

Project1 - FYS4460

Mikkel Metzsch Jensen
(Dated: March 14, 2021)

a) Maxwell Boltzman distribution

We simulate a $15 \times 15 \times 15$ Lennard Jones system with face-centered-cubic packing resulting in 13500 particles. We use temperature $T' = 2.5 \epsilon/k_B$ and simulate for 30000 timesteps with default timestep length 0.005τ . See script *a.in* for more details. We then looked at the velocity distribution for the last timestep in form of a histogram shown in figure 1.

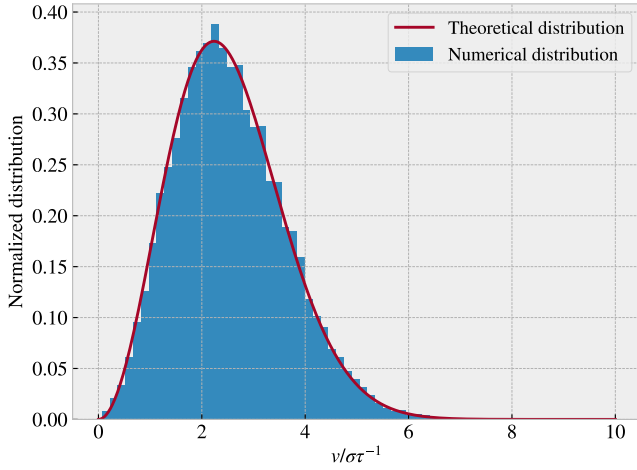


FIG. 1. Velocity distribution histogram for the final time-step t_n in the simulation. The numerical results for the simulation matches nicely with the theoretically Maxwell-Boltzman distribution given as $f(v) = \sqrt{\left(\frac{m}{2\pi kT}\right)^3} 4\pi v^2 e^{-\frac{mv^2}{2kT}}$.

We see that the final velocity distribution found in the simulation matches quite well with the theoretical Maxwell-Boltzmann distribution. In order to examine how fast this distribution is reached we calculate the inner product given as:

$$\frac{\sum_i h_i(t) h_i(t_n)}{\sum_i h_i(t_n)^2} \quad (1)$$

where h_i is the height of bin i in the histogram and t_n is the final time-step. By calculating 1 as a function of time we get the result shown in figure 2. As a quick estimate we can say that the distribution is reached after approximately 30τ when the inner product has roughly converted to 1. Since the inner product stays around 1 we can conclude that the distribution shown in 1 is indeed the steady state of this system.

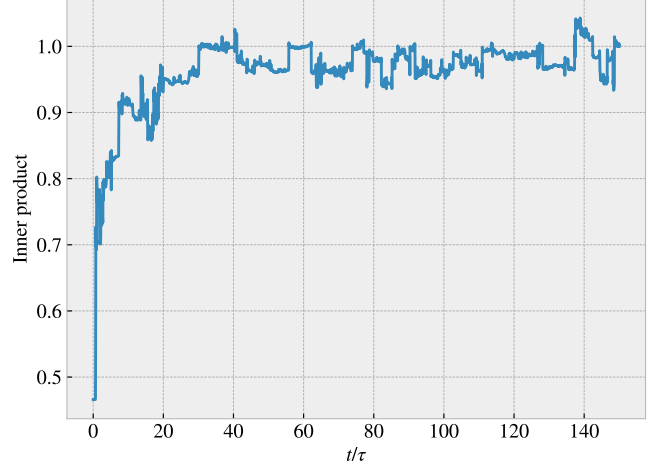


FIG. 2. Inner product as shown in equation 1. We see that it converges towards approximately 1 indicating that it have reached the distribution from 1.

b) Total energy

We use similar simulation settings as in a) (see script *b.in*), where we now output the total energy with the fix ave/time command. We look at the development of the total energy over time for different timesteps as showed in figure 3.

