Exercises NMSM 2024-2025

Nunziato Damino
matr. 2097564

January 18, 2025

Exercise 8: Off-lattice Monte Carlo simulations

Pen & Paper - Reduced units

In order to convert from reduced units to SI units, specifically temperature and time, we simply do the conversion:

$$T^* = \frac{k_B T}{\varepsilon} \iff T = \frac{\varepsilon T^*}{k_B} \qquad t^* = \frac{t}{\tau} \iff t = \tau t^* \quad , \tau = \sqrt{\frac{\bar{m}\sigma^2}{\varepsilon}} = \sigma\sqrt{\frac{\bar{m}}{k_B}\frac{k_B}{\varepsilon}}$$

Where σ, $\bar{m}$, ε are respectively the reference unit for length, mass and energy for the system considered.

If we consider a reduced temperature $T^* = 2$, then for Argon and Krypton :

$$T_{\mathrm{Ar}} = 239.6 \text{ K} \qquad T_{\mathrm{Kr}} = 328.0 \text{ K}$$

Moreover, if we consider a timestep of $\Delta t = 0.001\tau$, we have respectively for Argon and Krypton:

$$\Delta t_{\mathrm{Ar}} = 2.160 \text{ fs} \qquad \Delta t_{\mathrm{Kr}} = 2.650 \text{ fs}$$

Numerical exercise

In this exercise we study a Lennard-Jones fluid using a Monte Carlo approach.
The Monte Carlo simulation we want to implement is simple: we want to perform a Monte Carlo sweep consisting of N local moves, where $N = 500$ is the number of particles in the system, where each local move consist in a uniform displacement of a particle chosen at random.
Once a MC sweep is done, the new configuration is accepted or rejected using a Metropolis filter. A total of $T = 10000$ sweeps are performed and then the pressure is measured, discarding an appropriate amount of sweeps for equilibration.
The displacement in the local move is chosen such that the acceptance rate of said filter is $\sim 50\%$.

We begin by making some general considerations. Having set for our simulation the reference unit for length and energy repsctively $\sigma = 1$ and $\varepsilon = 1$, the reduced units coincides with the SI units.
One here has 2 choices in the code implementation: write a general code where each conversion between reduced and SI units is made, or to write a specific implementation for our system.
Both approaches have their merits : the first makes the code more mantainable and expandible, but slower, the second makes it more efficient, but problem-specific.
We choose for our implementation the second approach.

Of particular interest are some optimization made during the simulation :

- Given that we want to evaluate quantities in function of the number density ρ, we can either change the number of particle N or the simulation box volume V.
  While mathematically equivalent, given that most of the algorithms are dependent on N, is convenient to change V.

- In the implementation of the PBC one has to pay particular attention in not using the standard `round` implementation in Python, which implement the so called banker's round.
  One has instead to use the numpy function `np.round` (or even better `np.rint`) which implement the nearest integer round, the correct one and typical of C/Fortran.

- In the Metropolis filter if we pursue a naive approach, i.e. we evaluate the energy difference of the system between the final and the initial configuration as is, we found ourselves an algorithm of time complexity $\mathcal{O}(N^2)$, where N is the number of particle in the system.
  If we observe instead that in one local move the only quantity that changes is the potential contribution of the displaced particle, we can cut the time complexity to $\mathcal{O}(N)$.
  This also imply that the energy tail contribution in the context of the Metropolis filter is irrelevant.

We now show the result of the simulation for $T^* = 0.9$ and $T^* = 2$ in figures 1.1 and 1.2 respectively, in comparison with the supplied data.

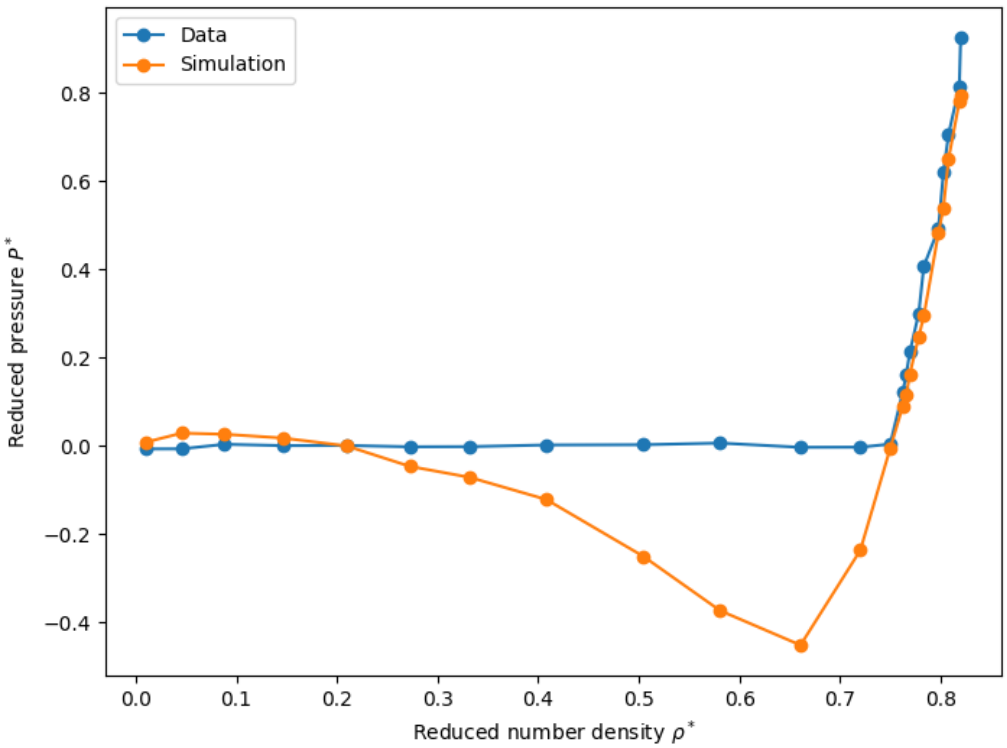

Figure 1.1: Comparison between the supplied data for the pressure-number density plot at $T^* = 0.9$

