

# SIMULATION OF THE MOLECULAR DYNAMICS OF ARGON

EPPENGA, M.D., FAASSE, G.M., TIRBAND DASTGERDI, C.

Computational Physics  
TU Delft  
Faculty of Applied Sciences  
MSc program Applied Physics

Delft, March 16, 2018

# ABSTRACT

In this simulation study, the thermodynamic behaviour of argon molecules in a Lennard-Jones potential are simulated using Verlet's algorithm in a finite box with periodic boundary conditions. Pressure and specific heat are computed and their errors are determined using bootstrapping. The pair correlation function is determined for different parameters and shows the expected behaviour for different phases.

The acquired values of the specific heat and pressure agree to a reasonable extent with those found in literature. However, uncertainties given by bootstrapping may give an unclear representation of the uncertainty of getting a similar value in a different realisation around phase transitions, due to uncertainties in the temperature.

Performance of the simulation is tested by increasing the number of particles in the simulation. The implementation shows  $\mathcal{O}(n^2)$  scaling, with the bulk of the computational intensity in the calculation of the potential for each particle due to all other particle.

This research is part of the course Computational Physics at the Delft University of Technology.

# Contents

<b>1</b>	<b>INTRODUCTION</b>	<b>1</b>
<b>2</b>	<b>THEORY</b>	<b>2</b>
<b>3</b>	<b>METHODS</b>	<b>3</b>
3.1	Simulating space . . . . .	3
3.2	Simulating the interaction forces . . . . .	3
3.3	Simulating the dynamics . . . . .	4
3.3.1	Euler's Method . . . . .	4
3.3.2	Verlet's algorithm . . . . .	4
3.4	Initial conditions . . . . .	5
3.5	Velocity correction . . . . .	5
3.6	Macroscopic properties . . . . .	6
3.6.1	Specific heat . . . . .	6
3.6.2	Pressure . . . . .	6
3.7	Pair Correlation function . . . . .	7
3.8	Bootstrapping . . . . .	7
<b>4</b>	<b>RESULTS AND DISCUSSION</b>	<b>8</b>
4.1	Energy conservation . . . . .	8
4.2	Pair Correlation function . . . . .	8
4.3	Temperature . . . . .	9
4.4	Specific heat . . . . .	10
4.5	Pressure and total internal energy . . . . .	11
4.6	Performance . . . . .	12
<b>5</b>	<b>CONCLUSION</b>	<b>13</b>
<b>6</b>	<b>REFLECTION GROUP WORK</b>	<b>14</b>

# 1 INTRODUCTION

In order to determine thermodynamic and macroscopic behaviour of a material, it is often important to study the dynamics on a molecular level. Processes on the molecular level happen on time and length scales much smaller than what typical experimental set-ups can handle. Advancements in computational power have allowed the use of simulations to gain insight into important quantities such as pressures and temperatures at which phase transitions take place.

This paper reports what methods and approximations were taken in order to simulate an ensemble of argon molecules, and reveals conclusions and physical quantities that can be drawn from this simulation.

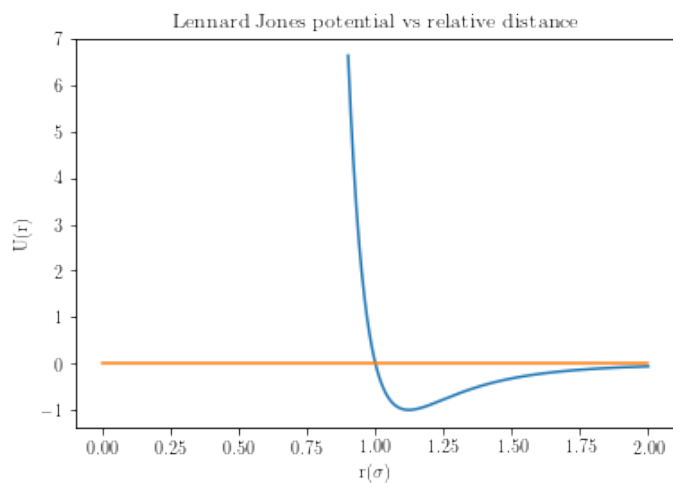
The structure of this report is as follows. First, an outline of the steps taken is presented in the 'Methods' chapter, showing how to simulate the environment and exactly how to calculate physical quantities from it. Then, the results will be presented and discussed so as to verify correct working of the simulation and compare values to those found in literature. The report concludes by providing a brief summary of the findings.