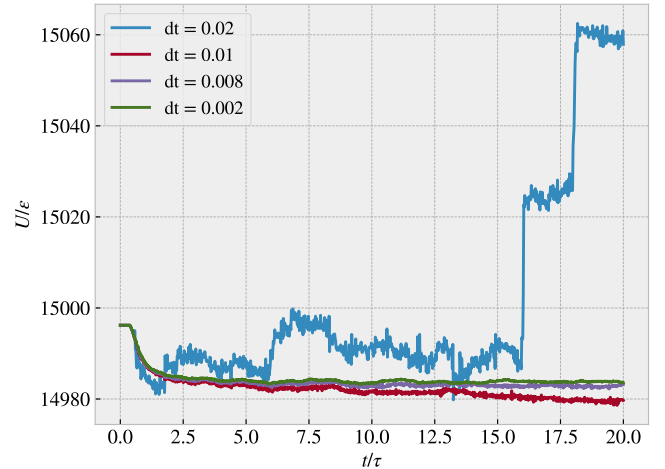


FIG. 3. Total energy over time for different timesteps dt .

We see as expected that the energy fluctuations is qualitatively bigger for bigger dt . We also notice that energy drops over the first period of the simulation. This is prob-

ably due to the system stabilization taken place. I'm not completely sure why this happens at NVE, where I would expect the energy to remain constant and instead having temperature fluctuations.

c) Temperature

We now consider the temperature in the simulation. We reuse the settings from script *b.in* in the modified version in script *c.in*. We use the equipartition principle:

$$\langle E_k \rangle = \frac{3}{2} N k_b T \quad (2)$$

where E_k is the kinetic energy, N the number of particles, k_b Boltzmann's constant and T is the temperature. By using this relation we find the temperature in the simulations for different dt . The result is showed in figure 4

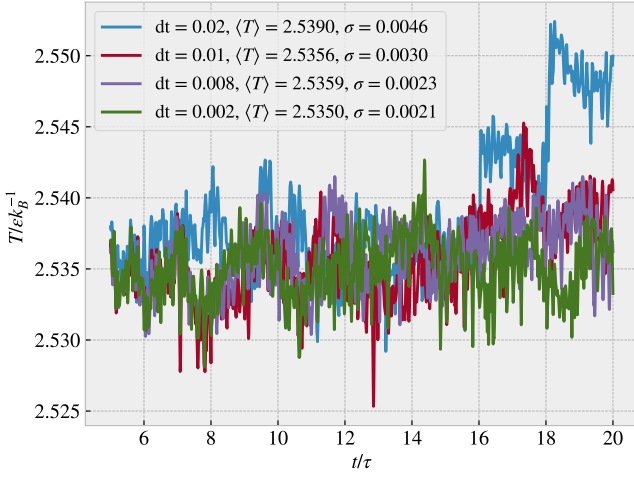


FIG. 4. Estimated temperature using equation 2 over time for different timesteps dt . Notice that the first period of time is excluded due to the equilibration of the system.

We initialized the simulations with temperature $T = 2.5$ which corresponds quite good to the mean value found here. All though we see a general offset of ≈ 0.035 compared to the initialized value of 2.5. This is due to the way that LAMMPS set the initial velocity distribution in order to reach the target temperature. In this process it does not take the potential energy into account which depends on the distance between the atoms at the initial state. When the simulation starts, the energy distribution with respect to kinetic and potential energy is redistributed such that it might not hit the target value exactly. The size and direction of the offset will therefore vary for different packing style and density.