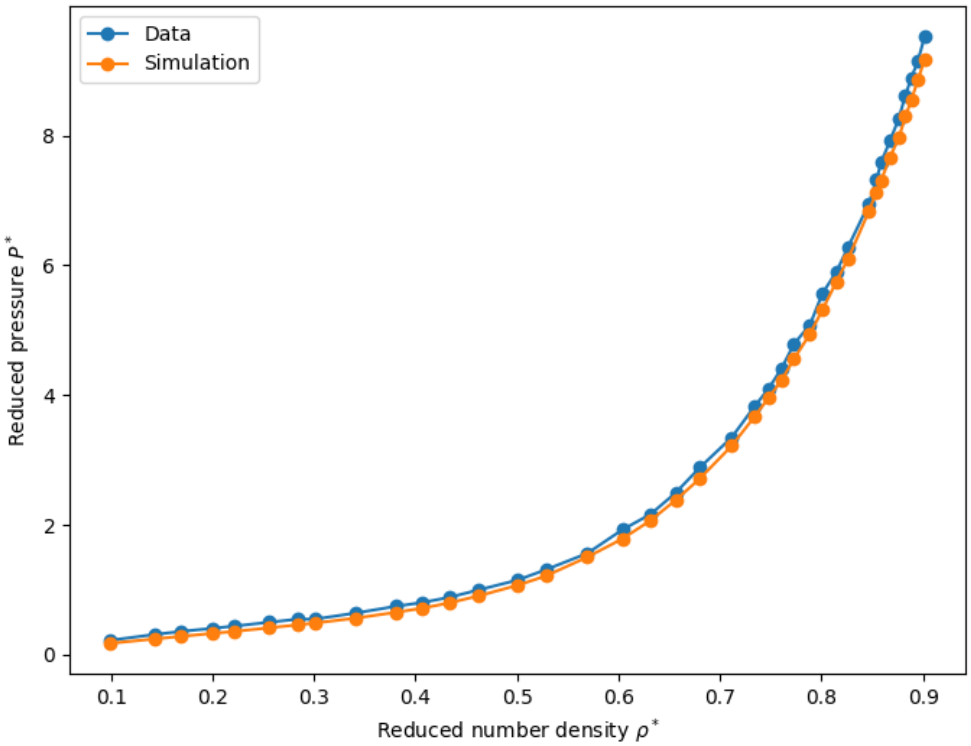

Figure 1.2: Comparison between the supplied data for the pressure-number density plot at $T^* = 2$

Is clear the good agreement for the simulation with the data for $T^* = 2$, above the critical temperature for a Lennard-Jones fluid, which is $T^* = 1.316$ [1], while it seems not good for $T^* = 0.9$.
This can be easily explained if we observe that in figure 1.1 the horizontal line is the saturated vapor pressure, and our simulation points indicate the densities of the coexisting vapor and liquid phases. Over a broad density range, the simulated system is observed to be metastable and even exhibit negative pressure. This phenomenon arises because, in finite systems, the formation of a liquid-vapor interface incurs a significant free-energy cost. For sufficiently small systems, this cost can be so substantial that phase separation becomes unfavorable [2]. Consequently, these issues are most pronounced in small systems and in scenarios where the interfacial free energy is particularly large. For this reason, standard NVT-simulations are not recommended for determining the vapor-liquid coexistence curve or for studying any strong first-order phase transitions in small systems.

Exercise 9: Integration schemes

The idea is to implement the velocity Verlet algorithm and the Gear predictor-correct algorithm of the 5th order specifically for the harmonic oscillator.
Before starting the quantitative analysis over these integration schemes, we start with a qualitative inspection against the analytical solution in figure 1.3.
Note that the integration timestep is $\Delta t = 0.001$ and $\omega = 0.1$.

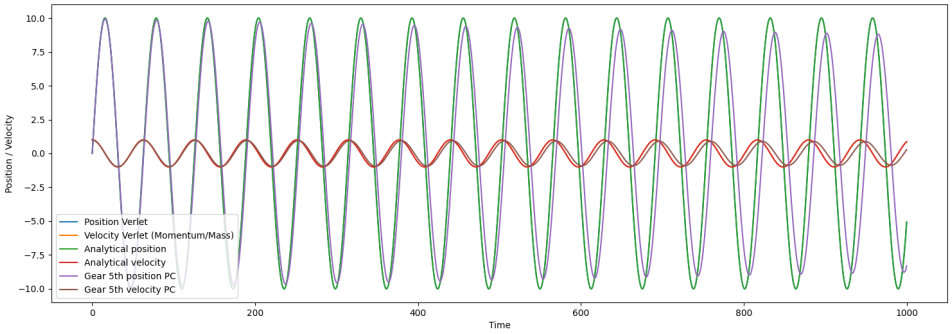

Figure 1.3: Comparison between the analytical solution for position and velocity in an harmonic oscillator with the velocit Verlet and the Gear 5th order predictor-corrector integration scheme.

Is clear that the Gear predictor-correct algorithm of the 5th order presents for long integration time an effect of amplitude reduction and delay, while the velocity Verlet algorithm do not presents these problems.
Now we study the energy conservation of the 2 algorithms. Again we plot the total energy of the system over time for inspection in figure 1.4 and for each algorithm we plot $\frac{E_{algo}(t)-E_0}{E_0}$ respectively in figure 1.5 for the velocity Verlet algorithm and the Gear predictor-correct algorithm of the 5th order.

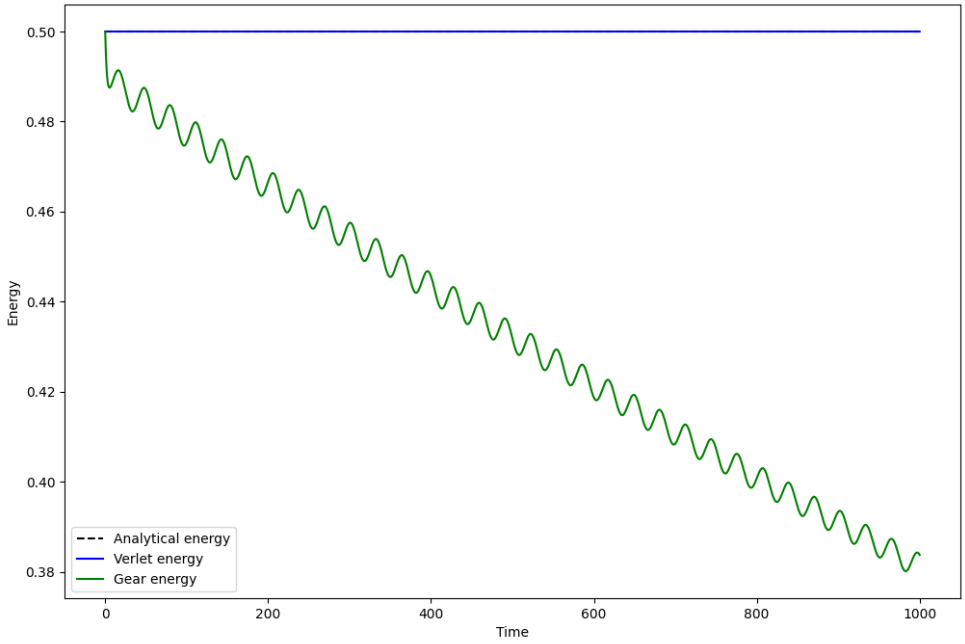

Figure 1.4: Comparison between the total energy in an harmonic oscillator and the energy evaluated with the velocity Verlet and the Gear 5th order predictor-corrector integration scheme.

