Molecular Dynamics Simulation

AP3082-CP

Delft University of Technology

Isacco Gobbi (5161347), Brennan Undseth (5021480), Ludwig Hendl (5154375)
1^{st} April 2020

Abstract

A molecular dynamics simulation using up to 365 particles interacting via the Lennard-Jones potential is implemented to study the thermodynamic properties of a system of argon atoms in the solid, liquid, and gas phases. Accurate results for the pair correlation function, diffusion coefficient, pressure, and heat capacity for all phases are reproduced and compared against the famous results published by Verlet and contemporaries with good agreement. Notably, the pair correlation function and a diffusion coefficient of $2.42 \pm 0.4 \times 10^{-5}\ cm^2/s$ for liquid argon are reproduced. The heat capacity shows good qualitative agreement and behaves as expected in the Dulong-Petit and ideal gas limits. Accurate values of pressure are found in the low-temperature regime. Discrepancies and limitations of the simulation method are also discussed.

Contents

1 Introduction

Molecular dynamics simulations open a window to study physical situations which are highly difficult to realize experimentally due to extensive pressures, temperatures or other parameters. They also provide a tool to reproduce experiments with the aim of gaining further insights on the involved physics. Two pioneers of such computer simulations are Rahman and Verlet [3][4]. In the 1960's, they began to study the molecular dynamics interaction between atoms and molecules, mainly Argon. This noble gas was a good choice since it consists of uncharged atoms not forming chemical bonds between each other. Rahman and Verlet were one of the first to simulate the behaviour of many body systems by measuring thermodynamic quantities with "computer experiments" that simulated a system of 864 particles interacting through a Lennard-Jones potential.

Within this report we want to reproduce the results obtained by Rahman, Verlet, and their contemporaries by simulating a similar microcanonical ensemble of Argon atoms interacting via a Lennard-Jones potential in the gaseous, liquid, and solid phases. First, we elaborate in more detail the methodology of our Python 3 simulation as well as the correctness checks that were used to verify proper operation. Then, we focus on the results of our simulations. The thermodynamic quantities of interest will be the pair correlation function, diffusion, pressure, and heat capacity. Each of these observables will be discussed in turn. Specifics of the Python code and an analysis of its performance are given in the Appendix. All of the code used to produce the results presented here is available in the master branch of the associated repository.

2 Method

We describe a microcanonical system where the particle number N, volume V, and the total energy E are held constant. We consider all atoms to interact through the two-body Lennard-Jones potential:

$$U(r) = 4\epsilon \left[\left(\frac{\sigma}{r}\right)^{12} - \left(\frac{\sigma}{r}\right)^{6} \right] \tag{1}$$

where $\epsilon = k_B \cdot 119.8$ K and $\sigma = 3.405$ Å. We use a system of natural units that minimizes use of physical constants by transforming $r \to \frac{r}{\sigma}$, $E \to \frac{E}{\epsilon}$, and $t \to t/\sqrt{\frac{m\sigma^2}{\epsilon}}$. With this convention, Equation 1 becomes:

$$U(r) = 4[r^{-12} - r^{-6}] \tag{2}$$

and the time evolution is subject to the equation of motion:

$$\frac{d^2x}{dt^2} = F(x(t)) = -\nabla U. \tag{3}$$

Before the time-evolution of N argon atoms can begin, they must be initialized in the fixed volume V with some energy E. Here, each atom is completely described by its position and velocity within the volume. We therefore seek to generate initial positions and velocities that are physical and lead to a physically meaningful system evolution.

To this end, we initialize particles in a sufficiently large cubic volume ($V = L^3$) at positions defined by the face centered cubic lattice (fcc) crystal structure, the crystal structure of solid Argon. For simulations of the solid phase, we can adjust the particle density such that the lattice constant is approximately 1.2σ (the minimum of the Lennard-Jones potential) and then the atoms retain the crystal structure as the simulation evolves. For liquid and gaseous phases, a lower density is used, and the fcc initialization ensures that particles begin sufficiently far apart that the repulsive r^{-12} term of the potential is not dominant at the beginning of the simulation. Velocities are randomly assigned to each particle according to a Maxwell-Boltzmann distribution with a characteristic temperature T that is set at the beginning of the experiment. The average velocity is adjusted to be zero such that there is no net particle drift.

To simulate a large number of particles efficiently, we make use of periodic boundary conditions such that an atom moving over the boundary of the box will re-emerge with the same velocity at the opposite face of the box. Interactions between particles according to Equation 2 also occur across the periodic boundaries. We use the minimal image convention, and only consider the pairwise interaction between a particle and the nearest image of all $N-1$ other atoms, such that no two atoms may be more than $L/2$ apart in each Cartesian coordinate.

Under these conditions, we evolve the particles' positions and velocities in finite time steps h using the Velocity-Verlet algorithm:

$$\begin{cases} x(t+h) = x(t) + h \cdot v(t) + \frac{h^2}{2} F(x(t)) \\ v(t+h) = v(t) + \frac{h}{2} \left[F(x(t+h)) + F(x(t)) \right] \end{cases} \tag{4}$$

This numerical method is a symplectic integrator which performs a transformation in phase space that is volume-preserving. The importance of this approach here is that, unlike more naive time evolution like the Euler method, the Verlet algorithm will conserve energy. Otherwise, our system would not be a microcanonical ensemble.

Since it is more meaningful for us to speak of a simulation at a certain temperature rather

than at a certain energy, it is important that our system, at equilibrium, maintains the desired constant temperature. Since temperature is one-to-one with total kinetic energy K, we aim to have the system reach equilibrium with a stable K. Initializing the particle velocities is not enough on its own, as some of this kinetic energy will be transformed into potential energy and vice versa. To overcome this, we allow the system to reach equilibrium by letting it evolve for some time, then scale the velocities by a uniform constant such that the kinetic energy corresponds once again to the desired temperature. We iterate this process until the equilibrium temperature matches the desired temperature within some tolerance. Of critical importance here is the size of the time step h and the number of time steps we allow the system to evolve before re-checking its temperature. Taking inspiration from Verlet, who used a time step of 10 fs, we set $h = 0.001$ in our natural units, corresponding to 2 fs [4]. A small time step allows the particles to move smoothly and not run into each other which prevents peaks in the potential energy (and subsequent velocity spikes) due to the r^{-12} repulsive term in the Lennard-Jones potential. We allow the system to evolve for 100 time steps (0.2 ps), as it was empirically found that there was no noticeable drift in temperature after the equilibration routine.

