

The basics of molecular dynamics in the microcanonical ensemble

<https://github.com/sver1/FYS3150-FYS4150>

Sverre Løyland

December 10, 2017

Code to perform molecular dynamics in the microcanonical ensemble has been implemented successfully and used to find the melting point of argon by calculating diffusion constants at different temperatures. The melting point was found to be a bit lower than 320 K.

1 Introduction

Molecular dynamics is computational method to derive thermodynamic properties of a system. By simulating a system over longer periods (on thermodynamic scale) and sampling quantities, certain properties can be calculated.

This project explores the basics of molecular dynamics, in particular the melting of solid argon. Here, 500 ^{40}Ar atoms interacting in Lennard-Jones potentials initially in a face-centred cubic lattice are simulated, and the simulation is repeated at different temperatures to derive a melting point.

2 Theory

2.1 Maxwell-Boltzmann distribution

The Maxwell-Boltzmann distributions are the probability distribution of the velocities, speeds and momenta of particles at equilibrium. The distribution assumes uncorrelated velocities and spherical symmetry of the distribution¹.

The form of the distribution used in the simulations is the distribution for the velocity given by

$$f_v(v_i) = \sqrt{\frac{m}{2\pi k_b T}} e^{-mv_i^2/2k_b T}. \quad (1)$$

¹Many sources says the particles need to non-interacting except in brief collisions, however this assumption is not necessary.

2.2 The equipartition theorem

The equipartition theorem relates temperature to energy averages. For gas, the theorem says the energy E is related to the temperature T by

$$E = T + V = \frac{3}{2}Nk_bT + 2\pi N\rho \int_0^\infty r^2 U(r)g(r) dr \quad (2)$$

where T and V are the kinetic and potential energy, U is the (spherical symmetric) interaction potential and g is the radial pair distribution function.

For an ideal gas where there are no interactions ($U = 0, V = 0$), this simplifies to

$$E = T = \frac{3}{2}Nk_bT. \quad (3)$$

In the simulations, the kinetic energy is assumed to be much bigger than the potential energy so an ideal gas is a good approximation.

2.3 Lennard-Jones potential

The Lennard-Jones potential is a spherically symmetric potential describing the Pauli exchange interaction and London dispersion force. The Pauli exchange interaction is a quantum mechanical effect related to the Pauli exclusion principle and the London dispersion is related to polarisability. The potential is on the form

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

where σ is the zero-point of the potential, i.e approximately how close atoms get) and ϵ is the depth of the potential at the minimum at $\sqrt[6]{2}\sigma$.

The parameters can be fitted with experimental values to give a fairly accurate potential which is mathematically and computationally very simple.

The corresponding force is given by

$$\mathbf{F} = -\nabla U(r) \quad (5)$$

$$= -\frac{dU(r)}{dr} \frac{\mathbf{r}}{r} \quad (6)$$

$$= -4\epsilon \left[12 \left(\frac{\sigma}{r} \right)^{11} \left(-\frac{\sigma}{r^2} \right) - 6 \left(\frac{\sigma}{r} \right)^5 \left(-\frac{\sigma}{r^2} \right) \right] \frac{\mathbf{r}}{r} \quad (7)$$

$$= 24\epsilon \left[\left(\frac{\sigma}{r} \right)^{12} - 2 \left(\frac{\sigma}{r} \right)^6 \right] \frac{\mathbf{r}}{r^2} \quad (8)$$

2.4 Einstein diffusion relation

The Einstein diffusion relation relates mean square deviation of particle positions to the time by

$$\langle r^2(t) \rangle = 6Dt \quad (9)$$

where D is a diffusion constant varying with temperature. Intuitively, we know the diffusion is a lot lower in solids than in liquids and the diffusion constant is used as an indicator of a phase transition.

3 Computational methods

3.1 Scaling

By convention, the mass and length units of molecular dynamics are chosen to 1 amu and 1 Å. The unit of energy is chosen to $\varepsilon = 1.65 \times 10^{-21}$ J and by using the scale factor $E = k_b T$, the temperature unit becomes 119.8 K.