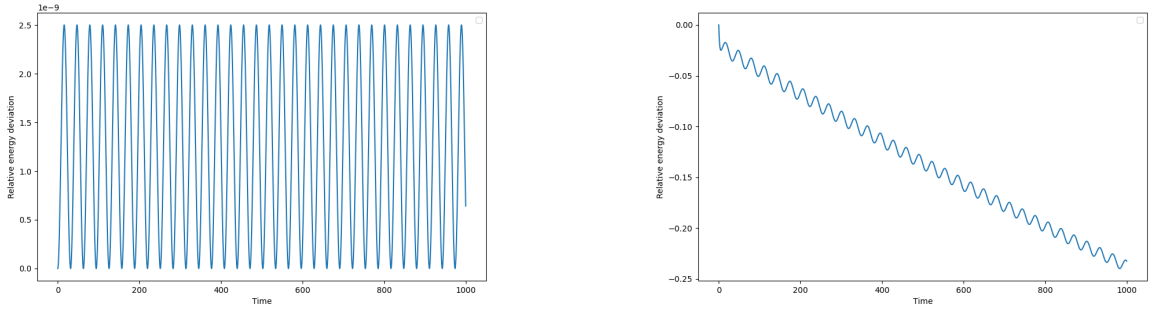

Figure 1.5: (Left) Relative energy deviation from the analytical value for the velocity Verlet integration scheme. (Right) Relative energy deviation from the analytical value for the Gear 5th order predictor-corrector integration scheme.

From this analysis we conclude, as we expected, that the velocity Verlet algorithm is symplectic while the Gear predictor-corrector is not.
Lastly we focus on the velocity Verlet to evaluate its stability for different values of ω. We choose for each iteration a timestep $\Delta t = 0.01\omega^{-1}$ and obtain figures 1.9.

We observe that the deviation is smaller for small ω.

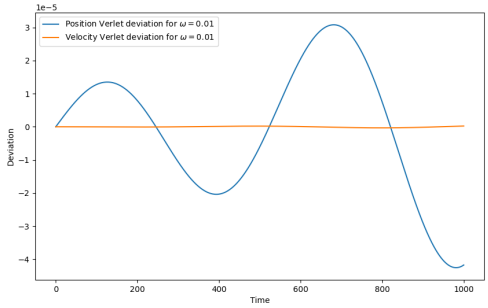

Figure 1.6: Position and velocity deviation for the velocity Verlet integration scheme for $\omega = 0.01$.

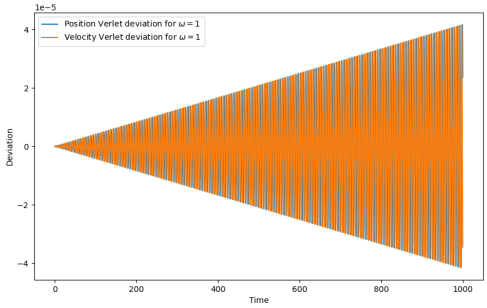

Figure 1.7: Position and velocity deviation for the velocity Verlet integration scheme for $\omega = 1$.

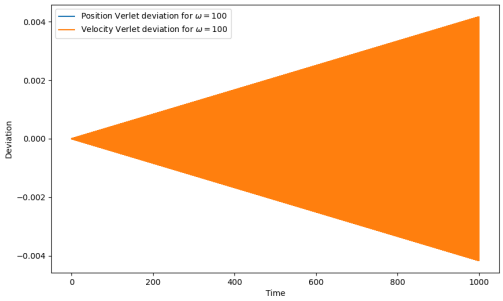

Figure 1.8: Position and velocity deviation for the velocity Verlet integration scheme for $\omega = 100$.

Figure 1.9: Comparison of position and velocity deviations for the velocity Verlet integration scheme at different ω values.

Exercise 10 : Interaction potentials & thermostats

Pen & Paper - Canonical fluctuactions

We want to prove that, in a canonical ensemble (NVT) :

$$\frac{\sigma_{T_K}^2}{\langle T_K \rangle^2} = \frac{\langle T_K^2 \rangle - \langle T_K \rangle^2}{\langle T_K \rangle^2}$$

Proof - From the equipartition theorem, in the NVT ensemble:

$$\langle T_K \rangle = \frac{2}{3Nk_B}\langle E_K \rangle \qquad \langle T_K^2 \rangle = \left(\frac{2}{3Nk_B}\right)^2 \langle E_K^2 \rangle$$

So that the temperature variance can be written as :

$$\sigma_{T_K}^2 = \langle T_K^2 \rangle - \langle T_K \rangle^2 = \left(\frac{2}{3Nk_B}\right)^2 (\langle E_K^2 \rangle - \langle E_K \rangle^2)$$

Moreover we have that the energy variance can be written in the NVT ensemble as:

$$\sigma_{E_K}^2 = \langle E_K^2 \rangle - \langle E_K \rangle^2 = k_B T^2 C_V$$

Also the heat capacity :

$$C_V = \frac{3}{2}k_B T$$

Putting all together:

$$\sigma_{T_K}^2 = \left(\frac{2}{3Nk_B}\right)^2 k_B T^2 \left(\frac{3}{2}k_B T\right) = \frac{2}{3N}T^2$$

Then one can evaluate the initial statement, in a NVT ensemble :

$$\frac{\sigma_{T_K}^2}{\langle T_K \rangle^2} = \frac{\frac{2}{3N}T^2}{T^2} = \frac{2}{3N}$$

Numerical exercise

The idea of the simulation is to simulate a LJ fluid in different ensembles. The simulation will be carried in LAMMPS.
For both ensembles we will consider a LJ system with reference units $\sigma = 1$, $\varepsilon = 1$ and $m = 1$ in a cubic simulation box of size $L = 10\sigma$ with periodic boundary conditions. The equation of motions will

be integrated using the velocity Verlet algorithm, as is the default option in LAMMPS. In each run some steps are discarded for equilibration.