Once equilibrium is reached, we can begin the proper experiment. Thermo dynamic observables are computed as the time average of the physical quantity of interest. For an observable A is given by:

$$\langle A \rangle = \frac{1}{N_T} \sum_{n>0}^{N_T} A_n \tag{5}$$

where N_T is the number of simulation timesteps. The correspondence between time average and ensemble average in this context is assumed in virtue of the ergodic theorem. In principle, to calculate the error on the thermodynamic averages one needs to compute the standard deviation of the gathered data. In this scenario one needs to pay particular attention to correlations, since, in general, data collected at subsequent simulation times will not be independent from previous states of the system. One way to quantify this is to calculate the autocorrelation function, which is defined as:

$$\chi_A(t) = \frac{1}{\sigma_A^2} \sum_n \left(A_n - \langle A \rangle\right)\left(A_{n+t} - \langle A \rangle\right) \tag{6}$$

Is is reasonable to assume that this function decays like an exponential $e^{-\frac{t}{\tau}}$, therefore we can estimate the *autocorrelation time* τ by fitting $\chi_A(t)$ to this function.

The error on on the thermodynamic average $\langle A \rangle$ is then given by:

$$\sigma_A = \sqrt{\frac{2\tau}{N}\left(\langle A^2 \rangle - \langle A \rangle^2\right)} \tag{7}$$

In general, we expect more accurate results for longer simulations with a large number of particles. However, simulations required truncation in order to allow time for simulating thermodynamic quantities for a range of system parameters. We find a good compromise by using $N = 365$ with N_T on the order of 10,000, simulating the system for a physical time on the order of 10 ps. In some cases, different particle numbers and simulation times are used, and these are discussed case-by-case in Results.

2.1 Correctness Checks

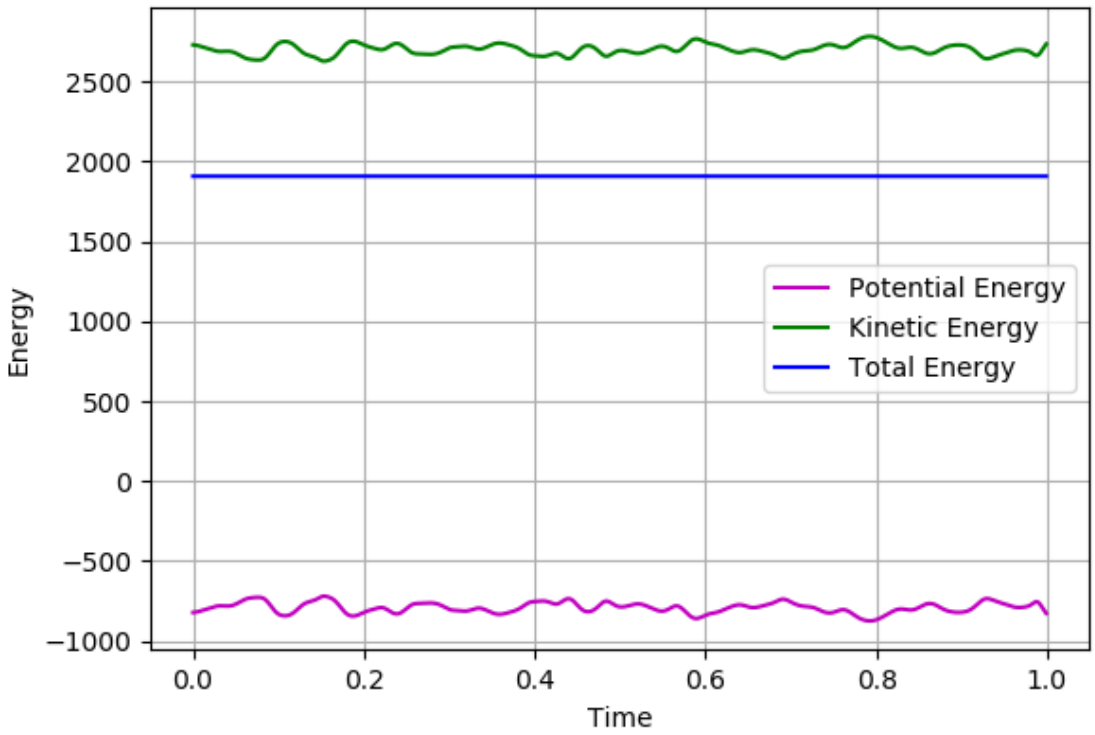

Figure 1: A short simulation of 365 argon atoms at 600 K for 1000 time steps of 0.001 in natural units (total time 2.15 ps). Plot begins after equilibrium is reached, and total energy is conserved throughout the simulation.

Proper operation of the simulation was verified in multiple stages. First, it was ensured that two particles interacting via the Lennard-Jones potential given by Equation 2 could evolve in 2 dimensions according to Equation 3 using a naive Euler method in a square with periodic boundary conditions. The particles were plotted in a scatter plot that was updated at east time step, such that discrepancies in the periodic conditions and particle interaction could be seen visually and debugging carried out accordingly.

Once this setup was working properly, the volume was extended to 3 dimensions, and

the numerical time evolution was upgraded to the Verlet algorithm given in Equation 4. By initializing two particles moving along a co-linear path and tracking the total kinetic energy K, total potential energy U, and total energy E, it could be verified that the simulation was energy-conserving. This allowed for the addition of more than 2 particles initialized in the fcc structure. By confirming energy conservation for simulations beginning with various particle velocities and densities, we could be confident in the correct operation of our implementation. Similarly, the energy plots could be used to verify the validity of our equilibration routine, by ensuring that the mean kinetic energy of the system did not drift after velocity re-scaling, as in Figure 1.