## 2 THEORY

In this project the goal is to simulate the molecular dynamics of a system of argon atoms. The only interactions which are considered in this system, are pairwise interactions between the atoms. These interactions are described by the Lennard-Jones potential, as given in equation 1

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

in which  $U$  is the interaction potential,  $\epsilon$  ( $\epsilon/k_b = 119.8 \text{ K}$  for argon) is a measure for the energy,  $r$  is the relative distance between two atoms and  $\sigma$  ( $\sigma = 3.405 \text{ \AA}$  for argon) is a characteristic length scale for this interaction. As seen in this equation, there are two terms with opposite signs. The  $+(\sigma/r)^6$ -term results in an attractive force, which will dominate for large values of  $r$ . This force is the result of dipole-dipole interactions between the argon atoms. The  $-(\sigma/r)^{12}$ -term results in a repulsive force, which dominates for small values of  $r$ . This repulsive force prevents the Argon atoms from forming molecules (as seen in nature). This behaviour is plotted in figure 1.



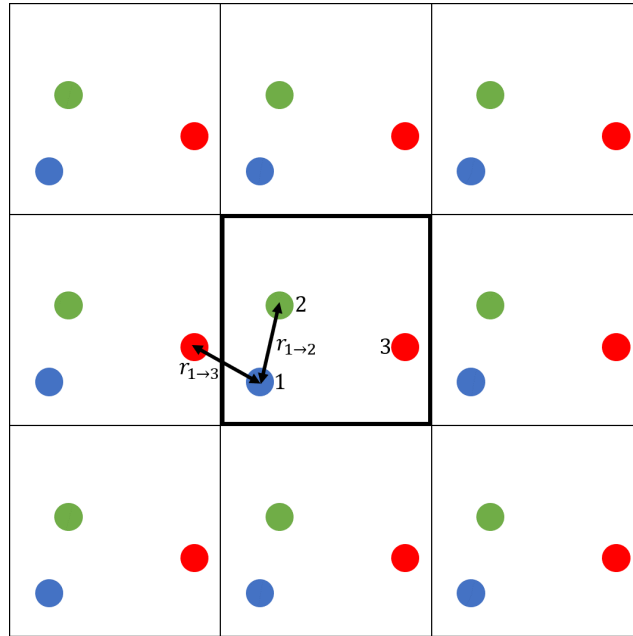
**Figure 1:** Plot of the Lennard-Jones potential vs relative distance between two atoms.  $r$  is displayed in terms of  $\sigma$  and  $U$  is displayed in terms of  $\epsilon$ . As seen from this figure, the force is strongly repulsive for  $r < \sigma$  and relatively weakly attractive for  $r > 1.2\sigma$

### 3 METHODS

This chapter discusses the methods used to make the simulation of the molecular dynamics of argon atoms. It contains a description of the formulas that are used for the simulation. Furthermore a description is given of how some macroscopic properties, such as the pressure and specific heat, can be determined from the simulation.

#### 3.1 Simulating space

To simulate sensible macroscopic valuable results, the argon atoms should be placed in infinite space with an infinite number of particles to interact with. To approximate this situation, which is impossible to simulate, a finite box with periodic boundary conditions was used. For 2D, this is illustrated in figure 2.



**Figure 2:** Particles in a 2D finite box with periodic boundary conditions. The nearest neighbours are shown via the  $r_{i \rightarrow j}$ . As seen in this example, the nearest neighbour does not need to be in the same box.

In figure 2 it is seen that the domain is copied 8 times for the 2D case. For 3D there would be 26 extra copies. Since only the physical domain is looked at (which is the middle copy), particles leaving this domain, should enter on the other side. Since all the copies of the atoms express exactly the same movement, this is automatically realised.

#### 3.2 Simulating the interaction forces

In the simulation, each particle interacts with all the other  $N - 1$  particles in the box. When first considering one particle interacting with the  $N - 1$  other particles, the total Lennard-Jones potential at a certain time  $t$  felt by this particle is calculated by summing up all the contributions for the

other  $N - 1$  atoms. However, due to the periodic boundary conditions, there is an infinite number of copies of each particle.

In this simulation, only the nearest copy is considered (as seen in figure 2). Since the atom that is looked at always resides in the box, the only copies that need to be considered for the Lennard-Jones potential are the ones inside the box with coordinates  $(x, y, z)$  and the ones with coordinates  $(x \pm L, y \pm L, z \pm L)$ . Due to the periodic nature of the simulated space, the minimisation problem of finding the nearest neighbour can be decoupled into minimising the coordinates  $(x, y, z)$  separately. For each coordinate, the following relations should hold for it to be the nearest neighbour coordinate:

$$\begin{aligned} -L/2 < x_i - x_j < L/2 &\rightarrow x_j' = x_j \\ x_i - x_j > L/2 &\rightarrow x_j' = x_j + L \\ x_i - x_j < -L/2 &\rightarrow x_j' = x_j - L \end{aligned} \quad (2)$$

The relations in equation 2 can be summarised in one line of code:  $(x_i - x_j + L/2) \% L - L/2$ , where  $\%$  is the modulo operator. This procedure is repeated for the three spatial coordinates.