NVE ensemble We begin by fixing the number density at $\rho = 0.2\sigma^{-3}$ and draw the initial velocities of the system from the Maxwell-Boltzmann distribution at $T^* = 1$, enforcing the total momentum to zero. We then study the system for different cutoffs, specifically we go from $r_c = 2^{1/6}\sigma$ to $r_c = 4\sigma$ with steps of $\Delta r_c = 0.2\sigma$.
For each run we discard 50000 steps for equilibration and consider 150000 steps for production.
We observe that at greater cutoff radii the total energy time series stabilize faster and with less variance around the mean value, which is higher for higher cutoff radii.
We plot at title of example the energy time series of the cutoff radius $r_c = 2^{1/6}\sigma$ and $r_c = 4\sigma$ after equilibration to show this in figure 1.10:

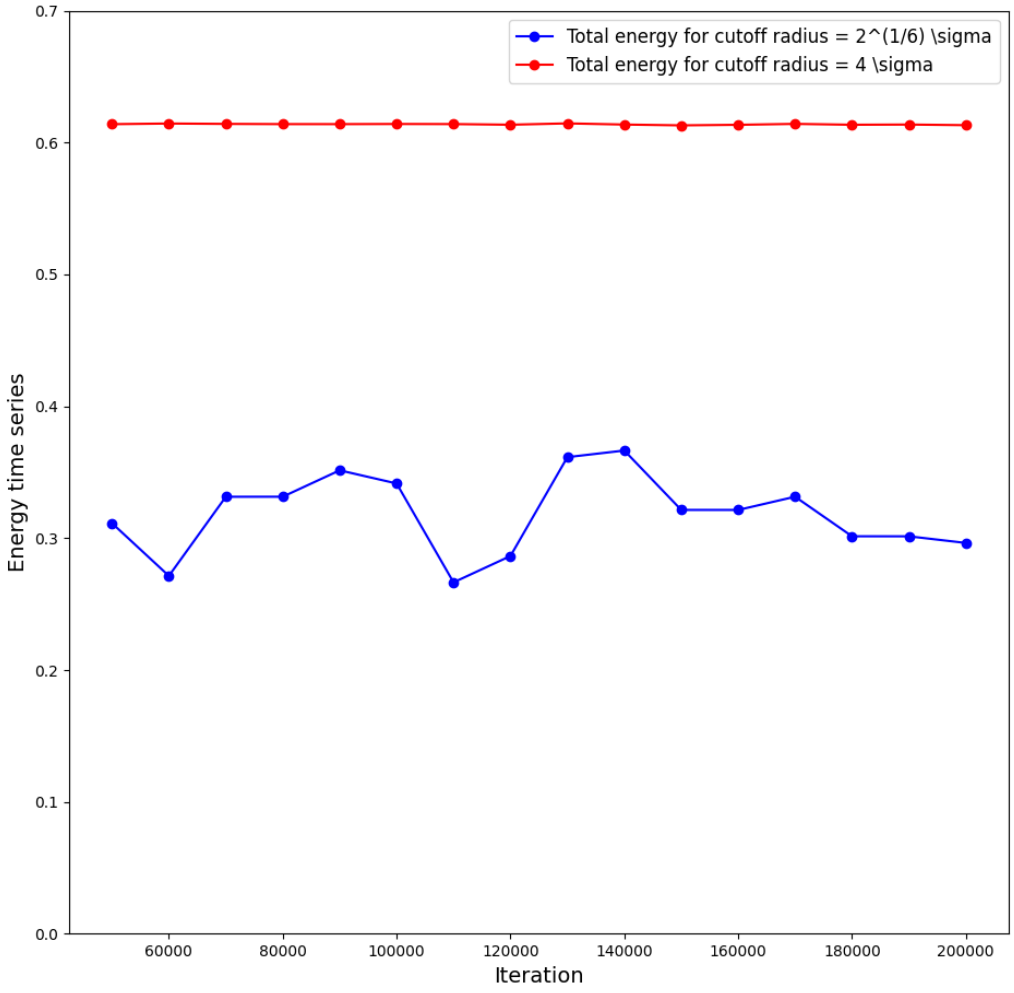

Figure 1.10: Energy time series comparison after equilibration for the cutoff radius $r_c = 2^{1/6}\sigma$ and $r_c = 4\sigma$

It seems then that one should always use higher cutoff radii, but the problem is that the higher the cutoff radius, the higher the computation time. We have for example for 200000 iteration for a

cutoff radius $r_c = 4\sigma$ a computation time $t_{computation} \sim 8\ s$ while for $r_c = 2^{1/6}\sigma$ a computation time $t_{computation} \sim 1\ s$.
Of course the general idea is to find a balance in the choice of a cutoff radius that a guarantees proper stationarity for the observable time series we want to investigate but also has a decent computation time.
The rest of the data is not reported, but can be found in the attached log.lammps file.
In order to evaluate better the difference between the different cutoff radii, we plot also the radial distribution function for the different cutoffs in figure 1.11.

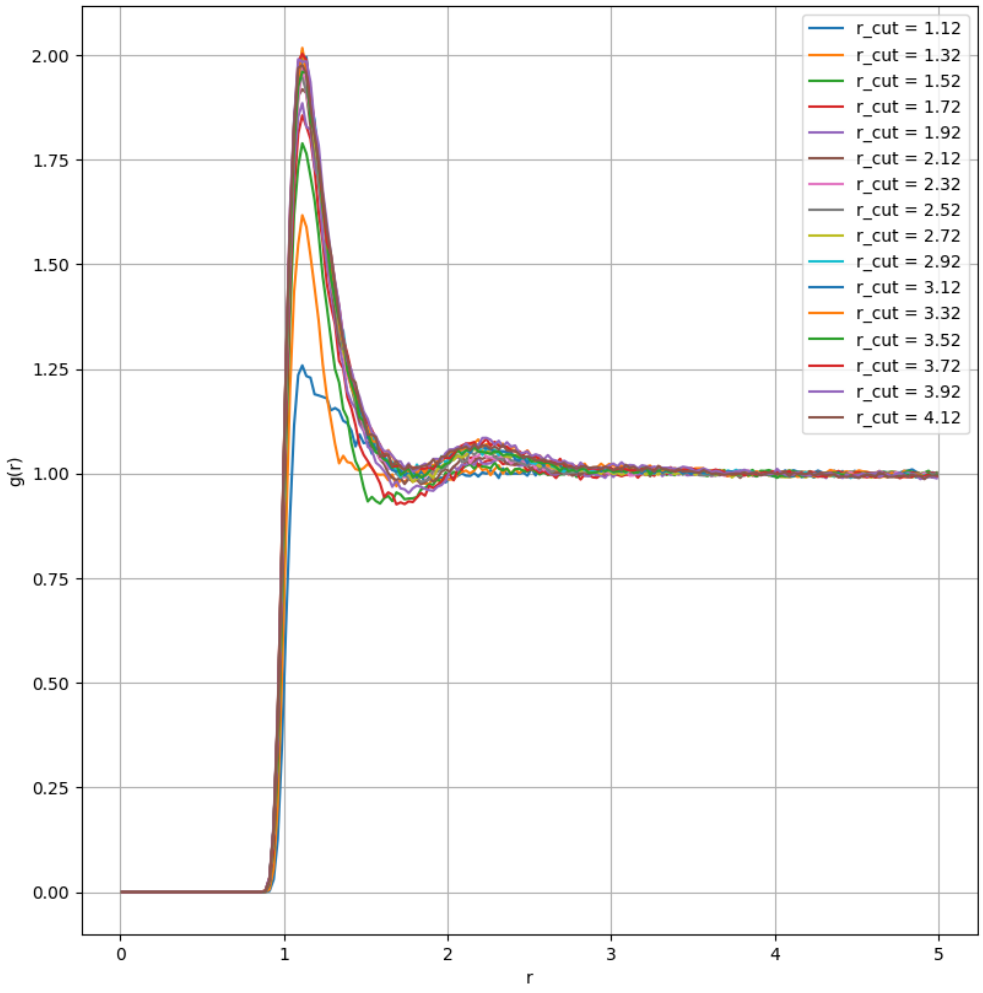

Figure 1.11: $g(r)$ for all simulated r_{cut} values

As we can see for all the cutoff radii the $g(r)$ reveal that the system is in a liquid state, but for higher cutoff this determination is clearer.