By looking at the standard deviation σ , we see that the fluctuations is generally lower with decreasing dt . In order to see how the fluctuations depends on the system size we also ran a series of simulations with fixed $dt =$

0.002 but with increasing system size. This is shown in figure 5

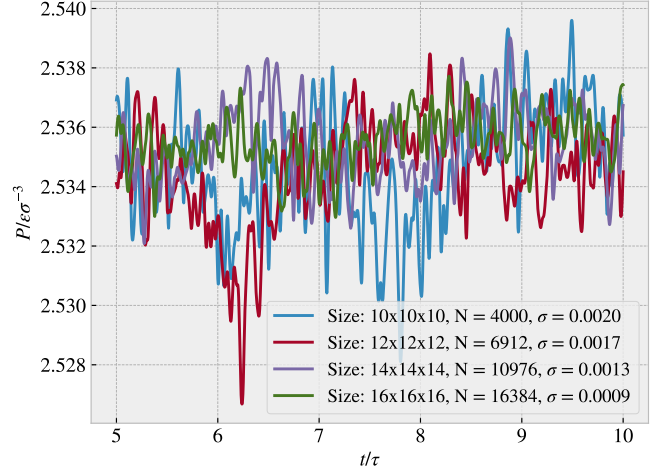


FIG. 5. Estimated temperature using equation 2 over time for different system sizes $a \times b \times c$ measured in number of unit cells. Each unit cell contains 4 atoms, which gives a total of $N = 4abc$ atoms.

We see that the fluctuations generally gets smaller for bigger systems.

d) Pressure as function of temperature

We now perform a series of simulations at different temperature, let it stabilize, and then collect the average values for the pressure $P(T)$ as a function of average temperature T . See script *d.in* and *d.py* for more info. By plotting the pressure $P(T)$ with a linear fit we get the result showed in figure 6

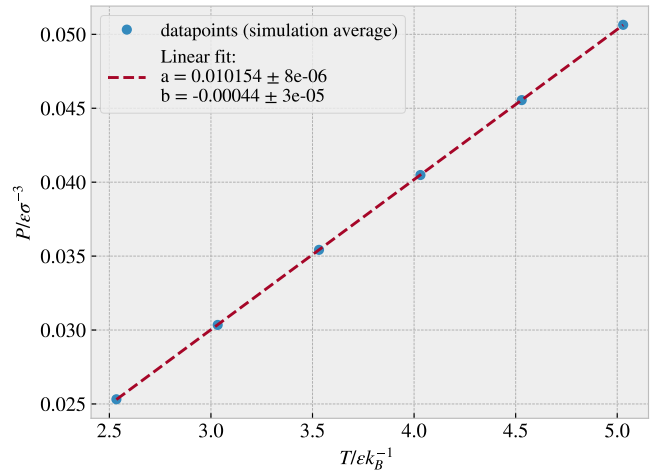


FIG. 6. Average measurements of temperature and pressure in stabilized simulations. The linear fit confirms the proportionality between pressure and temperature.

We see that the result fit nicely to a linear model which is consistent with both the ideal gas law and also the Van der Waals equation which we will address in the next exercise.

e) Pressure as function of both temperature and density

We now measure the pressure as a function of both temperature and density. We make a grid of values with 10 temperature points in the interval $T \in [1, 10]$ and 10 density points $\rho \in [0.02, 0.38]$ and simulate 100 times according to all the possible combination of these values. See script *e.in* and *e.py* for more info. We then consider two models to predict the relationship between pressure, temperature and density. The first model is ideal gas law:

$$PV = Nk_B T$$

Since we have $k_B = 1$ in our simulations this model predicts a good linear fit between PV and T with a slope of N . The second model to consider (which should be the one in use here) is the Van der Waals equation:

$$(P + a \frac{N^2}{V^2})(V - Nb) = NkT$$

From this we expect to see a linear fit between $(P + a \frac{N^2}{V^2})(V - Nb)$ and T with slope N .

With the data from the simulations we try to fit the data to both of these models by performing linear regression with a fixed slope of N . For the Van der Waals model we use a python script to choose parameters a and b such that the MSE of the fits gets minimized. See script *e.py* for more details. The results are shown in figure 7 and 8.

