Assignment 5: Metropolis Monte Carlo

Marta Casado Carrasquer

March 5, 2025

1 Introduction

Molecular Dynamics (MD) and Monte Carlo (MC) simulations are two computational techniques widely used in the study of various systems. While both methods are used to explore the properties of the systems at molecular level, they differ in their approaches.

MD simulations are deterministic methods based on solving Newton's equations of motion for particles within a system. By explicitly calculating the time evolution of particle positions and velocities, MD provides a dynamic picture of the system over time.

In contrast, instead of following deterministic pathways, MC generates configurations in a probabilistic way using techniques such as the Metropolis algorithm. This randomness allows MC to efficiently explore equilibrium properties and thermodynamic averages without the need of time-dependent information. As a result, MC simulations are particularly useful for studying equilibrium.

Together, these methods complement each other and form a nice toolkit for computational research in molecular systems.

In this report, a Metropolis Monte Carlo (MC) simulation for the NVT ensamble is implemented based on the Molecular Dynamics (MD) simulation build during previous assignments. Section 3 gives an analysis of the system by studying some time-independent correlations such as the average Radial Distribution Function $\bar{g}(r)$, specific heat capacity c_V or the average potential energy $\bar{U}$.

2 Theory and Methods

2.1 The Metropolis Method

The Metropolis Monte Carlo (MC) method is a stochastic algorithm used for sampling configuration space with a probability distribution proportional to the Boltzmann factor. In this section, we describe the method in detail, derive the Metropolis scheme for transition probabilities, and outline the specific implementation used in our simulation.

2.2 Markov Chain and Ergodicity

The Metropolis MC method is based on a Markov chain, where the probability distribution of states evolves according to a transition matrix that satisfies the detailed balance condition. Ergodicity ensures that all states can be reached, given a sufficient number of steps, allowing for proper sampling of phase space.

2.3 Boltzmann Distribution and Detailed Balance

In equilibrium, the probability of a system being in a given microstate i is proportional to the Boltzmann factor:

$$P(i) \propto e^{-\beta U_i}, \tag{1}$$

where $\beta = 1/(k_B T)$ is the inverse temperature, and U_i is the potential energy of the system in state i. The transition probability $\pi(o \to n)$ from an old state o to a new state n must satisfy the detailed balance condition:

$$P(o)\pi(o \to n) = P(n)\pi(n \to o). \tag{2}$$

Substituting the Boltzmann factors, we obtain

$$e^{-\beta U_o}\pi(o \to n) = e^{-\beta U_n}\pi(n \to o). \tag{3}$$

By choosing a symmetric proposal distribution, i.e., $\alpha(o \to n) = \alpha(n \to o)$, the Metropolis acceptance criterion is given by:

$$acc(o \to n) = \min\left(1, e^{-\beta \Delta U}\right), \tag{4}$$

where $\Delta U = U_n - U_o$ is the energy difference.

2.4 Algorithm Implementation

The algorithm proceeds as follows:

1. Generate an initial configuration of particles in a cubic lattice.
2. Compute the potential energy U_o of the system.
3. Select a random particle and propose a displacement:

$$x_{new} = x_{old} + \Delta(\text{rand} - 0.5) \tag{5}$$
$$y_{new} = y_{old} + \Delta(\text{rand} - 0.5) \tag{6}$$
$$z_{new} = z_{old} + \Delta(\text{rand} - 0.5), \tag{7}$$

   where Δ is a small displacement that controls how much the particle's position changes and rand is a random number uniformly distributed between $[0, 1)$.
4. Apply periodic boundary conditions.
5. Compute the new potential energy U_n.
6. If $\Delta U < 0$: the move is accepted and we add ΔU to the total potential energy.
7. If $\Delta U > 0$: accept the move with probability:

$$P_{\text{acc}} = \min(1, e^{-\beta \Delta U}). \tag{8}$$

   again, we add ΔU to the total potential energy.

   If the move is rejected, go back to the old configuration and keep the total potential energy the same.
8. Repeat for the determined number of steps.

2.5 Contribution from Rejected Moves

In the Metropolis MC algorithm, every attempted move contributes to the average that we are computing, regardless of whether it is accepted or rejected. This ensures that all possible configurations are explored in a fair and accurate way in the simulation.