### 3.3 Simulating the dynamics

For simulating the time evolution of the argon atoms, Newton's equation of motion is used:

$$m \frac{d^2 \mathbf{x}}{dt^2} = \mathbf{F}(\mathbf{x}(t)) = -\nabla U(\mathbf{x}(t)), \quad (3)$$

in which  $F$  is the net force,  $m$  is the mass of the atom and in this case  $U$  is the total Lennard-Jones potential for one atom, caused by the other  $N - 1$  atoms. Since the potential only depends on  $r = |\mathbf{r}_1 - \mathbf{r}_2|$ , the absolute distance between the atoms,  $\nabla U$  reduces to  $\frac{dU}{dr}(\mathbf{x}/r)$ . In the next sections, two methods are shown to numerically solve the equations of motion for all the particles.

#### 3.3.1 Euler's Method

A straight-forward method of discretising the dynamics of the system is dividing the continuous time  $t$  into timesteps of width  $dt$ . In the approximation that the time-step is very small, the equations of motion can be linearized, as seen in equations 4.

$$\begin{aligned} \mathbf{x}(t + dt) &= \mathbf{x}(t) + \mathbf{v} \cdot dt \\ \mathbf{v}(t + dt) &= \mathbf{v}(t) + \frac{1}{m} \mathbf{F}(\mathbf{x}(t)) dt \end{aligned} \quad (4)$$

Implementing this system of equations in the simulation and looping over each time-step in the domain results in a matrix  $x(t)$ , in which the position of all atoms at all time-steps is stored. A downside of the Euler method for integrating ordinary differential equations is that it does not conserve energy. Each time step a small error is added to the velocity, which introduces a drift of the total energy. An algorithm with conservation of energy is required since the simulation is performed in the micro-canonical ensemble (which conserves energy).

#### 3.3.2 Verlet's algorithm

A more sophisticated way of simulating the dynamics of the system is implementing Verlet's algorithm:

$$\begin{aligned}\mathbf{x}(t+dt) &= 2\mathbf{x}(t) - \mathbf{x}(t-dt) + \frac{dt^2}{2}\mathbf{F}(\mathbf{x}(t)) \\ \mathbf{v}(t+dt) &= \mathbf{v}(t) + \frac{dt}{2m}(\mathbf{F}(\mathbf{x}(t+dt)) + \mathbf{F}(\mathbf{x}(t)))\end{aligned}\tag{5}$$

With the Verlet's algorithm there is no drift of the total energy. This is a result of the time reversibility of the Verlet's algorithm [1].

### 3.4 Initial conditions

There are two components to the initial conditions that have to be taken into account: initial positions and velocity. Since argon forms an fcc lattice in the solid phase, the particles are initially put in this lattice. This kind of periodicity also prevents that particles are placed too close together (resulting in an enormous, non-physical force due to the  $-(\sigma/r)^{12}$  term in the Lennard-Jones potential).

Since the total kinetic energy of the atoms is a measure for the temperature, the initial velocity should be chosen in accordance with the temperature that is wanted for the system. To fulfil this requirement the velocities of the particles should be distributed with a Maxwell distribution. This requirement gives the following probability density function for the velocities:

$$P(v_{x,y,z}) = \frac{m}{\sqrt{2\pi k_b T}} e^{\frac{-mv_{x,y,z}^2}{2k_b T}}\tag{6}$$

Although the x,y and z components of the velocity are zero-mean, the total momentum of the system is not necessarily zero. To prevent the total system from drifting, the total momentum is subtracted from the system. This does not effect the temperature, since the standard deviation of the velocity-components (which is a measure for the temperature) stays the same.

### 3.5 Velocity correction

When the simulations starts, the system is set to have a certain temperature. However, when the simulations is running the system will evolve to an equilibrium state. During this process there will be an exchange between kinetic and potential energy. Therefore the temperature of the system will change during the simulation, since the kinetic energy is directly related to the temperature. To ensure that the system remains close to the initial temperature the velocities of the particles are rescaled when the system is relaxed to equilibrium. This rescaling of the velocities is performed a number of times until the temperature is converged to the initial temperature.

The time required to let the system come to equilibrium is the mean free path length divided by the average speed (such that on average, every particle has had approximately one interaction).

$$l = \frac{1}{\pi(2\sigma\rho)^2}\tag{7}$$

The equation used for the velocity rescaling, as found in the book of Jos Thijssen [1], is displayed in equation 8



$$v_{scaled} = v_{old} \sqrt{\frac{(N-1)3k_bT}{\sum_i m v_i^2}} \quad (8)$$

### 3.6 Macroscopic properties

To be able to compare our simulation to reality or other simulations, it is necessary to compute several macroscopic quantities. In this report, pressure, specific heat, total energy and the pair-correlation function are computed.