3.2 Velocity-Verlet integration

A mathematically equivalent method to the Verlet method is the velocity-Verlet method. It incorporates the velocities making it self-starting and easily able to output the velocities for other calculations, e.g. kinetic energy.

The velocity-Verlet method is given by

$$x_{n+1} = x_n + v_n \Delta t + \frac{1}{2} a_n \Delta t^2, \quad (10)$$

$$v_{n+1} = v_n + \frac{1}{2} (a_{n+1} + a_n) \Delta t. \quad (11)$$

3.3 Periodic boundary conditions

To avoid having to deal with special boundary conditions, periodic boundary conditions are chosen where the potentials are calculated using the minimum image convention, i.e. interactions with the closest image of the particle. This also allows in some sense for an simulation box of infinite size. When particle distances become greater than than half the box size in either dimension, the particle image is shifted back into the cell.

However, care has to be taken when calculating properties based on distance such as diffusion. If only the images are saved, the particles can not diffuse more than a cell width in each direction so actual positions have to be calculated as well.

4 Implementation

The molecular dynamics were implemented in C++ based on a partially completed (almost spaghetti-like) code from andeplane on Github.com with some erroneous code. Figure 1 shows the code flow in the final implementation. The code does not output every integration step as writing to hard disk is slow. This also prevents some correlation in thermodynamic properties. When many simulations were done, the code to write the trajectory file was removed so save even more computation time.

In addition, a script for running the simulations and analysing the results were written in Python 3 to extract the melting point. By linear regression of the mean square

deviation over time, a diffusion constant is extracted together with the mean temperature. The diffusion constants are then plotted against the mean temperatures.

5 Analysis

In each simulation 500 ^{40}Ar atoms in a face-centred cubic lattice with lattice constance 5.26 Å ($\rho = 500 \cdot 18 \text{ amu} / (5 \cdot 5.26 \text{ Å}^3) = 822 \text{ kg/m}^3$) were simulated for 5001 integration steps with a time-step of 1 fs at a range of different temperatures. First a wide range of temperatures were simulated to find the approximate melting point and then narrower ranges of temperatures were simulated until a satisfactory value could be read off. Figure 2 shows the diffusion constants over temperatures. There is a very apparent kink at about 320(10) K.

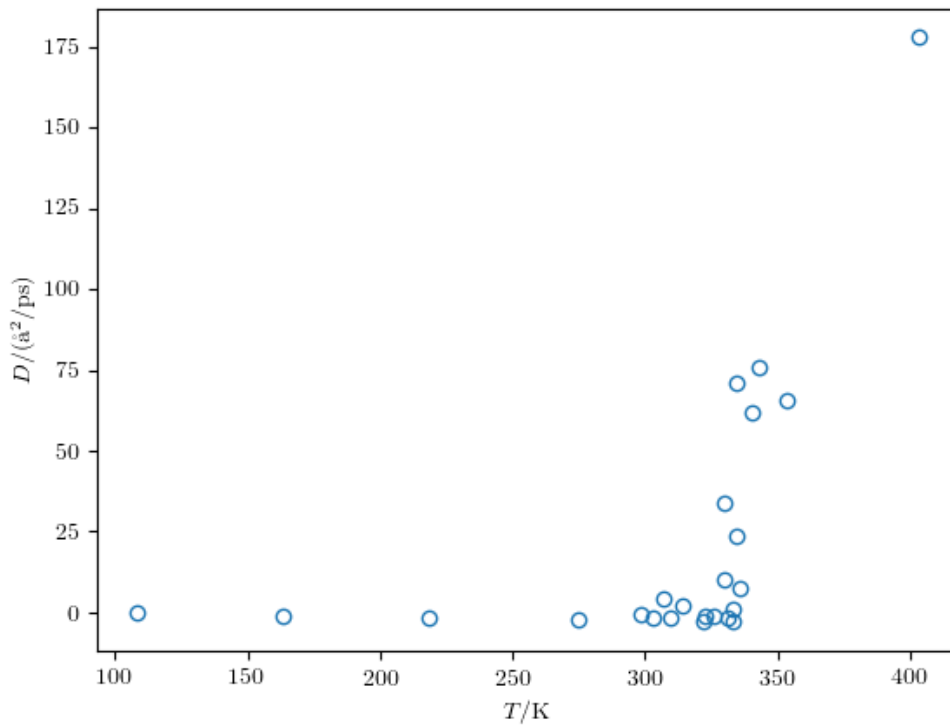


Figure 2: Plot of diffusion constants over temperatures.