2.6 Choice of Step Size and Acceptance Ratio

A crucial aspect of Metropolis MC simulations is selecting an appropriate trial displacement step size Δ. If the step size is too small, the particles only make tiny moves and the simulation takes a long time to explore all the possible configurations. On the other hand, if the step size is too big, most moves are rejected because they make the system less stable so the simulation does not collect useful data. The optimal acceptance ratio for our implementation is around 30%.

2.7 Specific Heat Capacity c_V

We already gave a definition for the Specific Heat Capacity in Assignment 3. Nevertheless, since we are now working in the NVT ensamble, we use the following formula for its computation:

$$c_V \approx \left(\frac{3k_B}{2} + \frac{N\sigma^2(u)}{k_B T^2} \right) \tag{9}$$

where N is the total number of atoms, $u = U/N$ is the potential energy per atom, T is the real temperature of the system and the variance in u is computed as:

$$\sigma^2(u) = \frac{1}{n} \sum_{i=1}^{n} (u_i - \bar{u})^2 \tag{10}$$

from a large number n of time steps.

2.8 Previous Implementations

Calculations for the Potential Energy, Radial Distribution Function (RDF) $g(r)$ and block averaging are obtained using the implementations from previous Assignments: 2, 3 and 4.

3 Results and Discussion

The simulation was run $N = 125$ atoms, 200,000 steps and equilibrium was reached at around 11,000 steps. Acceptance ratio of 33%.

Here, we present the results obtained for the potential energy regarding the step size Δ choice and once equilibrium is reached: the calculations for the average potential energy $\bar{U}$ with its corresponding standard error $\sigma(\bar{U})$, average RDF $\bar{g}(r)$ and specific heat capacity c_V.

A comparison is made between the results obtained using MD and MC as well as a brief comment about their computational cost and efficiency.

3.1 Potential Energy Analysis

The following graphs plot the Potential Energy id the system for different values of Δ:

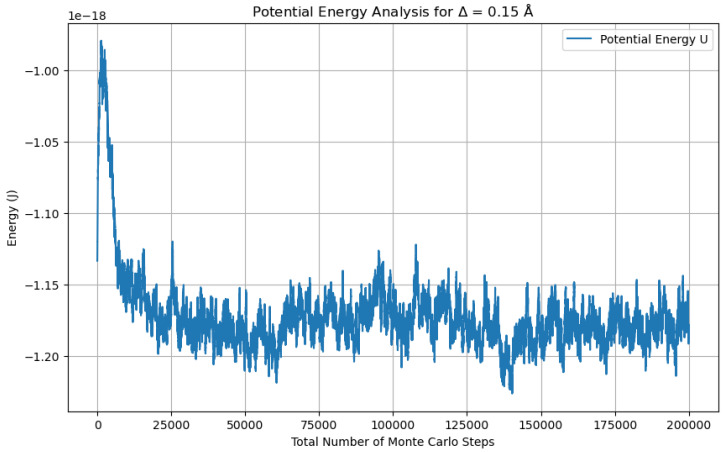

Figure 1: Potential energy over Monte Carlo steps for $\Delta = 0.15$ Å.

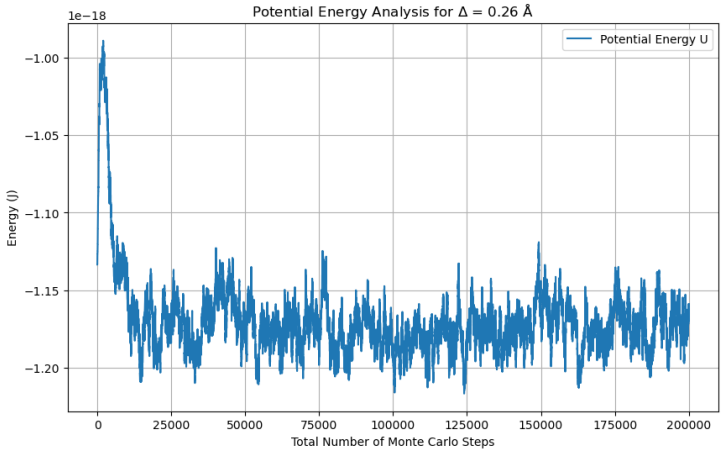

Figure 2: Potential energy over Monte Carlo steps for $\Delta = 0.26$ Å.