NVT ensemble In order to simulate the same LJ system in an NVT ensemble we need to introduce a thermostat in the simulation.
We do 2 separate runs: the first with velocity rescaling (non-canonical) and the second with the Nose-Hoover thermostat (canonical). For both runs we simulate the system at $T^* = 2$ for different decreasing number density values $\rho < 0.2$.
Given that LAMMPS outputs values per particle we should find in each simulation $\langle E_K \rangle = \frac{3}{2}T^* = 3$, while for the fluctuactions

$$\frac{\sigma^2_{T_K}}{\langle T_K \rangle^2} = \frac{2}{3}$$

. For a simulation with 50000 steps for equilibration and 450000 for production we found for the 2 runs at different density values, the following mean kinetic energy and mean temperature value, shown in table 1.1.

ρ^*	$\langle E_K \rangle$	$\langle T_K \rangle$
Non-Canonical		
0.075	2.94	1.98
0.100	2.95	1.98
0.125	2.95	1.98
0.150	2.96	1.98
0.175	2.96	1.98
Canonical		
0.075	2.49	1.68
0.100	2.90	1.96
0.125	3.04	2.05
0.150	3.16	2.11
0.175	2.81	1.88

Table 1.1: Table showing values for the mean temperature and kinetic energy for different values of number density for the non-canonical and canonical thermostat

In both thermostat the mean kinetic energy is consistent with the expected value.
In the non-canonical iteration of the NVT ensemble we observe that there are no fluctuactions for the mean temperature, so that is not consistent with the prediction that we expect for a canonical ensemble, while in the canonical iteration we see consistent fluctuactions.
As before all additional data can be found in the relative log.lammps file.

Exercise 11 : Langevin and Brownian dynamics

Pen & Paper - Brownian time and length scales

We begin by considering the Langevin equation in the form:

$$m\ddot{\vec{x}} + m\gamma\dot{\vec{x}} + \sqrt{6m\gamma k_B T}\bar{\xi}(t) = 0$$

In the particular case in which the particle is considered spherical, with a diameter $\sigma = 10^{-8}\ m$, has roughly the same density of water and diffuse in water, we have in the limit of low Reynolds number that $m\gamma = 3\pi\eta\sigma$, where η is the water viscosity.
With the parameters considered we have:

$$m = \frac{4}{3}\pi\rho\frac{\sigma^3}{8} \sim 10^{-22}\ kg \qquad m\gamma = 3\pi\eta\sigma \sim 10^{-10}\ kg \cdot s^{-1}$$

So in this case we can neglect the inertial force and write:

$$m\gamma\dot{\vec{x}} + \sqrt{6m\gamma k_B T}\bar{\xi}(t) = 0$$

Implying that we are in the diffusion regime. We can then simply estimate the time to diffuse in water over a distance equal to the particle diameter by :

$$\sigma^2 = 6Dt = 6\frac{k_B T}{3\pi\eta\sigma}t \qquad \Longleftrightarrow \qquad t \sim 4 \cdot 10^{-7}\ s$$

Numerical exercise

We consider a gas of $N = 1000$ non interacting Brownian particle in a simulation for the reference units are set as $\sigma = 1$, $\varepsilon = 1$ and $m = 1$. The simulation box is a cubic box of side $L = 20\sigma$, where periodic boundary conditions are considered. We start by setting all the particles in the origin of our simulation box.
In order to study the diffusity properties of the system we decided to study it in unwrapped coordinates.

Brownian motion in the bulk In absence of any potential we study the system in the overdamped limit, by implementing for its evolution a first order integrator such as the Euler-Maruyama integrator, which algorithm can be found in the relative section.
We study the system mean square displacement for varying temperature values $0.1 < T^* < 2$ while $\gamma^* = 1$, and obtain figure 1.12.

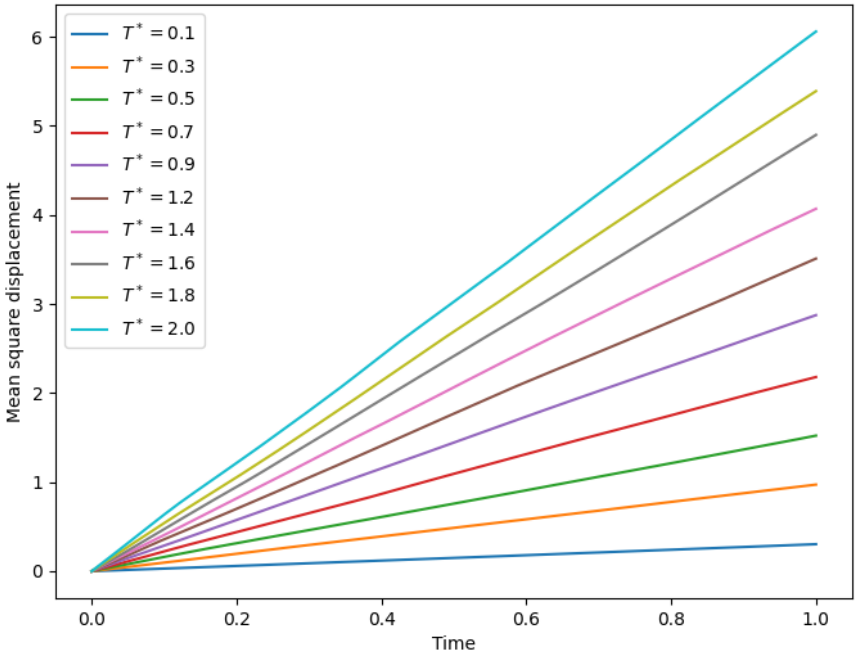

Figure 1.12: Mean square displacement for $0.1 < T^* < 2$ while $\gamma^* = 1$

All MSD are proportional to time as expected and clearly higher the temperature, higher the kinetic energy and consequently the diffusivity.
Suppose instead we fix $T^* = 1$ and vary the friction coefficient $10 < \gamma^* < 100$. Then we obtain 1.13

Then we obtain the converse, higher the friction coefficient, lower the diffusivity. Finally we can plot the distribution of the N particles along the x axis for $T^* = 1$ and $\gamma^* = 1$. We obtain 1.14 :

As expected, due to a non zero diffusivity, particles spreads over time.