#### 3.6.1 Specific heat

For computing the specific heat ( $C_v$ ), two different methods are used. The first method is direct computation of the specific heat. This is done by varying the temperature around the set temperature ( $\pm 10\%$ ) and computing the total energy per particle for each temperature. In this simulation, 10 different T-values are used. In the approximation that  $U(T) \approx C_v T + c$ , such that  $U(T)$  is linear, the  $C_v$  is easily computed by obtaining the slope of this function, using the scipy curve fitting tool. The error in  $C_v$  is found by computing the error in the slope of the fit.

In the second method, a relation is used between the specific heat and fluctuations in the kinetic energy. This dependence is shown in equation 9 [1]

$$\frac{\langle \delta K^2 \rangle}{\langle K \rangle^2} = \frac{2}{3N} \left( 1 - \frac{3N}{2C_v} \right), \quad (9)$$

in which  $\langle \dots \rangle$  denotes the time-average,  $K$  and  $\delta K$  are the kinetic energy and the fluctuations in the kinetic energy respectively and  $N$  is the total number of particles in the system. The error in this case is computed by the method of bootstrapping (as explained in section 3.8).

Note that both ways of computing the specific heat, result in a specific heat that is independent of the number of particles ( $C_v$  is the specific heat per particle). 'Normalising' the specific heat to the number of particles in this way is convenient when comparing the results to the literature (without having to worry about the number of particles in the simulation).

#### 3.6.2 Pressure

The pressure can be calculated using the virial theorem [1]:

$$\frac{\beta P}{n} = 1 - \frac{\beta}{3N} \left\langle \sum_{i=1}^N \mathbf{r}_i \nabla_i V_N(R) \right\rangle \quad (10)$$

where  $P$  is the pressure,  $n$  the density,  $\beta = \frac{1}{k_b T}$  with  $k_b$  the Boltzmann constant and  $T$  the temperature,  $N$  the number of particles,  $\mathbf{r}$  the particle position vector, and  $-\nabla V_N(R)$  the force acting on the particle. The  $\langle \dots \rangle$  denotes the time average of the quantity between brackets [1].

### 3.7 Pair Correlation function

Phase transitions in the simulation are hard to quantify. One quantity that gives insight into phase transition (as well as other physical quantities) is the correlation function. The pair correlation shows how the distribution of other particles depends on the location of a single particle. This pair correlation function strongly depends on the phase of the system. For a solid system in a lattice, there will be distinct peaks in the function. For a gas, the location of other particles will be uncorrelated for larger distances and the correlation function should converge to unity. Computation of the correlation is done by keeping a histogram of all relative particle positions starting from when the system is stable and averaging this histogram. Then from this histogram  $n(r)$  the pair-correlation  $g(r)$  can be computed as [1]:

$$g(r) = \frac{2V}{N(N-1)} \frac{n(r)}{4\pi r^2 \Delta r} \quad (11)$$

where  $\Delta r$  is the bin size of the histogram,  $N$  is the total number of particles and  $V$  is the volume of the box.

### 3.8 Bootstrapping

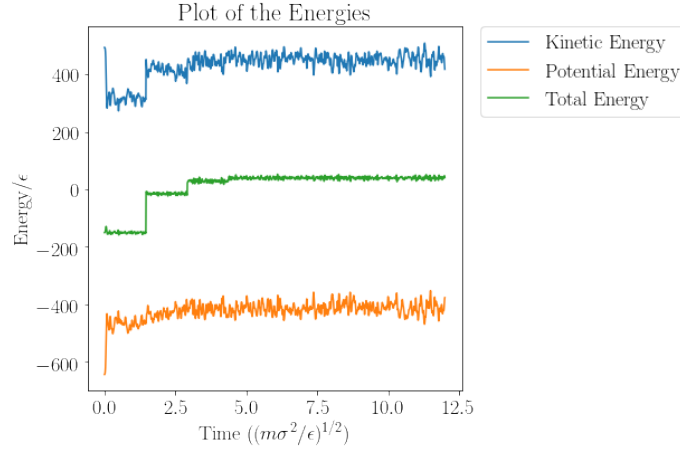
The errors in the temperature, pressure and value of specific heat computed using equation 9 in this simulation will be computed using bootstrapping. Bootstrapping can be applied to any physical quantity that is computed through a statistical property of the data (such as the mean or variance of the samples). The data will be resampled and computed several times. The resulting variance of these randomly sampled data sets is then a good estimate of the physical quantity's variance. However, the resulting uncertainties in this simulation should be taken with a grain of salt. It may give a good impression of the uncertainty of the physical quantity for a certain realisation of the simulation. But different realisation may give significantly different results. Despite the temperature re-scaling algorithm, the realised temperature may differ from the temperature that was set, which can have a significant impact on the results, especially around phase transitions.

## 4 RESULTS AND DISCUSSION

This chapter contains the results of macroscopic properties that are computed from the simulation. The result of the simulation are compared with values from literature for comparison.

