\right)}$$

where Δt is the time step of the simulation, τ_T is the relaxation time, T_0 is the reference temperature and $T\left(t\right)$ the current temperature of the system, which is given by

$$T\left(t\right) = \frac{2E_{\text{kin}}}{3N},$$

where the temperature is measured in eV.

Hint:

$$u \cdot \frac{\text{Å}^2}{\text{fs}^2} = 103.6427 \text{ eV}$$

To avoid the "flying ice cube" phenomenon, the center-of-mass velocity is removed at every time step. For a similar reason ("rotating ice cube effect") the total angular momentum with respect to the origin

$$\mathbf{L} = \sum_i m_i \mathbf{r}_i \times \mathbf{v}_i$$

is also removed at every time step.

Problem

Write a molecular dynamics code to simulate argon in the liquid and the gas phase. The argon atoms interact with each other through the Lennard-Jones potential

$$U(\mathbf{r}_{ij}) = 4\epsilon\left[\left(\frac{\sigma}{r_{ij}}\right)^{12} - \left(\frac{\sigma}{r_{ij}}\right)^{6}\right] \qquad (7)$$

where the parameters for Ar are $\varepsilon = 0.0104$ eV, $\sigma = 3.40$ Å. The mass of one argon atom is $m = 39.948u$, where u is the atomic mass unit.

Since computers only crunch numbers, you have to fix a (dimensionless) unit system you use in the program. International standard units are impractical for this purpose, so your program should use eVs as the unit of energy and temperature (so Boltzmann's constant is $k_B = 1$), and Å for the unit of distance. Time is measured in femtoseconds, and the time step used is $\Delta t = 0.5$.

Hint: since the energy is in eV as a function of Ångströms, the force will be in units of eV/Å. You want this in Å/fs^2, which will give a proportionality constant of

$$\begin{aligned}(\mathbf{a})\frac{\text{Å}}{\text{fs}^2} &= \frac{(\mathbf{f})\frac{\text{eV}}{\text{Å}}}{(m)\,u} \\ (\mathbf{a}) &= \frac{\frac{\text{eV}}{\text{Å}}}{u\frac{\text{Å}}{\text{fs}^2}}\frac{(\mathbf{f})}{(m)} \\ &= 9.648537 \cdot 10^{-3}\frac{(\mathbf{f})}{(m)}\end{aligned}$$

where $(\mathbf{a})$, $(\mathbf{f})$ and (m) are now dimensionless. Similarly you can calculate for the kinetic energy

$$\begin{aligned}(T)\,\text{eV} &= \frac{1}{2}(m)(v)^2\frac{u\cdot\text{Å}^2}{\text{fs}^2} \\ (T) &= \frac{1}{2}(m)(v)^2\cdot\frac{u\cdot\text{Å}^2}{\text{fs}^2\text{eV}} \\ &= 103.6427\cdot\frac{1}{2}(m)(v)^2\end{aligned}$$

NVE

First, check that a suitable time step is used by performing runs with a few values of Δt and observing the evolution of the energy. When the time step is too large, the total energy will grow exponentially. How big a time step can you use before this happens?

The program should read in the number of particles, the volume of the simulation cell and the value of the total energy of the system.

NVT

Implement Berendsen temperature control, and perform a simulation of the liquid (80K – 6.94 meV) and the gas (300K – 25.9 meV) system. The density of the gas is $1.784\frac{\text{g}}{\text{l}}$, so you should use a box size $L = 33.38\sqrt[3]{N}$ Å. The liquid has a much higher density of $1.40\frac{\text{g}}{\text{cm}^3}$, so the box size you should use is

$L = 3.62\sqrt[3]{N}$ Å. You should use $O\left(10^3\right)$ atoms in the simulation to get a good result.

Analysis of the runs

Compute the pair correlation function of the particles $g(r)$ for the NVT runs. $g\left(r\right)$ describes the distance distribution of the particles in the system

$$g\left(r\right) = \frac{2L^3}{N^2}\left\langle \sum_{i=1}^{N}\sum_{j<i}\delta\left(r - \left|\mathbf{r}_i - \mathbf{r}_j\right|\right)\right\rangle \quad (8)$$

Note that $g(r) \to 1$ when $r \to \infty$.

In practice the pair correlation function is calculated on a binned grid. Whenever two particles are found at the distance r of each other, the bin corresponding to the distance r is incremented by one. The correlation function is summed over all the atoms in the system and also averaged over a large number of time steps.

Visualize the evolution of the temperature and energy over time. Since the initial configuration is a bad approximation for the equilibrium system, the system will thermalize at the start of the run. How much time do you have to allow for the thermalization?

Finally, compute the pair correlation functions of the liquid and of the gas, using only the thermalized parts of the trajectories, and plot them in the same figure.