Subsequent correctness checks depended on comparing calculated observables to results published in literature. Reproducing pair correlation functions for gaseous, liquid, and solid phases allowed us to pinpoint good operating points for our simulation, and replicating values for pressure, heat capacity, and diffusion gave us good confidence in the dynamics as our system scaled to contain hundreds of particles. For all simulations, the plotted energies could be consulted to once again ensure energy was conserved and that no unexpected drifts occurred in kinetic or potential energies. Deviations in our computed observables from the literature highlighted areas of improvement for our simulation. Some of these were able to be addressed, and they are discussed as they are presented in the following section.

3 Results

3.1 Pair Correlation Function

The Pair Correlation Function (PCF), sometimes called $g(r)$, can give an idea of the state of the system for a chosen temperature T and density ρ. Informally, it is the distribution of particle-particle distances r normalized with respect to an isotropic distribution. Formally the PCF is defined as:

$$g(r) = \frac{2V}{N(N-1)} \frac{\langle n(r) \rangle}{4\pi r^2 \Delta r} \tag{8}$$

where $\langle n(r) \rangle$ is the expectation value for the number of particle pairs separated by a distance in $[r, r+\Delta r]$. An isotropic distribution is the case one would expect, for example, for an ideal gas. On the other hand, a crystalline structure where particle motion is confined to a small region around the lattice site will present a distribution that is sharply peaked around site-site distances.

In order to highlight these differences we simulated our system for three values of ρ and T corresponding to three different phases of Argon. In particular for the liquid case we chose the same set of parameters as in the aforementioned Rahman simulation [3]. All

	Solid	Liquid	Gas
T [K]	10	94.4	300
$\rho\sigma^3$	0.81	0.81	0.01

Table 1: Pair Correlation Function and Diffusion parameters.

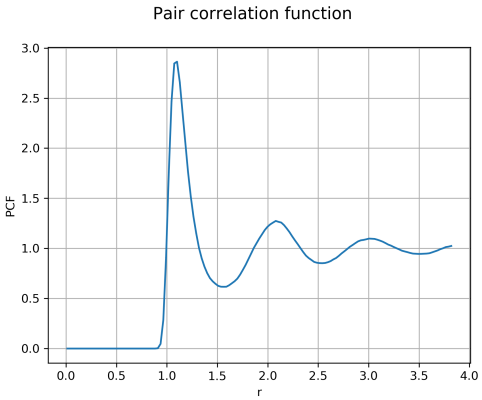

Figure 2: Simulated pair correlation function for liquid Argon with $T = 94.4\ K$ and $\rho\sigma^3 = 0.81$. r is plotted in units of σ.

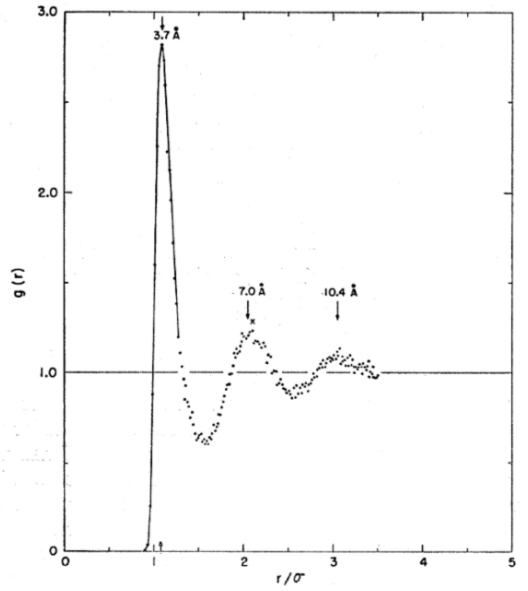

FIG. 2. Pair-correlation function obtained in this calculation at 94.4°K and 1.374 gcm^{-3}. The Fourier transform of this function has peaks at $\kappa\sigma = 6.8, 12.5, 18.5, 24.8$.

Figure 3: Simulated pair correlation function for liquid Argon from [3]

simulations were performed with $N = 365$ for 20000 time steps ($h = 0.001$, total time of 43 ps).

We can see that our simulated pair correlation function for liquid argon, shown in Figure 2, is in good agreement with the simulation presented by Rahman, shown in Figure 3.

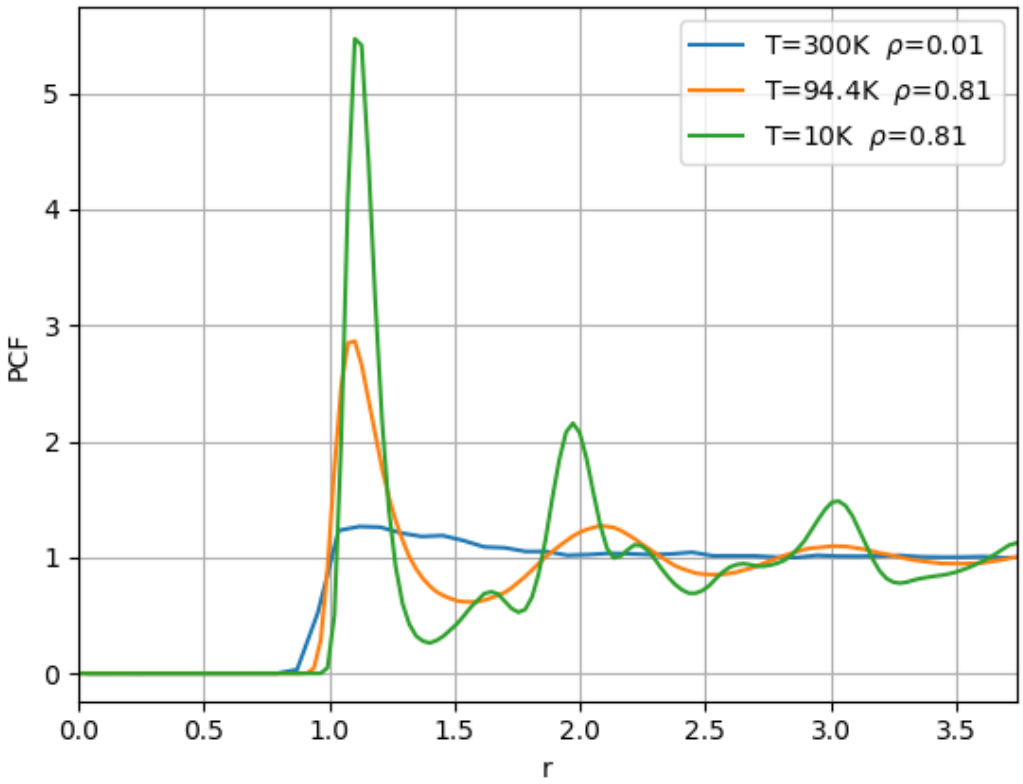

Figure 4: Simulated pair correlation function for the gaseous (blue), liquid (orange), and solid (green) regimes. r is plotted in units of σ.