### 4.1 Energy conservation

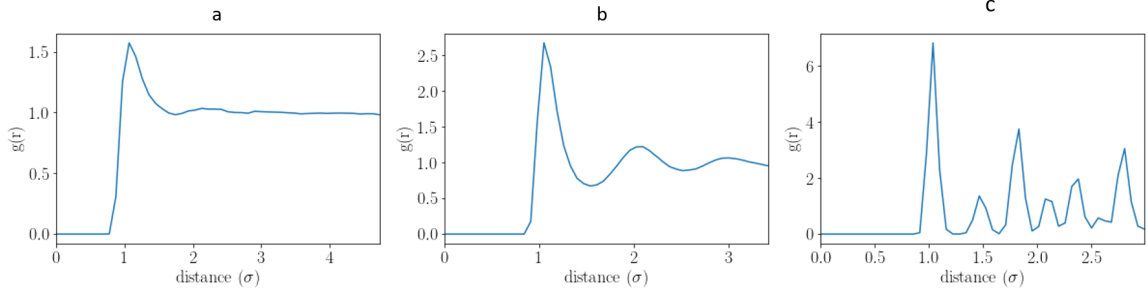
A plot of the kinetic energy, potential energy, as well as the total energy of the system is shown in figure 3. The figure shows energy jumps in time. These jumps are a consequence of the velocity re-scaling. This re-scaling has as effect that magnitude of the velocity suddenly changes and hence the total energy of the system changes. From the figure it can be seen that energy is conserved well. There are slight variations in the total but these are small compared to the variations in kinetic and potential energy, and there is no energy drift.



**Figure 3:** Kinetic energy, potential energy, and total energy versus time for a system of 256 particles and  $T=1$ .

### 4.2 Pair Correlation function

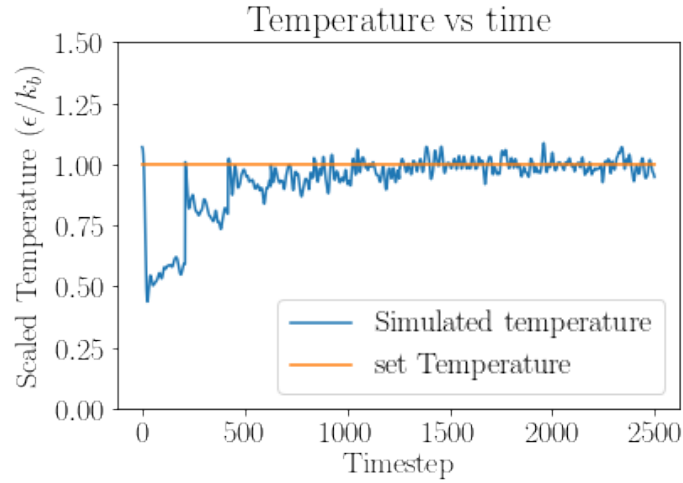
In order to gain insight into phase transitions, the pair correlation function of the system has been plotted for the gas, liquid, and solid phase respectively in Figures 4 a, b and c. The results are in line with what is expected. The peaks in the solid phase are notably higher than those in the other phases (note the scale of the images), because the particles are in a roster where they do not deviate too much from their positions. The correlation function goes to 1 in the gas phase for larger distances, but is still zero for very small distances due to the repulsive force from the Lennard-Jones potential. It appears that even in the gas phase, particles that approach each other interact and are slowed down. Therefore, they are more likely to be at a distance of approximately  $\sigma$  from each other. The liquid phase shows a behaviour that is in between that of the solid phase and the gas phase, in that it is non-zero for all distances larger than sigma but has noticeable peaks too.



**Figure 4:** Pair correlation functions computed for a) the gas phase ( $\rho = 0.3$ ,  $T = 3$ ), b) the liquid phase ( $\rho = 0.8$ ,  $T = 1$ ) and c) the solid phase ( $\rho = 1.2$ ,  $T = 0.5$ ).

### 4.3 Temperature

To verify the temperature stabilisation in the system, a plot of temperature vs time is shown in figure 5.

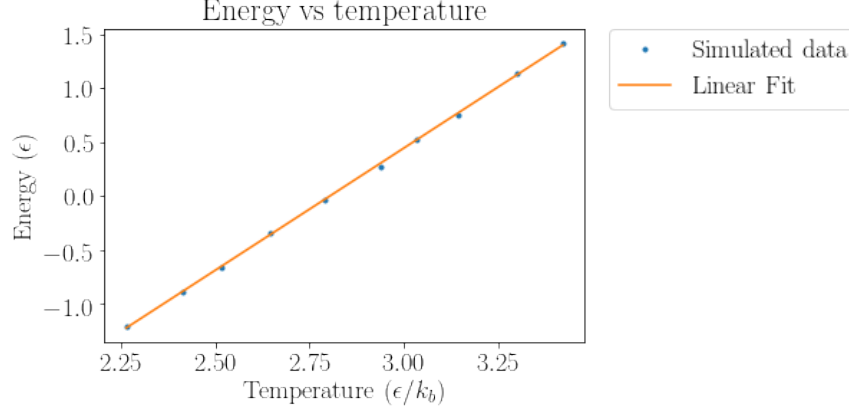


**Figure 5:** Plot of the temperature vs time for 256 atoms. The temperature correction behaves as expected and results in the correct set temperature.