The diffusion constants are calculated from linear regression of the whole simulation, not just at equilibrium so if the solid melts at the end of the simulation, only a small diffusion constant is observed or not observed at all if the simulation never reaches equilibrium in the simulation time. This mean the melting point is lower then the kink.

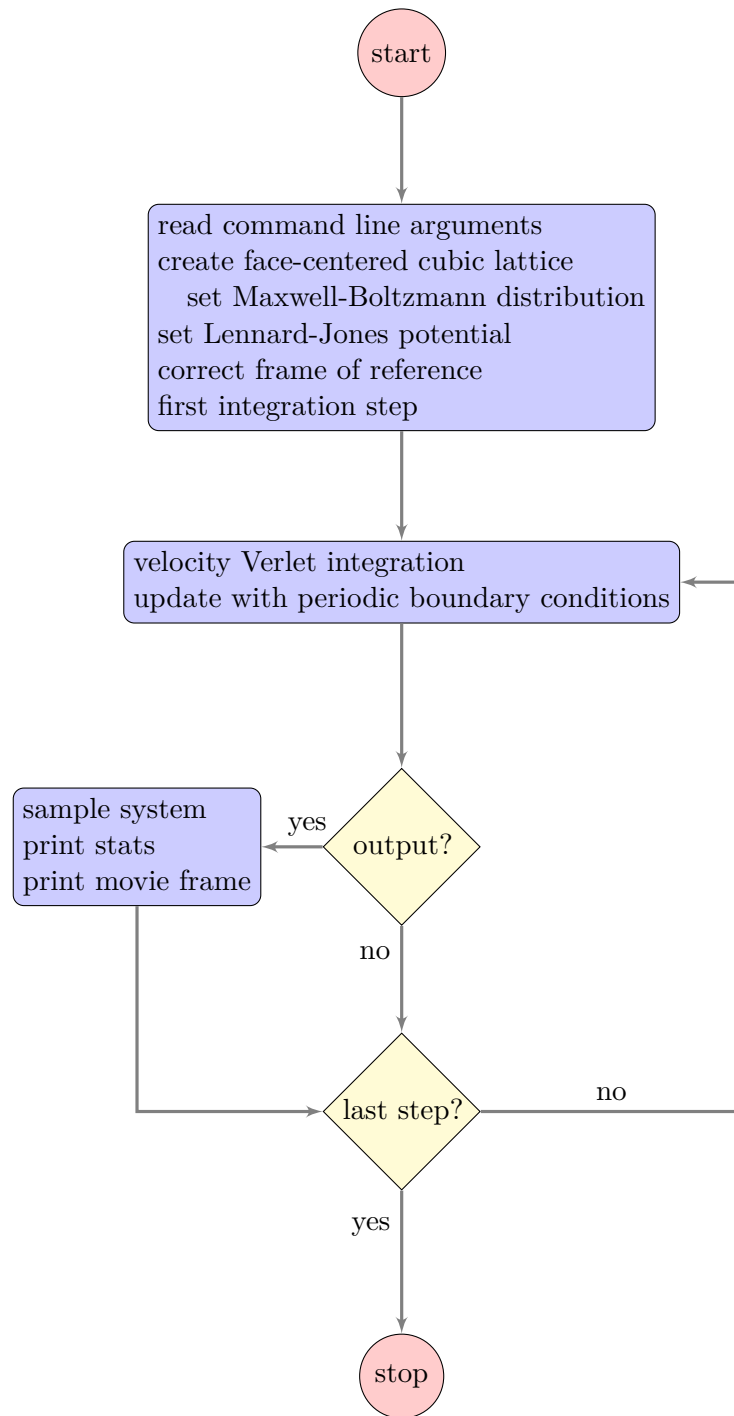


Figure 1: Flowchart for the final implementation of the molecular dynamics simulation.