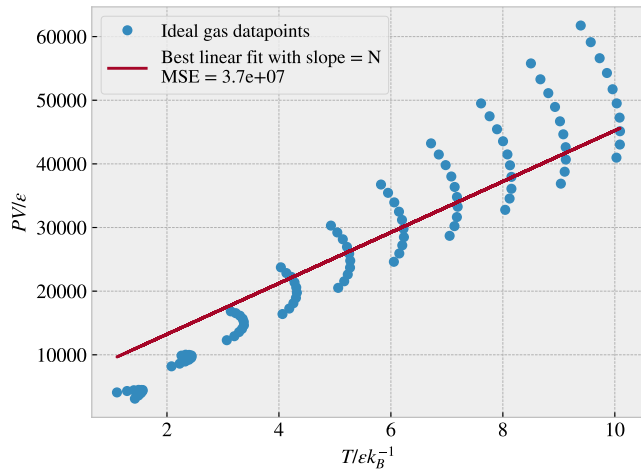


FIG. 7. Linear fit for ideal gas relation.

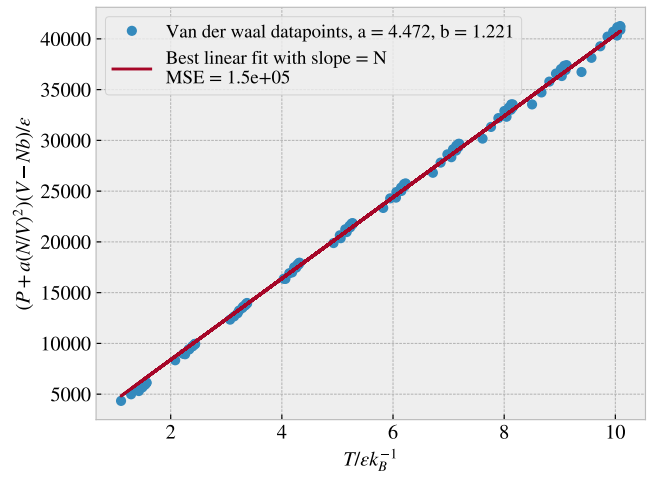


FIG. 8. Linear fit for Van der Waals relation.

We see that the van der waals fit is clearly the better one here, and we estimate the parameters to be $a = 4.472$ and $b = 1.221$

f) Diffusion - mean square displacement

We now look at the diffusion in the liquid phase of the Argon model. We measure the mean square displacement $\langle r^2(t) \rangle$ as a function of time for different temperatures. We use density $\rho = 0.8$ and temperatures $T = 0.9, 1.0, 1.1, 1.2$. See script *f.in* and *f.py* for more info. The results are shown in figure 9. By finding the slope $\langle r^2(t) \rangle / t$ we can estimate the diffusion constant through the theoretical relation:

$$\langle r^2(t) \rangle = 6Dt, \quad \text{when } t \rightarrow \infty \quad (3)$$

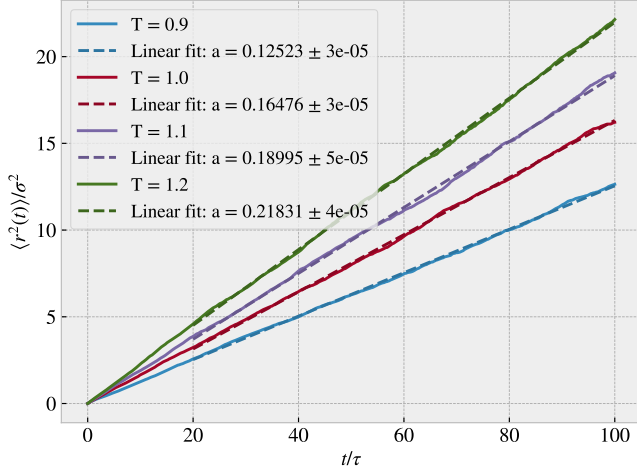


FIG. 9. Measure of the mean square displacement $\langle r^2(t) \rangle$ as a function of time t for different temperatures T in the liquid phase of the Argon model. For this we used density $\rho = 0.8$. The slope estimates the diffusion constant D as $6D$ according to equation 3