As seen in figure 5, the temperature stabilises quickly towards the set temperature.

## 4.4 Specific heat

In case of the method of acquiring the specific heat ( $C_v$ ) by direct computation, the assumption is made that the total energy per particle ( $U_{pp}$ ) depends linearly on temperature ( $T$ ). The slope of this linear dependence is  $C_v$ . As seen in figure 6, this dependence is indeed linear. Thus, the approximation is valid (as reflected later on in the fit errors).



**Figure 6:** Example of a linear fit of the  $C_V$  for  $T = 2.89$ ,  $\rho = 0.85$ . Qualitatively it is clear that the approximation of a linear dependence is indeed valid. The small errors in the fit parameters (table 1) quantitatively confirm this.

For the two methods of computing specific heat, the results are given in table 1.

**Table 1:** Results of the two methods of computing the specific heat and the corresponding errors.  $C_{v,fluc}$  is computed with equation 9, while  $C_{v,direct}$  is computed by direct computation. In these simulations 256 particles were simulated for 5000 timesteps. For fitting  $C_{v,direct}$ , 10 different temperatures were used.

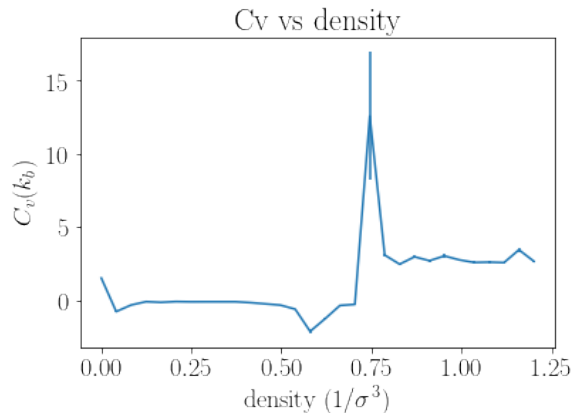
$T(\epsilon/k_b)$	$T_{sim}$	$\rho(1/\sigma^3)$	$C_{v,direct}(k_b)$	$C_{v,fluc}(k_b)$
2.89	2.87	0.85	$2.22 \pm 0.04$	$2.13 \pm 0.03$
1.13	1.14	0.85	$2.50 \pm 0.07$	$1.93 \pm 0.05$
2.84	2.82	0.75	$2.06 \pm 0.03$	$1.94 \pm 0.02$
4.62	4.52	0.45	$1.72 \pm 0.02$	$1.742 \pm 0.007$

Most of these results are very similar to the simulations of Lebowitz and Verlet [2]. Furthermore, the two methods of computing the specific heat give comparable results for all combinations of  $T$  and  $\rho$ , except for the case where  $T = 1.13$  and  $\rho = 0.85$ . It is suspected that this is the case, because this combination gives a state of the system close to a phase transition/liquid phase, a lot of fluctuations. Therefore the uncertainty of the simulation is much higher than the error found due to the bootstrapping method. Due to the random initial conditions, the simulation gives a very different result when running it multiple times in this combination of  $T$  and  $\rho$ , since the temperature is stabilised at a slightly different value every realisation.

When comparing the two methods, it is concluded that in most cases, it is preferable to use the method based on the fluctuations, since the computation time is way shorter. This is because direct computation requires at least three realisations of the whole simulation (preferably more). However, as seen before, in some cases the 'fluctuation-method' does not compute an accurate result (for example near phase transitions). In these situations, direct computation is preferable, due to the increased accuracy.

From table 1, it is seen that the specific heat in our simulation behaves as one would expect. It

increases for increasing density. Figure 7 reflects this behaviour as well, in a plot of the specific heat vs the density (in the case of  $T = 0.5$ ). The first notable detail in this graph is the high peak



**Figure 7:** Plot of the  $C_v$  vs density for  $T = 0.5$ , for 256 particles with errorbars. The plot was computed using the fluctuations in kinetic energy.

around  $\rho = 0.75$ , with a significant errorbar. This spike in  $C_v$  is explained by a phase transition. In a phase transition, energy added to the system does not increase the temperature, but keeps the phase transition going, resulting in a high value for  $C_v$ . The dip just before the peak (resulting in a negative value for  $C_v$ ), is not so easy to predict. In some rare cases  $C_v$  becomes negative for certain atomic clusters near a phase transition [3]. This could be an explanation, although our simulation in principle simulates an 'infinite number of particles'. Therefore, it is more likely that the approximations made in this simulation do not apply to every regime in the plot.