Figure 4 shows the pair correlation function for the three phases of argon. Although we were only able to compare the liquid case to literature, the other two plots show the expected behaviour: for the gas phase it shows a nearly flat, isotropic distribution, as we would expect from an ideal gas. This is due to the fact that in this limit, the density is so low that the attractive tail of the Lennard-Jones potential is negligible and the event of a collision between particles is rare enough to be smoothed out by the time average. At the other extreme, the parameters for what we would expect to be a solid show sharper peaks, indicating the presence of a more ordered structure.

3.2 Diffusion

Calculations of the diffusion coefficient D for different phases of the microcanoncical ensemble of Argon atoms were performed. In literature, the coefficient is referred to as self-diffusion once equilibrium is reached. To make our results comparable with literature and the pair correlation function analysis (PCF), we choose the same parameters as given in table 1. Those parameters refer to the different phases of the system and will

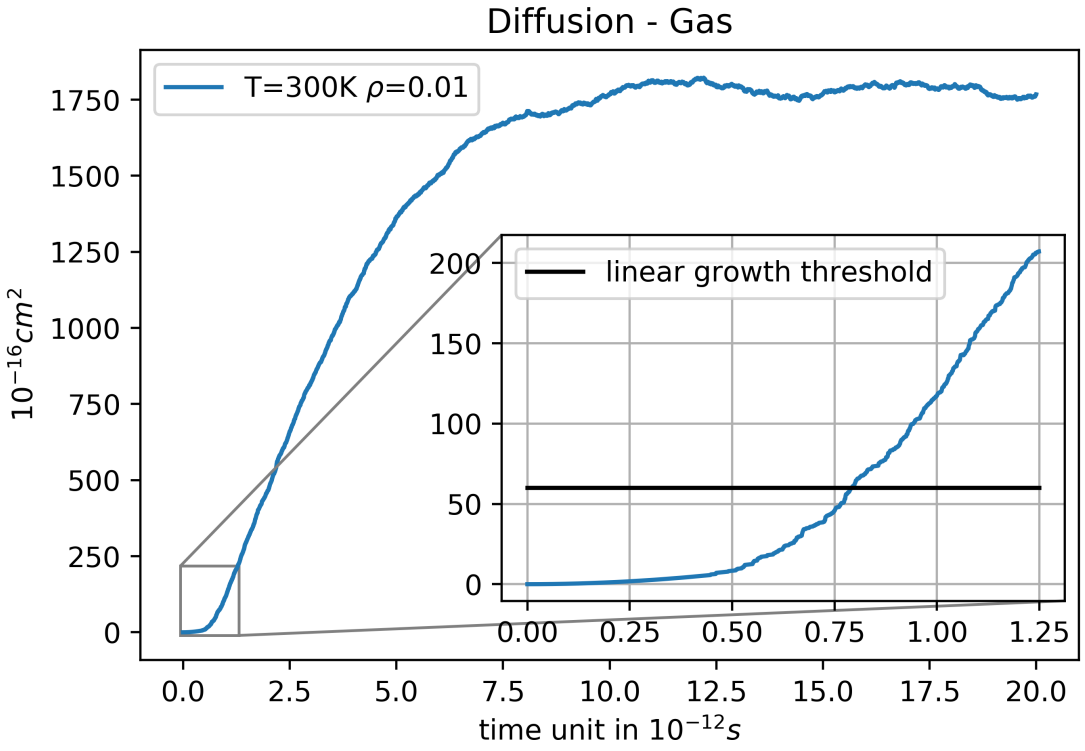

Figure 5: Mean square displacement in gas phase. Inset shows zoom into first 1.25 time units. The horizontal line at 60 $Å^2$ indicates the start of the linear regime. The linear fit corresponds to an experimental diffusion coefficient of $D = 5.5 \times 10^{-3} \pm 2.6 \times 10^{-3} cm^2/s$.

be compared to Rahman's simulation [3] as well as the preceding section. Since the diffusion coefficient specifies the random thermal movement of particles we expected the results of the PCF and the presented diffusion data to correlate, which is indeed the case. The diffusion coefficient can be calculated from the mean square displacement $\langle r^2 \rangle$ of all particles i as given according to literature ([3], [1]):

$$\langle r^2 \rangle = \frac{1}{N} \sum_{i=1}^{N} \left(r_i \left(t + t_0 \right) - r_i \left(t_0 \right) \right)^2 . \tag{9}$$

Here, t_0 is the time origin set to the point when the system reached equilibrium. We plot the mean square displacement (MSD) for a range of times t to understand the motion of particles depending on the phase of the system. From the plots, we can calculate the corresponding diffusion coefficients using the Einstein relation $D = \lim_{t \to \infty} \frac{\langle r^2 \rangle}{6t}$. Values for D are obtained by fitting the linear part of the MSD curve to the function $6Dt + C$ where the slope corresponds to six times the diffusion coefficient.