The diffusion constants estimated from figure 9 is listed in table I

TABLE I. Estimated diffusion constants from figure 9

Temperature [$T/\epsilon k_B^{-1}$]	Diffusion constant [σ^2/τ]
0.9	$0.02087 \pm 3e-5$
1.0	$0.02746 \pm 3e-5$
1.1	$0.03166 \pm 5e-5$
1.2	$0.03638 \pm 4e-5$

We then plot the diffusion constant as a function of temperature as shown in figure 10.

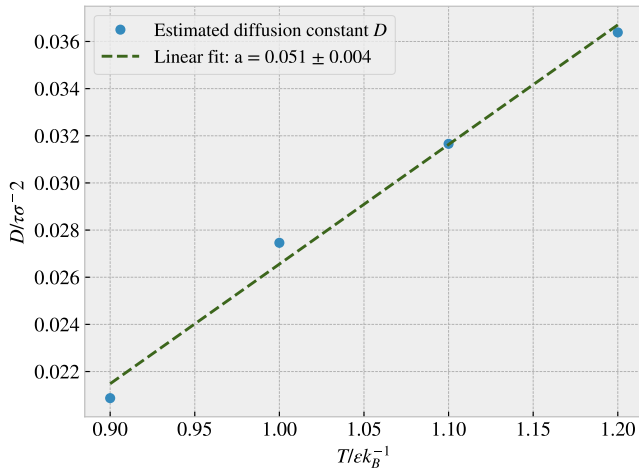


FIG. 10. The diffusion constant values from table I plotted against the temperature used.

From these datapoints it appears that the diffusion

constant is proportional to the temperature with proportionality constant $a = 0.051 \pm 0.004$. This can be related to the Stokes-Einstein equation:

$$D(t) = \frac{kT}{6\pi\mu r}$$

where μ is the solvent viscosity, and r is the radius of the diffusing particle.

Another thing to notice is the behaviour of the mean square displacement measurements during the first few timesteps, which is highlighted in figure 11.

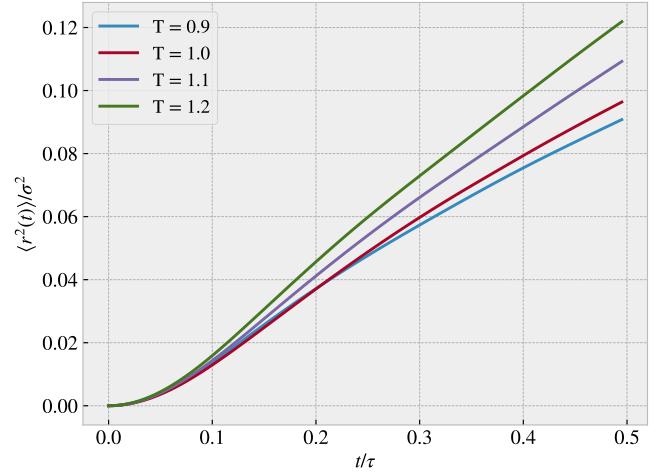


FIG. 11. A zoom on the beginning of figure 9 which shows a quadratic development before going into the linear relation as discussed.