To induce melting and reach equilibrium faster, a set of random atoms can be removed. This is called the voids method. When more and more the atoms are removed, the observed melting points converges to a better estimate.

To get a more accurate melting point range, a simulation in the canonical ensemble where the temperature is varied throughout the simulation can be used, but this requires implementation of a thermo-“stat” that can adjust the temperature. The melting point can then be estimated from e.g. the density. However, since the temperature is changing, the system never reaches equilibrium and there will be hysteresis. By annealing, a corresponding result can be extracted for the freezing process, but this simulation “lags” in the opposite direction so the melting point have to be in-between the observed melting points.

The melting point is difficult to compare to values in the literature as these are usually in the isothermal–isobaric ensemble. To do calculations in the ensemble, both a thermostat and a barostat would need to be implemented.

The relation between the initial temperature and the final temperature is shown in fig. 3. As can be seen, the temperature falls to about half of the initial temperature with slight variations depending on the whether the solid melts or not.

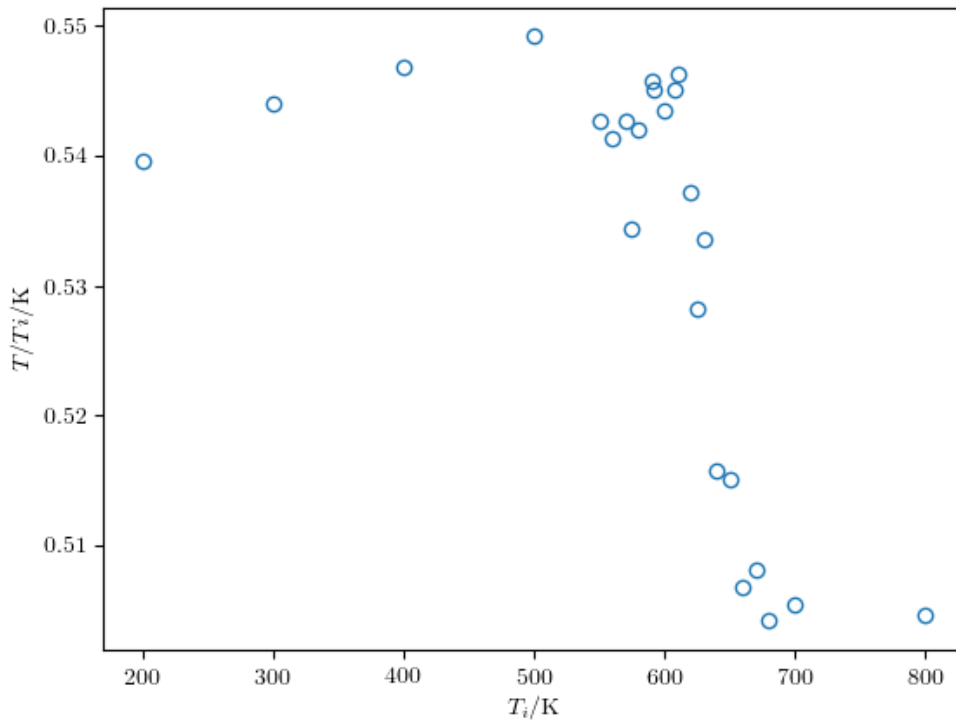


Figure 3: Plot of ratio of temperatures and initial temperature over initial temperatures.

6 Conclusion

The desired algorithms were successfully implemented and a melting point extracted. The melting point is probably a bit below 320 K for the conditions describes earlier. However, this value does not really represent a usefully quantity as it is in the micro-canonical ensemble, but the code serves as a base to build better simulations.