All simulations were performed with $N = 365$ for 20000 time steps (h=0.001, total time

43 ps). The curves present an average of 20 individual simulations with the same number of different time origins t_0 verifying the time independence of the diffusion coefficient for systems in equilibrium. We expect the movement of atoms, hence the diffusion coefficient, to be greatest in the gas phase, lower in the liquid phase and taking the smallest value in the solid case. As the PCFs already provided clear evidence of the different phases for the parameters in Table1, we observe the expected behavior of the diffusion as shown in figure 5 and 6. The gaseous diffusion curve shows three areas. Initially we do not observe linear growth but rather quadratic growth until approximately 60 $Å^2$ (inset, horizontal line). This in turn relates to approximately half the nearest neighbor distance of 15.96 Å in the low density gas phase. Until atoms moved this distance one can expect little interaction and hence quadratic growth of the MSD (ballistic case) whereas beyond this distance collisions start to have an effect by linearising the curve and giving rise to the actual diffusion (figure 5). The third area starts after approximately 5 time units and represents an upper bound of the MSD resulting from the simulation being finite. Since we are not summing over the distance gained by periodic boundary condition jumps, there is a maximum distance atoms can travel with respect to their starting point. This maximum is reached for the gas phase since atoms are initialized with a much higher velocity compared to the liquid and solid case.

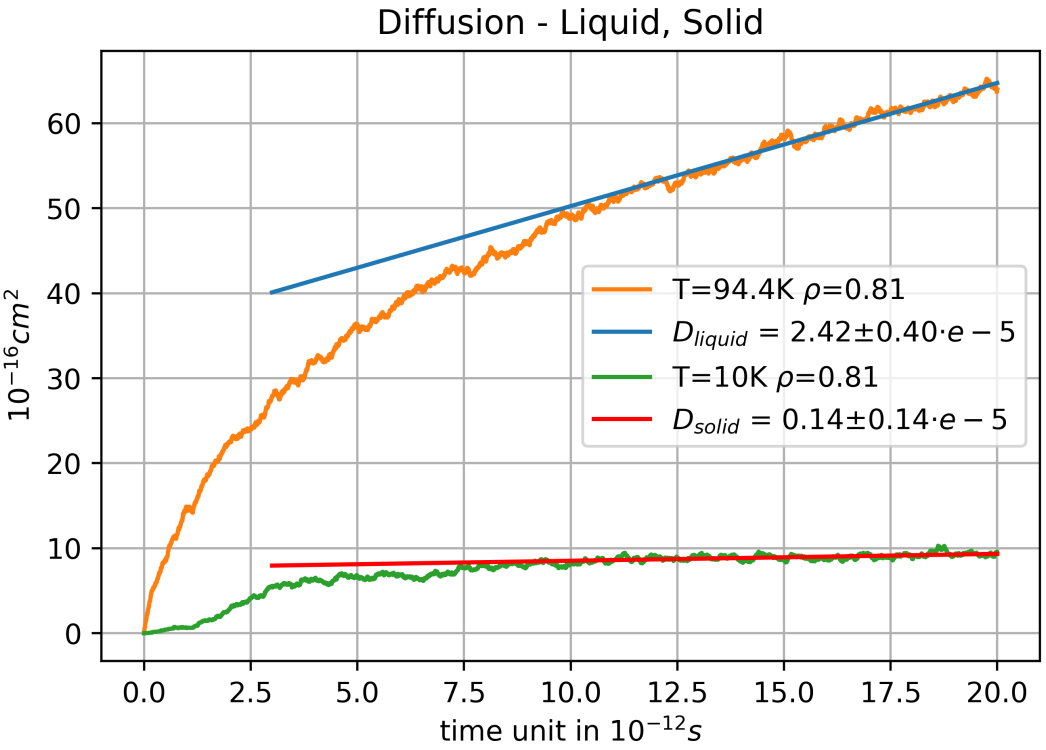

Figure 6: Mean square displacement in liquid (orange) solid (green) phase. Linear fits in the late time limits for calculations of the diffusion coefficients D.

The diffusion coefficients were calculated as described above. A plot of the linear fit is only shown for the liquid and solid case (figure 6) in the limit of large simulation times ($t \geq 13$ time units). The corresponding results are given in table 2. The errors corresponding to the linear regression are negligible small.

D in cm^2/s	Solid	Liquid	Gas
Simulation	1.4×10^{-6}	2.42×10^{-5}	5.5×10^{-3}
Simulation Error	$\pm 1.4 \times 10^{-6}$	$\pm 0.4 \times 10^{-5}$	$\pm 2.73 \times 10^{-3}$
Rahman	-	2.43×10^{-5}	-

Table 2: Diffusion coefficients calculated for parameters given in table 1 with respective errors and the value obtained by Rahman for the same parameters.[3].

The simulated diffusion coefficient of $2.42 \pm 0.4 \times 10^{-5}\ cm^2/s$ for the liquid phase agrees very well with the value obtained by Rahman [3]. The solid phase shows a value decreased by one order of magnitude as a result of reduced mobility in the a solid. On the contrary, D in the gas phase is increased by two orders of magnitude resulting from the great freedom of the atoms. Generally we can see that our expectation of the diffusion on different phases is in good agreement with the simulations.

3.3 Pressure

The pressure of the system was calculated on the basis of the virial theorem [4].

$$\frac{P}{k_b T \rho} = 1 - \frac{1}{k_b T 3N} \left\langle \sum_{i,j}^{N} r_{ij} \frac{\partial U_{ij}}{\partial r_{ij}} \right\rangle \tag{10}$$

Here, k_b is the Boltzmann constant, T the temperature, ρ the density, N the number of particles, r_{ij} the distance between the particles, and finally $\frac{\partial U_{ij}}{\partial r_{ij}}$ the partial derivative of the Lennard-Jones potential with respect to particle distances. Simulation values of the pressure are kept in the natural unit of $\frac{\epsilon}{\sigma^3}$. The pressure was extracted from simulations using 365 atoms and 10000 timesteps ($h = 0.001$).

Figure 7 displays two simulated isochores for the densities $\rho = 0.45$ and $\rho = 0.85$ along with values published in [4]. The simulated values agree well with literature in the range of low temperatures.