The quadratic development of MSD can be explained by the fact that the atoms move as $\vec{r}(t) = \vec{v}_0 t + \vec{r}_0$ when far away from other atoms. In the beginning of the displacement measurements the atoms travel uninterrupted for a short period of time and therefore the displacement goes as $r^2 \propto v_0^2 t^2$. This is well showcased in the beginning of figure 11.

g) Radial distribution function (rdf)

We simulate two system corresponding to a argon in liquid phase and solid phase respectively. In both cases we let the system stabilize for 10000 timesteps and then compute the rdf as an average over another 10000 steps. Here we used $dt = 0.005$ and $T = 1.0$. For the liquid phase we used $\rho = 0.8$ and for the solid phase we used $\rho = 1.2$. We then compute the radial distribution function $g(r)$ in LAMMPS. See script *g.in* and *g.py* for more info. The results are showed in figure 12

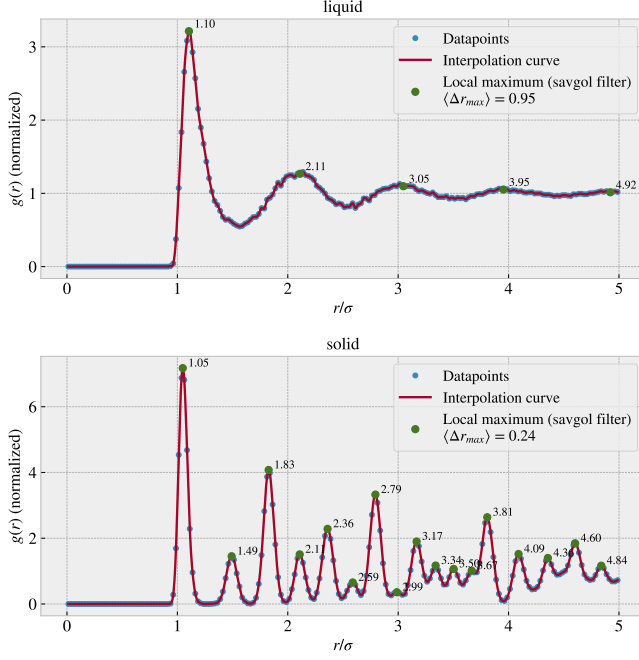


FIG. 12. Radial distribution for argon in liquid ($\rho = 0.8$) and solid ($\rho = 1.2$) phase. We used a temperature of $T = 1.0$ in both simulation and calculate the average distribution over 10000 timesteps with $dt = 0.005$ after stabilization for 10000 timesteps first.

As indicated on the plot we get distinct peaks in the radial distribution at certain distances. We see that the peaks are more sharp and narrow for the solid phase than the liquid. This makes perfectly sense since the atoms in the solid organizes itself in a crystal lattice to a greater extent than the liquid. If we take the average difference between these peaks (not including the distance from the main atom to the first peak) we get the averages shown in the figure which is as follows:

$$\begin{aligned}\langle \Delta r_{max} \rangle_{liquid} &= 0.95 \\ \langle \Delta r_{max} \rangle_{solid} &= 0.24\end{aligned}$$

From this we see that the atoms in the solid are packed more closely together, as expected when comparing the solid and liquid phase of argon.

j) Identifying differences and similarities between Stillinger-weber and Lennard Jones potential

We look at the two-particle part of the Stillinger-Weber potential V_2 . When ignoring the cut-off function we see that this is almost on the same form as the Lennard Jones potential $U(r)$:

$$\begin{aligned}V_2(r_{ij})_{(no\ cut-off)} &= A_{ij}\epsilon_{ij} \left[B_{ij} \left(\frac{\sigma_{ij}}{r_{ij}} \right)^{p_{ij}} - \left(\frac{\sigma_{ij}}{r_{ij}} \right)^{q_{ij}} \right] \\ U(r) &= 4\epsilon \left[\left(\frac{\sigma}{r} \right)^{12} - \left(\frac{\sigma}{r} \right)^6 \right]\end{aligned}$$

From the Lennard Jones potential we have the corresponding Stillinger-Weber parameters $A_{ij} = 4$, $B_{ij} = 1$, $p_{ij} = 12$, $q_{ij} = 6$. From the Si.sw file we read that these parameters are set to

$$A = 7.049556277 \quad B = 0.6022245584 \quad p = 4.0 \quad q = 0.0$$

Here we see the most significant difference in p and q which determines the strength of the Pauli repulsion (p) and Coulomb attraction (q) respectively. Thus we have a minimal strength of the Coulomb attraction and also a heavily reduced strength of the Pauli repulsion, which is further reduced by $B < 1$. The difference in A is not really significant.