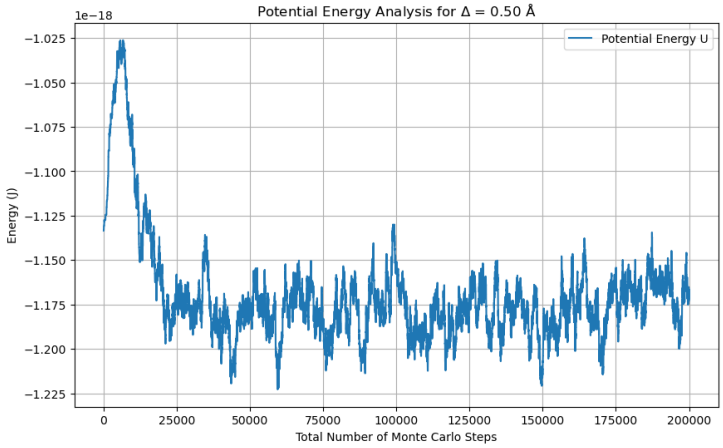

Figure 3: Potential energy over Monte Carlo steps for $\Delta = 0.50$ Å.

Simulation from Figure 1 took 7 minutes and 32 seconds to run and gave an acceptance ratio of 56%. The simulation in Figure 2 took 7 minutes and 14 seconds to run with an acceptance ratio of 33%. Lastly, Figure 3 took 8 minutes and 12 seconds to run and gave an acceptance ratio of 11%.

What we first note is that the bigger the step size, the lower the acceptance ratio is.

The step size is the smallest for Figure 1, causing the system to take small incremental steps without exploring distant configurations efficiently. Hence, while the simulation still converges, the process is less efficient compared to the optimal acceptance ratio.

Δ is the largest for Figure 3, which leads to frequent rejections (178,428 out of 200,000 total moves) due to large energy differences. Moreover, the potential energy is the most uneven of the three. There are larger fluctuations with a poor exploration of configuration space (areas much more sampled than others due to the high amount of rejected moves). Furthermore, the simulation suffers from slow equilibration (after 25,000 steps) compared to the optimal acceptance ratio.

Figure 2 gives the optimal acceptance ratio of 33% and the fastest simulation. The plot shows well-sampled trajectory of the potential energy, indicating efficient exploration of phase space. The system reaches equilibrium quickly (around 11,000 steps) and fluctuations are representative of thermal equilibrium at the given temperature. There is a good balance between accepting and rejecting moves, ensuring both efficient exploration and accurate sampling. Figure 2 resembles the most the analysis of the potential energy in MD.

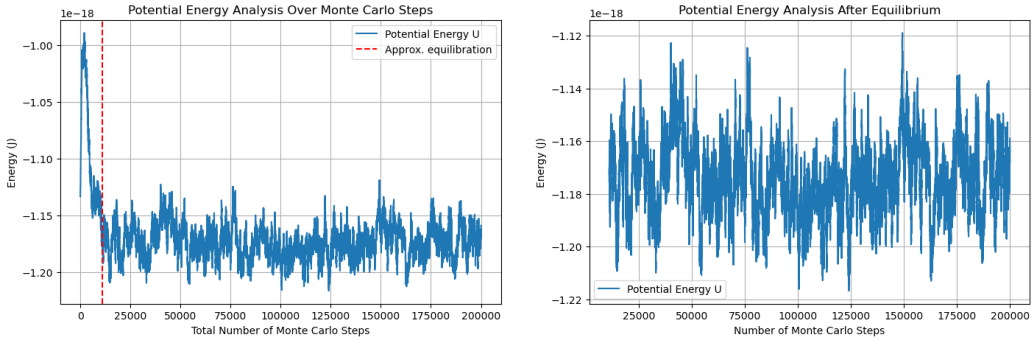

Figure 4: Potential energy evolution over Monte Carlo steps for an acceptance ratio of 33%.

3.2 Analysis after equilibrium

The average potential energy with its corresponding standard error is of:

$\bar{U} = (-1.1744 \pm 0.0013) \times 10^{-18}$ J. Hence, $\sigma(\bar{U}) = 0.0013 \times 10^{-18}$ J $= 1.3 \times 10^{-21}$ J for a block size of 5120, see Figure 5.