During our simulations we observe fluctuations in the temperature of the system which leads to deviations in the calculated pressure values. Little changes in temperature in-

fluence the pressure significantly as implied by Verlet [4]. One possible reason for the temperature fluctuations originates from the equilibration process during the initialization of the simulations. It was observed that for extended initialization times we found increasingly reproducible pressure values using our simulation. This was especially the case for low densities where we find atoms moving greater distances as described in section 3.2. Thus, atoms are prone to less interaction which might result in better averaging of the inter-atomic distances which in turn influences the pressure.

However, at low density ($\rho\sigma^3 = 0.45$) we obtain results closer to Verlet at higher temperatures than we do at high density ($\rho\sigma^3 = 0.85$). The flattening out of our simulated pressure values in figure 7 might correspond to the size of the periodic box used in our simulations. Stagnation of pressure values arises when the distance between nearest neighboring atoms cannot increase anymore. This is similar to the effect observed for the diffusion in gas phase: for high temperatures a limit due to the box size is reached. Both observables depend mainly on atom positions. This motivates us that our simulation is not set up to obtain reasonable values of pressure for parameters beyond a certain temperature for constant densities since we operate at a limited simulation volume. It may also be that our simulation reaches asymptotic pressures at a lower temperature due to using fewer atoms.

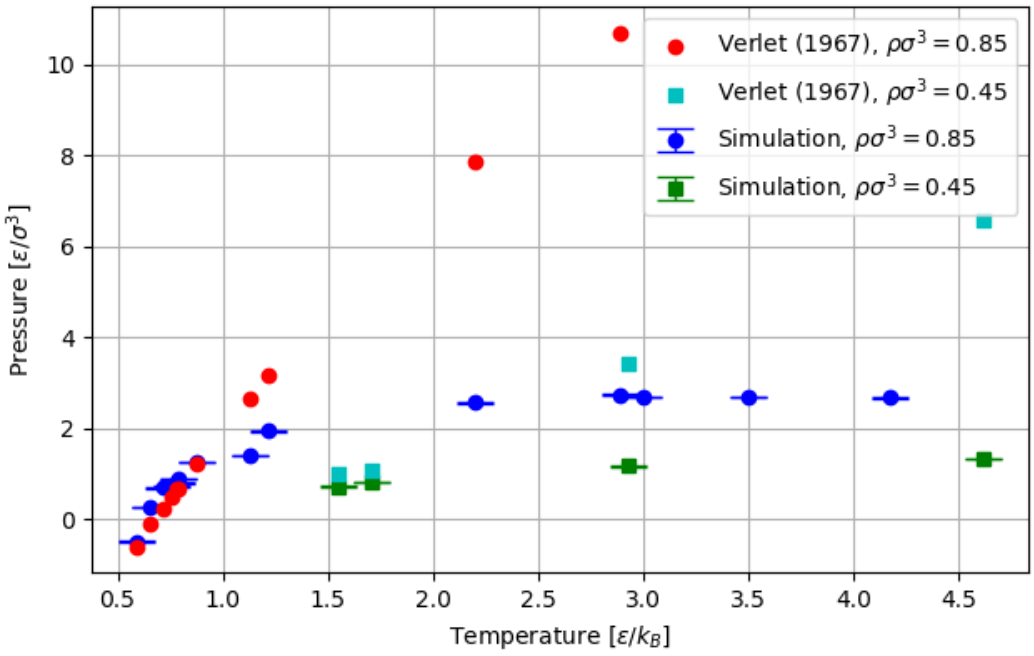

Figure 7: Pressures calculated for $\rho = 0.85$ and $\rho = 0.45$ for a series of temperatures ranging from solid to gas. The experimental results are in good agreement with the values reported by Verlet for low temperatures.

3.4 Heat capacity

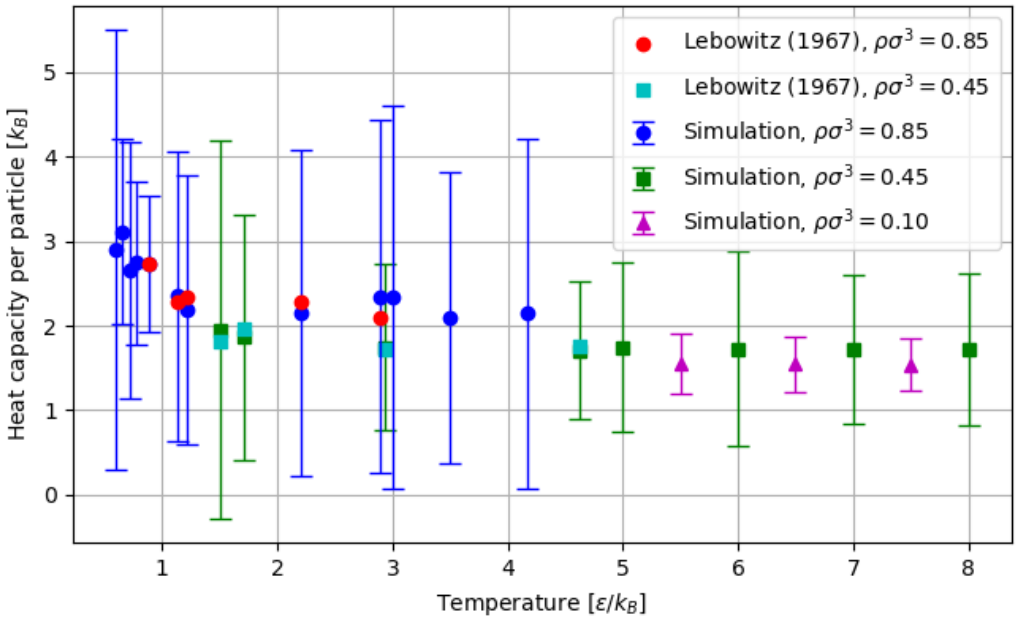

Figure 8: The heat capacity per atom for $\rho\sigma^3 = 0.85$, $\rho\sigma^3 = 0.45$, and $\rho\sigma^3 = 0.10$ for a series of temperatures ranging from high-temperature solid to ideal gas limits. $N = 63$ Argon atoms were simulated for a total of 8000 time steps (see body for discussion). Values reported by Lebowitz et al. are plotted for comparison.