k) Simulating three states of Si with Stillinger-Weber potential

We now want to visualize the simulations of the Si system in solid, liquid and gas states. By a quick look on Wikipedia: <https://en.wikipedia.org/wiki/Silicon> we find that the melting point of silicon is $T_{melting} = 1687$ K and the boiling point is $T_{boiling} = 3538$ K. We therefore run the simulations on temperatures: 400 K, 3000 K and 8000 K to achieve a simulation in solid, liquid and gas states respectively. See script *k.in* for more info. We visualize the final frame of our simulations in Ovito shown in figure 13.

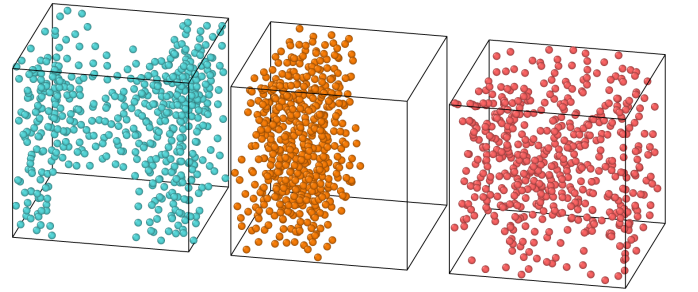


FIG. 13. Three different states of Si simulated with the Stillinger-Weber potential. From the left we have Si in a solid state (blue) at $T = 400$ K, a liquid state (orange) at $T = 3000$ K and a gas state at $T = 8000$ K

On the figure we see that the solid arranges itself in a more detailed defined structure with clusters of atoms and empty space in between. The liquid also arranges itself in a cluster but not with any distinct gaps and holes. When looking at the simulation over multiple timesteps we see that the atoms in the solid simulation remain fixed in its initial area while the atoms in the liquid simulation wander slowly. In the gas state we see that the atoms are more evenly distributed in the simulation box, and when looking at multiple timesteps, we see a great mixing of the atoms as expected.

l) Determining diffusion constant $D(T)$ as a function of temperature

We now run similar simulations as in f) but with the Stillinger-Weber potential. We find the diffusion constant as a function of temperature using the mean square displacement. See script *l.in* and *l.py* for more info. The result is shown in figure 14.

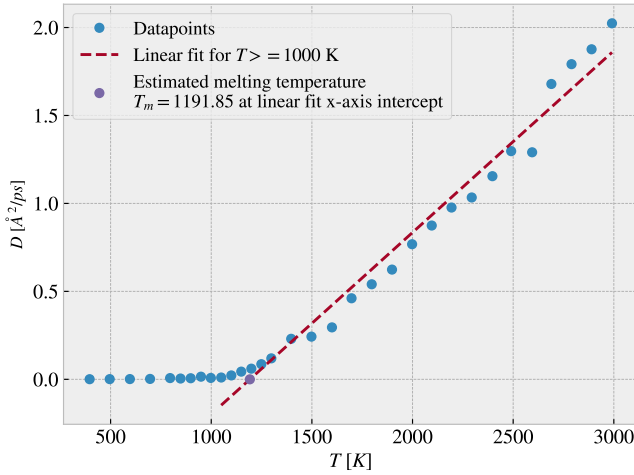


FIG. 14. The diffusion constant $D(T)$ as a function of temperature T . We see that the diffusion constant is approximately equal to zero until the estimated melting point $T_M = 1191.85$ K found as the intercept with the x-axis for a linear fit at $T > 1000$ K.