In the limit of an ideal gas (so for very low densities/high temperatures), the specific heat in the simulation should behave according to the equipartition theorem [1], which states that  $K = 3(N - 1)k_bT/2$ . Since for an ideal gas, there are no interactions,  $U_{pot} = 0$ , so  $C_v = \frac{\partial U}{\partial T}|_V = \frac{\partial K}{\partial T}|_V = 3(N - 1)k_b/2$ . Since our units are dimensionless and the  $C_v$  is normalised to the number of particles, in the limit of  $N \gg 1$ ,  $C_v \approx 3/2$ . Indeed our simulation shows a  $C_v$  of 1.50 for  $\rho = 0.005$  and  $T = 10$ , so it gives the expected results in the limit of an ideal gas.

## 4.5 Pressure and total internal energy

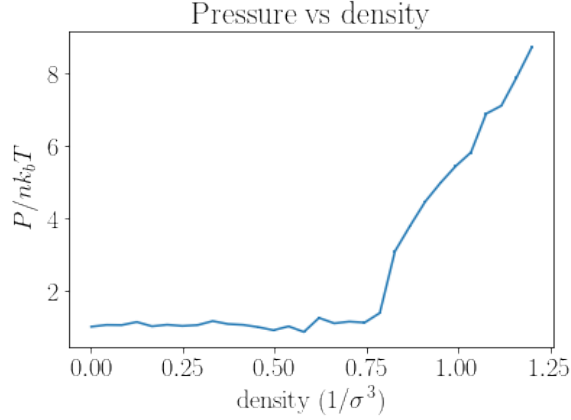
**Table 2:** Results of the computed pressure and internal energy and corresponding errors. In these simulations 256 particles were simulated for 10 000 timesteps.

$T(\epsilon/k_b)$	$T_{sim}$	$\rho(1/\sigma^3)$	$\beta P/n$	$U(\epsilon)$
1.0	0.98	0.90	$2.79 \pm 0.01$	$-6.3313 \pm 0.0009$
	0.99	0.88	$1.11 \pm 0.02$	$-5.828 \pm 0.002$
	0.99	0.80	$1.04 \pm 0.01$	$-5.4453 \pm 0.0005$
	1.00	0.75	$1.00 \pm 0.01$	$-4.8182 \pm 0.0006$

When comparing table 2 to the results found in the book by Jos Thijssen [1], it is seen that the results are of the same order of magnitude. The pressure increases with increasing density as expected.

Furthermore for low densities, the pressure should approach 1, since the virial approaches zero. In

equation 10 it is seen that the second term vanishes in that situation, resulting in a pressure of 1. Furthermore, figure 8 shows an increasing pressure for increasing density as well. Furthermore, in



**Figure 8:** Plot of the pressure vs density for  $T = 0.5$  and 256 particles with errorbars.

case of an ideal gas, the quantity  $P\beta/\rho$  should be 1, as follows from the ideal gas law, which is observed in figure 8. The sudden increase in pressure around  $\rho = 0.75$  could be explained from a phase transition (which is also seen in figure 7).

The discrepancies between the book [1] and the simulation are because the simulation is very sensitive to input parameters. As seen in table 2, there is a significant difference in pressure between  $\rho = 0.88$  and  $\rho = 0.9$  (which is suspectedly due to a phase transition).

Lastly, the internal energy increases with decreasing density, since the potential energy increases (and gets closer to 0, since it is negative) due to the increased distance between the atoms (which can be understood from figure 1). Note that the uncertainties in the energy per particle are very small (table 2), meaning energy conservation is satisfied to a great extent within one realisation.

## 4.6 Performance

For the computations it is important that the algorithm has a high efficiency, meaning that the time required to do the simulations is as short as possible. To achieve this goal the algorithm uses the vectorisation together with the numpy scientific computing package when possible. For loops are used by looping over the time and looping over different simulations.