The heat capacity was calculated according to a formula adapted from Lebowitz et al. [2]. In the paper, it is shown that the heat capacity per particle C could be related to the fluctuations in kinetic energy $\langle dK^2 \rangle = \langle K^2 \rangle - \langle K \rangle^2$ according to the relation:

$$\frac{\langle dK^2 \rangle}{\langle K \rangle^2} = \frac{2}{3N}\left(1 - \frac{3}{2C}\right) \tag{11}$$

where N is the number of particles. Lebowitz et al. apply a slightly modified formula, where factors of $\langle K \rangle$ have been replaced with the temperature via the relation $\langle K \rangle = \frac{3}{2}NT$. The results reported in [2] are given for a microcanonical ensemble with a similar setup to ours, and are thus good values to compare our results with.

The simulated heat capacities per particle are plotted in Figure 8. Good qualitative agreement is found between our results and [2] for both dimensionless densities $\rho\sigma^3 = 0.85$ and $\rho\sigma^3 = 0.45$. In the former case, the heat capacity is seen to drop from roughly $C \approx 3$ to $C \approx 2$ over a temperature range from $T = 0.6$ to $T = 4$ (corresponding to physical temperatures of 71 K and 476 K respectively). The $C \approx 3$ regime is expected from the Dulong-Petit law, which would predict a heat capacity of $3Nk_B$ for a high-temperature

solid. As the melting point of solid Argon at this density is roughly 80 K, the $T = 0.6-0.8$ range shows this limit. The decrease in C over the remaining temperature range agrees with experimental values for liquid Argon as noted in [2]. The lower-density $\rho\sigma^3 = 0.45$ simulations were carried out at higher temperatures ranging from $T = 1.5$ to $T = 8$ (178 K and 951 K) as this was the closest to the "ideal gas" limit for which we could compare results to literature. Again, we see good agreement with the results from [2]. If we push our simulation further into the ideal gas regime by letting $\rho\sigma^3 = 0.1$ at high temperatures, we find an asymptotic heat capacity per particle of 1.54. This agrees with the theoretical $C = \frac{3}{2}k_B$ for a monatomic ideal gas following from the equipartition theorem.

Uncertainties are calculated by rearranging Equation 11 into the form:

$$\frac{\langle K^2\rangle}{\langle K\rangle^2} = 1 + \frac{2}{3N} - \frac{1}{NC} \tag{12}$$

The uncertainty of the left hand side of Equation 12 can be calculated by propagating the uncertainties $d\langle K^2\rangle$ and $d\langle K\rangle$ as computed via the autocorrelation method:

$$\left(d\frac{\langle K^2\rangle}{\langle K\rangle^2}\right) = \left|\frac{\langle K^2\rangle}{\langle K\rangle^2}\right|\sqrt{\left(\frac{d\langle K^2\rangle}{\langle K^2\rangle}\right)^2 + \left(\frac{2d\langle K\rangle}{\langle K\rangle}\right)^2 - 2\frac{\mathrm{Cov}[\langle K\rangle, \langle K^2\rangle]}{\langle K^2\rangle\langle K\rangle^2}} \tag{13}$$

where $\mathrm{Cov}[\langle K\rangle, \langle K^2\rangle]$ is the covariance between K and K^2. This result can be propagated to give an appropriate uncertainty in the heat capacity per particle dC. As seen in Figure 8, the uncertainty in each simulation is sizeable, particularly for the $\rho\sigma^3 = 0.85$ solid/liquid phase simulations, and in some cases is of the same magnitude as the result itself. Lebowitz et al. state that their statistical errors are around 20%, much lower than for our results, although they are not given explicitly. Another peculiarity that was found is that the simulation here required a much greater number of time steps compared to Lebowitz et al for adequate convergence. There, they simulated using 1200 time steps for a total of 10 ps. Performing the same time evolution, our simulation would give inaccurate results with large uncertainties. It was also found that the heat capacity results were sensitive to particle number. While Lebowitz et al. used an 864 particle simulation, the maximum number of particles that we used here, 365, would produce inaccurate results that appear to converge slowly to the expected values for very long simulation times (50 ps). However, our error propagation method would yield nonsensical values if we carried out the simulation for too many time steps. Our simulation is thus whipsawed by these opposing restraints. It is for this reason that 63 particles were simulated for 8000 time steps for a total of 17 ps simulation time. It is unknown why the fluctuations in kinetic energy seem to vary more for large particle simulations with high densities. This

point of improvement is the limiting factor for further investigation of heat capacity at high densities and low temperatures with our simulator.

4 Conclusion

We have programmed a Python 3 simulator to experiment with the molecular dynamics of a microcanonical ensemble of argon atoms interacting under a Lennard-Jones potential. Using the Verlet algorithm to simulate the time-evolution of the system, we were able to reproduce a variety of thermodynamic observables in the solid, liquid, and gas phases of argon to a good agreement with both molecular dynamics literature and experiment. The pair correlation function was simulated for all three phases. The well-studied liquid-phase PCF was reproduced with excellent qualitative agreement. Solid and gas phase PCFs exhibited the expected shape for these regimes. The diffusion coefficient for the three phases could also be analyzed. Again, a diffusion coefficient of $2.42 \pm 0.4 \times 10^{-5} cm^2/s$ for the liquid phase was in excellent agreement with the value of $2.43 \times 10^{-5} cm^2/s$ reported in literature. Solid and gas phase diffusion coefficients were significantly smaller and larger respectively, as is expected. The equilibrium pressure was also studied for high and low argon densities. We found good agreement between our simulation and the literature for low temperatures, with our simulated pressure deviating at higher temperatures. It was argued that the nature of the finite box size with periodic boundary conditions and a limited number of particles could explain this discrepancy. The heat capacity was also simulated, and again good agreement was found between our results and literature for multiple densities and temperature ranges. In the high-temperature solid limit, a heat capacity of roughly $3k_B$ was found, in agreement with the Dulong-Petit law. For a low density gas at high temperature, a value of $\frac{3}{2}k_B$ was found, as we expect in the ideal gas limit. Large uncertainties and sensitivity to particle number and simulation duration were also discussed as limiting factors for specific heat simulations. In summary, we have shown that a molecular dynamics simulation with the Lennard-Jones potential can accurately predict numerous thermodynamic properties of a system of argon atoms.