From the development of the diffusion constant as a function of T we see that there is a radical change around the estimated melting point $T_M = 1191.85$ K. This is the point where the atoms start to break out of the crystal structure and travel around. We didn't control the pressure which will effect the melting temperature, and this might explain the deviation from the known melting temperature at $T_{\text{melting}} = 1687$ K and the found melting temperature $T_m = 1191.85$ K.

m) SPCE water system

Using the water SPCE system we calculate $D(T)$ and $g(r)$ with similar methods as used in g) and l). See script *m.in* and *m.py* for more info. The results are shown in figure 15 and 16.

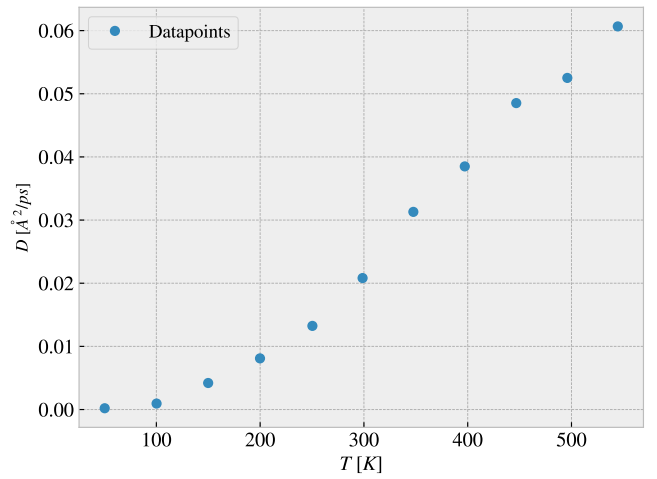


FIG. 15. Diffusion constant for water SPCE as a function of temperature

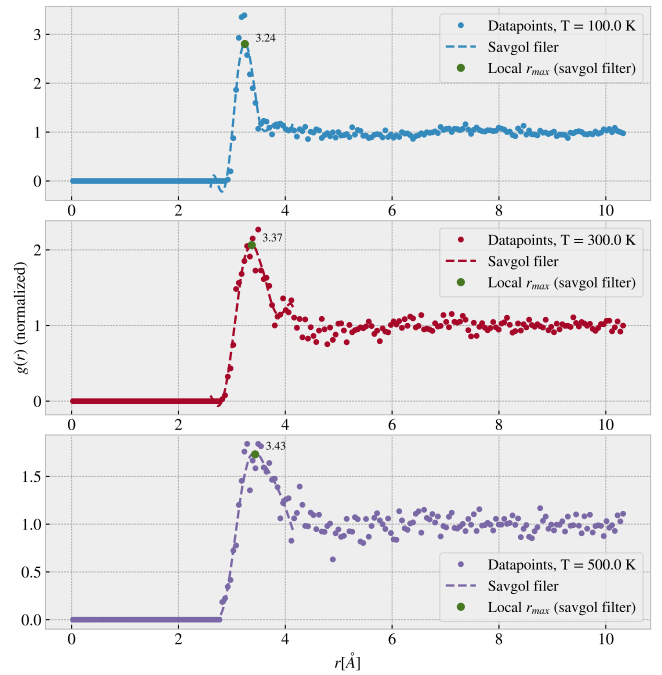


FIG. 16. Radial distribution for water with SPCE for three different temperatures corresponding to expected solid, liquid and gas phase.

From figure 15 we see a somewhat linear trend similar to the result for liquid argon (figure 10) and melted Si (figure 14). We would expect $D(T) = 0$ at the melting temperature of water at $273.14(100^\circ)$. This is not the case here, and this tell us that the low temperature behaviour isn't very accurate. When looking at the radial distribution in figure 16 we see almost no difference between the temperatures 100 K, 300 K and 500 K, which should represent all the three phases of solid, liquid and gas. From this we must conclude that the model (under

our simulation conditions) lack the ability to describe the phase transistions. However it seem to work great

as a description of the material transport (diffusion) as a function of temperature at higher temperatures.