Brownian motion in an harmonic trap Suppose now we set an harmonic trap in the simulation box origin. Then if we plot the MSD by varying the harmonic constant $0.1 < K^* < 10$ by letting the other parameter be unitary we obtain figure 1.15:

As we can see the higher the harmonic constant is higher the more pronounced the caging effect around the harmonic trap is. This can be seen also by the x position distribution in presence of $K^* = 10$ in figure 1.16

In this case the harmonic trap prevents the natural diffusivity seen in the bulk case, caging the particles.

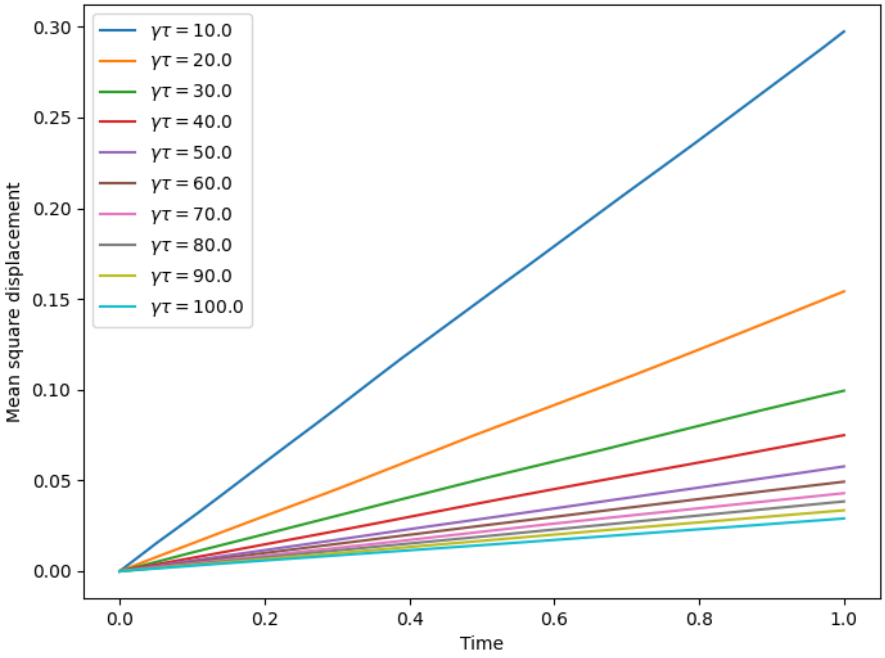

Figure 1.13: Mean square displacement for $10 < \gamma^* < 100$ while $T^* = 1$

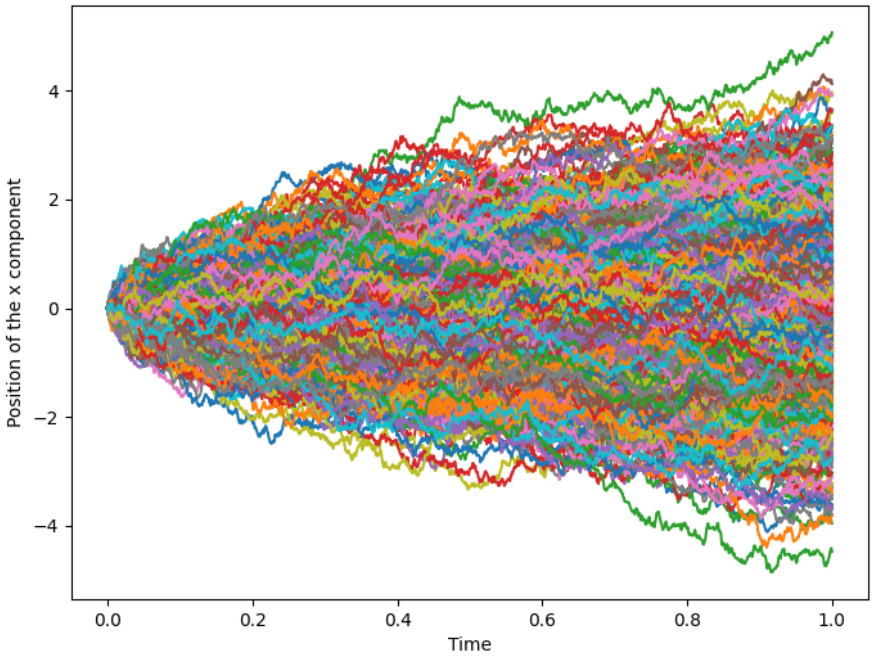

Figure 1.14: Distribution of the x position for 1000 Brownian particles

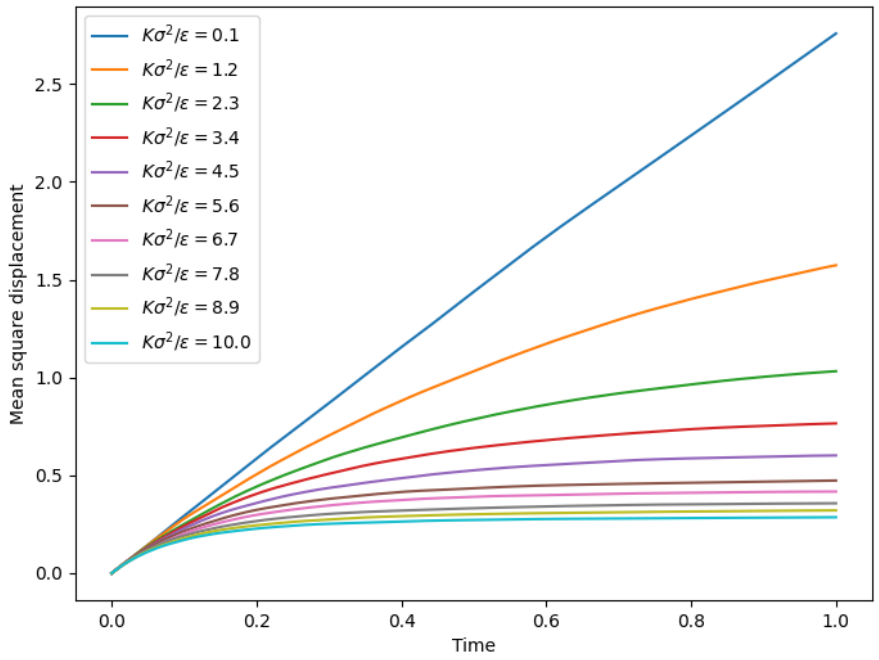

Figure 1.15: Mean square displacement for $0.1 < K^* < 10$ while $T^* = 1$, $\gamma^* = 1$