Blocking average for MD with 20,000 steps gave a result of:

$\bar{U} = (-1.1770 \pm 0.0014) \times 10^{-18}$ J for a block size of 1280.

We can observe very similar values for both simulations. Block sizes where chosen accordingly to reaching the plateau in the corresponding blocking analysis plots. Since the number of steps differ from one simulation to another, MC has more number of blocks sizes than MD. Furthermore, MD

reaches equilibrium at 1000 steps and MC does at 11,000 steps. Hence, the difference in block size choice for arriving at the plateau.

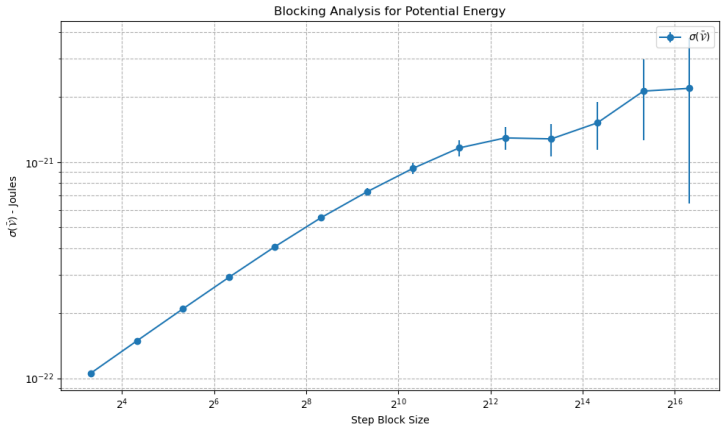

Figure 5: Block Analysis for $\sigma(\bar{U})$. Plateau observed after block size 2^{12}.

Specific heat capacity for MC gives a value of: $c_V \approx 20.67$ JK^{-1}mol^{-1} which is very similar to the one obtained in MD: $c_V \approx 19.47$ JK^{-1}mol^{-1}

Similarly, the average RDF $\bar{g}(r)$ resembles completely the one obtained for MD.

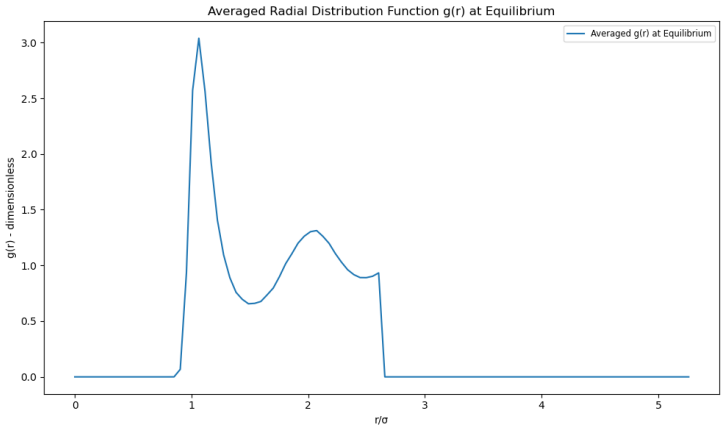

Figure 6: Time-averaged RDF at equilibrium for MC simulation.

It is worth noting why there is a sudden drop off at about 2.6 r/σ. The function does not converge to one due to the minimum image convention and the condition we have imposed for $r < L/2$ where L is the size of the box. Meaning that the maximum possible pair distances between atoms are limited by half the box length. Once r reaches this limit, there are simply no more atom pairs available at larger distances to contribute to $g(r)$.

3.3 Efficiency versus Simplicity

We have been able to observe how MD and MC simulations yield very similar results.

Nevertheless, there is a key difference. MD is much more efficient and faster than MC, with a run-time of 45 seconds and 20,000 steps it achieves very similar results to MC, which needs 200,000 steps and more than 7 minutes to run (we are moving one atom at a time). MD reaches equilibrium at 1000 steps while MC does at 11,000 steps.

On the other hand, the Metropolis Monte Carlo algorithm is much simpler to implement than MD. When we get rid of time in MC, we forget about velocity and force calculations as well as their implications such as velocity rescaling.

We conclude that, given the properties asked to study for our given system of liquid Argon, the choice between MC or MD is made by answering the following question:

What would you give up: efficiency or simplicity?