A Source code

Here we present some aspects that we found particularly worth noting about the code.

All the function used in the simulation are contained in a separate file called `simulation_func_vector.py`, while the main program, `main.py`, only contains a dictionary in which the user can specify parameters and flags for the simulation setup. These include physical parameters (e.g. temperature and density) simulation parameters (e.g. number of timesteps) and flags for saving or showing plots. This way we have a clear and convenient way of changing the setup without the need to access the core of the simulation directly.

As mentioned above, the particle initialization is done according to an fcc lattice, therefore not all particle numbers are suitable for such a configuration. To facilitate the user at the execution this issue will be handled by the `init_position` function which will round the number of particles to the closest allowed number and calculate the system volume according to the user specified density or lattice constant. Particles are then placed onto an fcc lattice one by one, first on the underlying cubic lattice and then on each face sequentially. This way there is no need to precompute the exact number of particles before running a simulation but we can start with a "close enough" guess and the program will adjust it properly for us.

Finally, one of the most challenging parts of this project was to write a code that was easy to debug and change while still being fast enough to be usable. Initially we used global variables but soon we realized that this would have made our code impossible to debug in the long run. Changing to local variables was indeed a big upgrade for the overall readability.

After we started simulating, we realized that the code was still too slow to run with any significant number of particles. This was due to the fact that pairwise interaction was calculated with two nested for loops. The problem was overcome exploiting numpy vectorization and the comparison between the two versions of the code is shown in the next section.

A.1 Performance

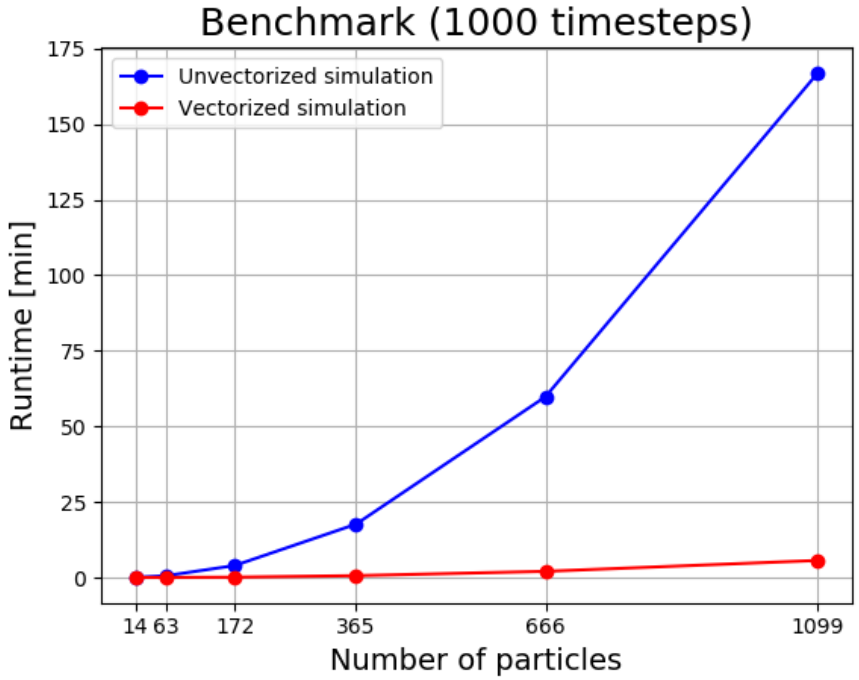

Figure 9: Benchmark comparing the two versions of the code: vecorized and unvectorized

The figure shows the improvement given by the vectorized version of the code. As we can see the difference in computation time is enormous for big particle numbers. One final remark is that despite being much faster its predecessor this new version of the code is still highly inefficient in resources: every particle-particle distance component is stored in a numpy array, irrespective of the symmetry of pairwise interaction. Therefore every computation involving the particle coordinates is done twice. Despite this fact, this version of the code is substantially faster than its predecessor and efficient enough to allow us to run all the desired simulations in a reasonable amount of time.

A.2 Work division

Work was distributed equally throughout the weeks while the core of the simulation software was in development. Brennan prototyped the initial version of the simulator and implemented periodic boundary conditions, debugged the first version with working energy conservation and thermal equilibration, and investigated the heat capacity observable. Isacco and Ludwig were responsible for implementing the Verlet algorithm and particle initialization. Ludwig modularized the user-facing script from the core simulator. Isacco improved the efficiency of the code to operate in its present form, as well as developed the autocorrelation error estimation method. Isacco was responsible for the pair

correlation function analysis, and Ludwig investigated diffusion and pressure and set up main suggestions for the experiment plan. All members contributed equally to debugging and report writing.

References

[1] RA Fisher and RO Watts. "Calculated Self-Diffusion Coefficients for Liquid Argon". In: *Aust. J. Phys.* 25 (5 1972), pp. 529–538. DOI: 10.1071/PH720529. URL: https://doi.org/10.1071/PH720529.

[2] J. L. Lebowitz, J. K. Percus, and L. Verlet. "Ensemble Dependence of Fluctuations with Application to Machine Computations". In: *Phys. Rev.* 153 (1 1967), pp. 250–254. DOI: 10.1103/PhysRev.153.250. URL: https://link.aps.org/doi/10.1103/PhysRev.153.250.

[3] A. Rahman. "Correlations in the Motion of Atoms in Liquid Argon". In: *Phys. Rev.* 136 (2A 1964), A405–A411. DOI: 10.1103/PhysRev.136.A405. URL: https://link.aps.org/doi/10.1103/PhysRev.136.A405.

[4] Loup Verlet. "Computer "Experiments" on Classical Fluids. I. Thermodynamical Properties of Lennard-Jones Molecules". In: *Phys. Rev.* 159 (1 1967), pp. 98–103. DOI: 10.1103/PhysRev.159.98. URL: https://link.aps.org/doi/10.1103/PhysRev.159.98.