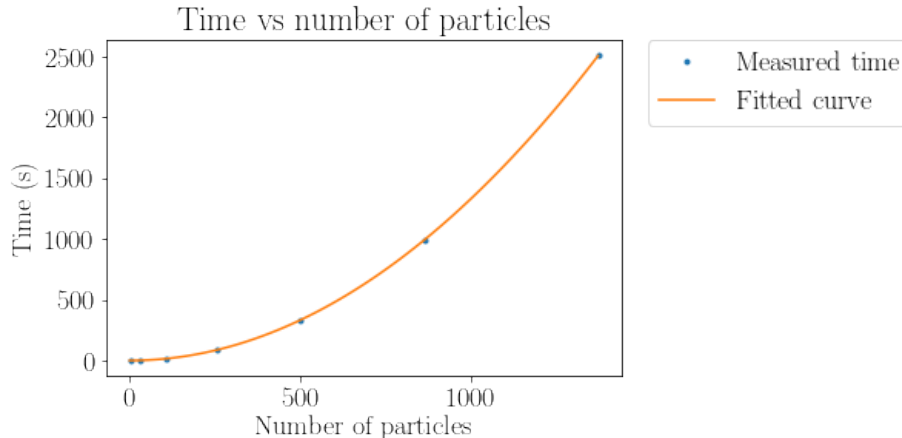
For validation of the efficiency of the code the simulation is performed for different number of particles and the running time of the simulation is computed. The number of particles is chosen such that it is an integer number of unit cells within a cubic volume. Each simulation is performed performed for 3000 time steps.

The result for the simulation are shown in the figure 9. From the figure it can be seen that the simulation time is quadratic depended on the number of particles. This quadratic dependency on the number of particles is expected since the computational intensive calculations are the minimal distance calculation between the particles and potential energy calculation, which both scale with number of particles squared.

A quadratic formula is fitted to the found simulations times. The formula found can be seen below:

$$T_{simulation} = 1.32 \cdot 10^{-3} N^2 \quad (12)$$

where  $T_{simulation}$  is the time required to do the simulation in seconds, and  $N$  is the number of particles. The R-value of the fitted regression line is 1.



**Figure 9:** Plot of the simulation time versus the number of particles, together with the fitted curve to the data points

## 5 CONCLUSION

In this report, a bottom-up simulation of interacting Argon atoms is shown. Various properties and macroscopic quantities are simulated and behave as expected in general.

The pressure increases with density and has the same order of magnitude as found literature values [1]. Discrepancies are suspected to be caused by small errors in the input parameters, which result in huge uncertainties near phase transitions. The simulated values for the specific heat, computed using direct and indirect methods correspond to a reasonable extend with each other and the literature [2]. Also the computed internal energy corresponds with literature values.

In case of these three physical quantities, the error found by a bootstrapping method indicated very small uncertainties within one realisation. However, due to uncertainties in input parameters, (such as temperature, which is slightly different every realisation), the actual uncertainties in the output is larger than these bootstrap-errors. To estimate the error between different realisations, one could perform the simulation multiple times for the same input parameters and compute the standard deviation (which could be done in further research).

Furthermore, phase-transitions are inspected by computing the pair-correlation function. As expected, this function shows distinct peaks for the solid phase (where there is strong correlation between the atoms placed in a periodic lattice). The gas-phase resulted in a near-uniform pair correlation function, due to the chaotic and uncorrelated positioning of the atoms. The liquid phase shows characteristics of both gas and solid states of the system as expected.

Lastly, the numerical performance of the simulation is high, because of the extensive use of the numpy library. Almost no for-loops are used (which slow down simulations in python dramatically) and all functions are optimised for large simulations. As expected, the simulation time scales as  $N^2$ .



## 6 REFLECTION GROUP WORK

In this study, we worked together as a group of three students. The bulk of the programming, we performed together, as one of us coded and the other two brought up ideas for solving the problems we ran into along the way. We varied these roles frequently, such that everyone got to do a lot of everything. Some parts were easily decoupled and distributed over the team, such that we could work on these individually (such as cleaning up the messy script into well organised functions, writing functions for calculating macroscopic properties, creating an automatic saving system and so on). Also in this case there was clear communication as to what problems were encountered, such that we could all bring in ideas for every separated part.

Throughout the project, the work-load was evenly distributed. This included the last week, in which the report had to be written and the code had to be cleaned up. In writing the report, the following distribution was made:

**Martijn:** Methods (Dynamics, Initial conditions, Velocity correction, pressure), Results (Performance, created system for saving results).

**Cyrus:** Introduction, Abstract, Methods (Pair correlation, Bootstrapping), Results (Energy conservation, Pair correlation).

**Richard:** Theory, Methods (Space, Forces, Specific heat), Results (Temperature, Specific heat, Pressure/Internal energy), Conclusion.

All in all, the collaboration was well-coordinated and the workload was evenly distributed.

## References

- [1] J. Thijssen. *Computational Physics*. Cambridge University Press, 2007.
- [2] J. L. Lebowitz, J. K. Percus, and L. Verlet. Ensemble dependence of fluctuations with application to machine computations. *Phys. Rev.*, 153:250–254, Jan 1967.
- [3] Martin Schmidt, Robert Kusche, Thomas Hippler, Jörn Donges, Werner Kronmüller, Bernd von Issendorff, and Hellmut Haberland. Negative heat capacity for a cluster of 147 sodium atoms. *Phys. Rev. Lett.*, 86:1191–1194, Feb 2001.