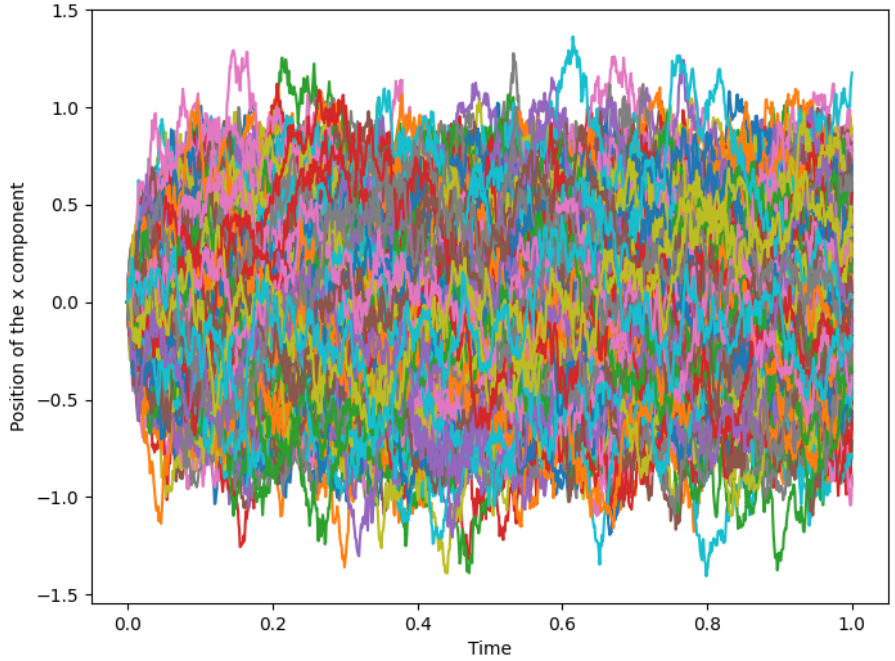

Figure 1.16: Distribution of the x position for 1000 Brownian particles with $K^* = 10$

Exercise 12 : The multiple histogram method

We are going to use in this exercise the multiple histogram method (MHM) to evaluate the energy variance per degree of freedom:

$$C_N(\beta) = \frac{1}{N}(\langle E^2 \rangle - \langle E \rangle^2)$$

For the grafted polymer model saw in chapter 6, where N is the number of monomers.
Then we will find the maximum value of the inverse β for which the maximum $C_N(\beta)$ is achieved, for different values of N.
Finally these value are plotted against N^ϕ where $\phi = 1/2$, to extrapolate the infinite size limit.

Before showing the results we comment briefly on the implementation of the MHM. In general in the MHM one has to put particular attention in evaluating the partition function, specifically in order to avoid over/under-flow due to the floating point structure, and specifically some special implementations are required, such as the log-sum-exp trick.
But due to the low number of degrees of freedom in our system we proceeded to a naive implementation of the algorithm, given that we are well within the floating point boundary.
Notice that anyway our algorithm is not scalable for large system, and a proper implementation is necessary in that case.
Also, to reduce our computation time, we raised the threshold to 10^{-3}.
We present our results in figure 1.17.

As is clear, the evaluation is wrong, as it presents a negative temperature. This is due to the fact that the points are evaluated without error due to time constraint, so the whole procedure of best fit fails.

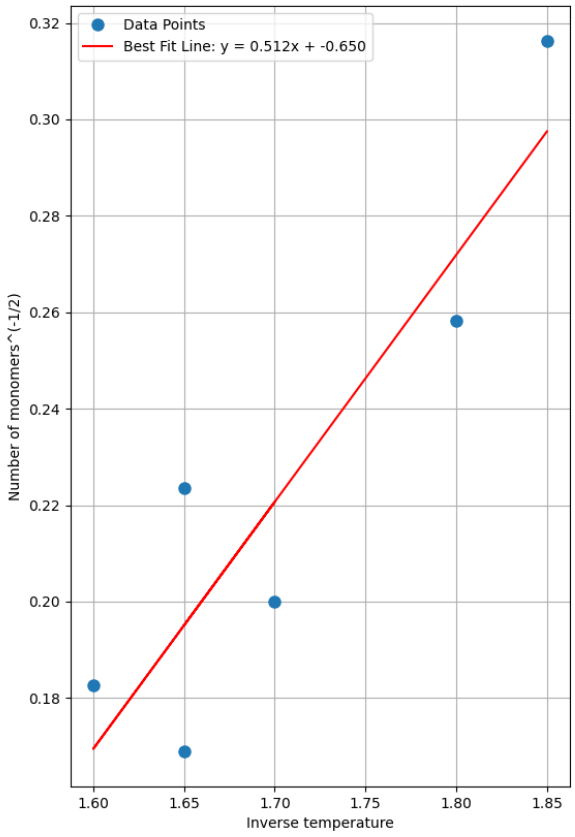

Figure 1.17: Infinite size limit determination

Exercise 13 : Exercise on the cell list

We want to simulate in a square box of side $B = 100$, N Brownian particles with reflective boundary condition, with a mutual repulsive potential.
Half of the particle will have radius $R_1 = 1.25$ and the other half $R_2 = 1$ in order to prevent the system crystallization at $T^* = 1$.
The initial positions will be initialized randomly inside the box and a non-compenetration check will be enforced through the evolution, that happens through the Euler-Maruyama algorithm. The repulsive potential $U(r_{ij}) = \varepsilon \exp\left(-r_{ij}^2/2\sigma_{ij}\right)$ depends on the reference length $\sigma_{ij} = R_i + R_j$. We will cutoff the resulting force at $r_{cut} = 4\sigma_{ij}$. Moreover we set $k_B = 1$ and the diffusity through the mobility parameter $D_i = \mu_i T$ that depends on the particle radius $\mu_i = 1/R_i$. In order to achieve a large N in our simulation, we will use the linked list cell method, in order to reduce the force calculation routine time complexity from $\mathcal{O}(N^2)$ to $\mathcal{O}(N)$. Given that $\max(r_{cut}) = 8R_1$ we choose this as side of our cell in the algorithm, so that is guaranteed that all force calculation are within the cutoff radius.
Now that we are set, in order to study the diffusivity we plot the mean square displacement along the x position of the particle of type 1 and 2, and of all the particles, for different values of $\rho = N/B^2$.

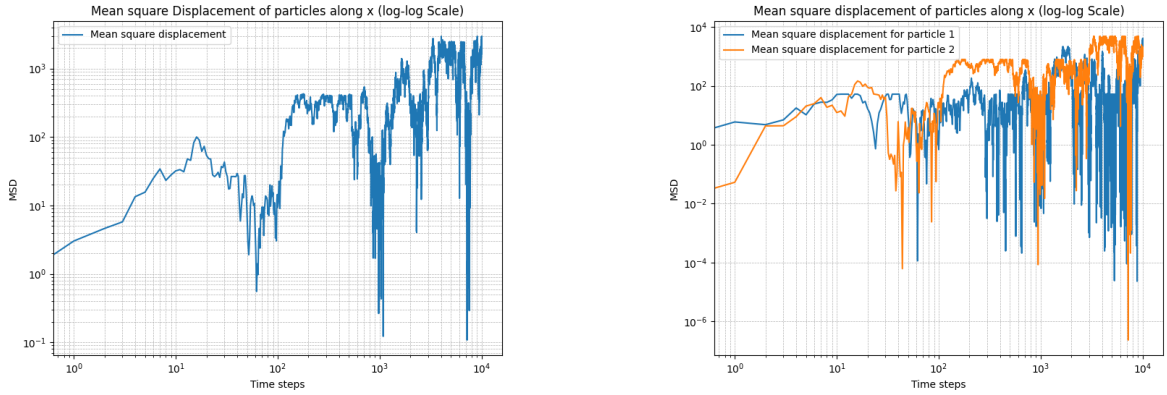

Figure 1.18: (Left) MSD for $\rho = 2 \cdot 10^{-4}$. (Right) MSD for particles of type 1 and 2.

We observe that the MSD follows $2D_i t$ up to a saturation due to the square confinement. For high ρ we don't observe a caging in the MSD for lower times, but we observe it in the computation time, as the system makes at roughly half timestep begins to make many attempts ($\sim 10^6 - 10^7$) to try to pass to non-compenetration check.
In general we can see that the MSD for the 2 particles are roughly the same, while we can see that for particle 1 we have a slightly lower diffusivity.

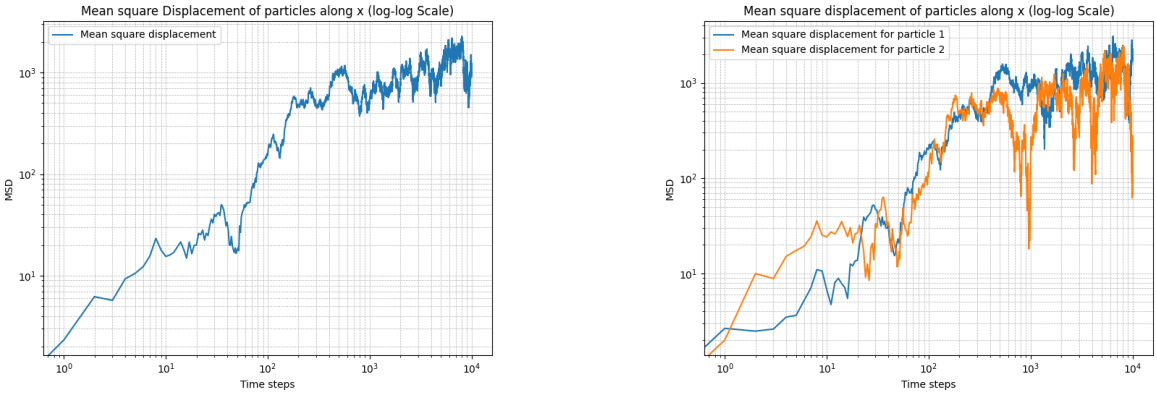

Figure 1.19: (Left) MSD for $\rho = 1 \cdot 10^{-3}$. (Right) MSD for particles of type 1 and 2

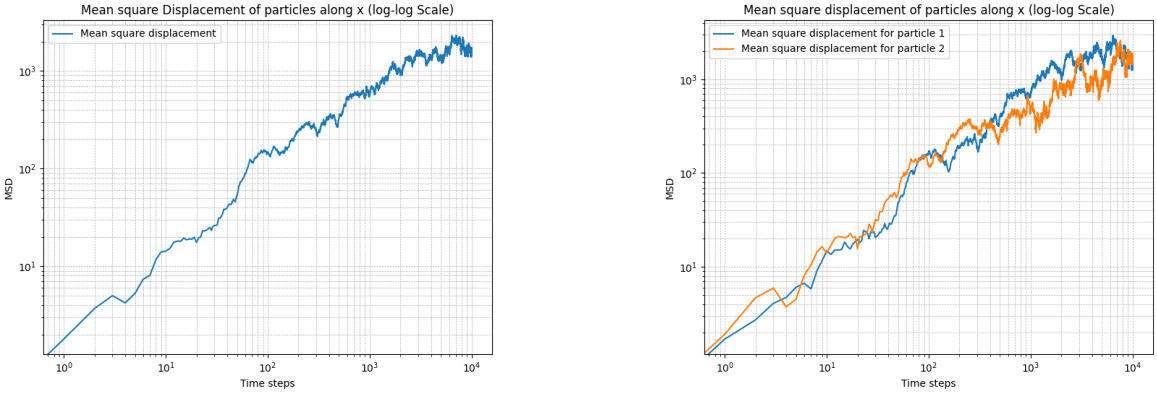

Figure 1.20: (Left) MSD for $\rho = 5 \cdot 10^{-3}$. (Right) MSD for particles of type 1 and 2.

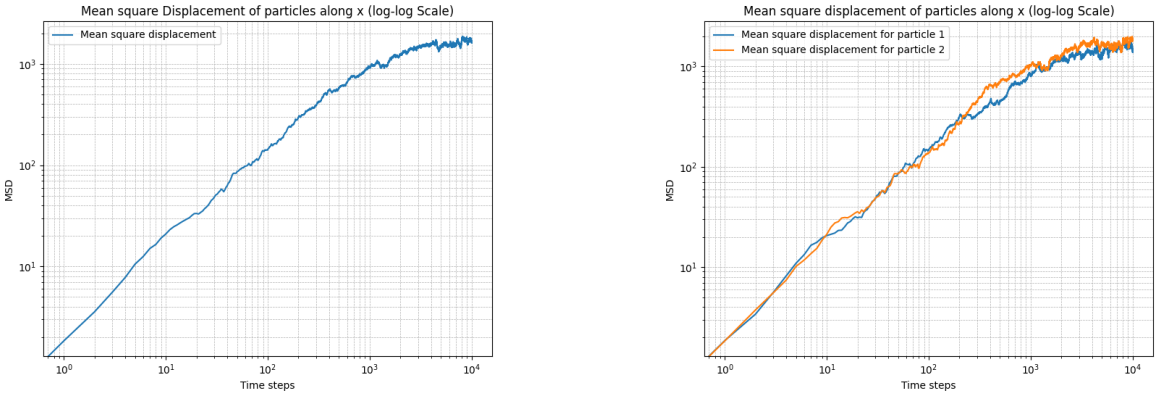

Figure 1.21: (Left) MSD for $\rho = 0.02$. (Right) MSD for particles of type 1 and 2.

Bibliography

[1] J. K. Johnson, J. A. Zollweg, and K. E. Gubbins. The lennard-jones equation of state revisited. *Molecular Physics*, 78(3):591–618, 1993.

[2] Z. W. Salsburg and W. W. Wood. Equation of state of classical hard spheres at high density. *Journal of Chemical Physics*, 37(4):798–804, 1962.