

Heat capacity of gases using molecular dynamics

Dag Kristian Dysthe

Department of Physics, University of Oslo, PObox 1048 Blindern, 0316 Oslo

Compute C_V and C_P for an atomic LJ fluid and solid and N_2 gas. Compare the obtained values to ideal gas and experiments. Relate the gas results to your experimental results for N_2 and determine how many degrees of freedom the N_2 models have.

I. PREREQUISITES

You need to bring your own laptop with the following programs installed and working:

- LAMMPS
- Molecular visualization software like Ovito
- Matlab or Python

II. EXPERIMENT + MD LAB COMBINED

This lab is intended for you to get to know some more about molecular dynamics simulation, measurement of sound velocity of gases and to learn about specific properties of model gases versus real ones. The heat capacity and the speed of sound are macroscopic, thermodynamic properties of matter. In the experiments you perform macroscopic measurements, the instruments know nothing about the nature of the molecules in the gases. In the MD simulations you only manipulate and measure microscopic quantities like velocities of the individual atoms and you must yourself take statistical averages to obtain the macroscopic properties. Most experiments and simulations will be performed on ideal gases where you can find simple relations between what you measure and the degrees of freedom of the gas.

Document your work and what you learn and deliver one single report (including both experiments and simulations) at the end of the day. The report may be written in Python notebook, Matlab livescript or in a word processing system like Word or LaTeX. The report you deliver should be a single PDF file.

This document formulates a series of tasks to make sure you get help doing simulations to learn about models, MD simulations, degrees of freedom, etc. You may answer the specific tasks in the report appendix, but more importantly we want you to write the main body of the report about what the combination of experiments and MD simulations taught you. More guides for writing reports can be found on the course webpage.

III. TASKS IN MD PART OF LAB

Read the LAMMPS input scripts `heatcapLJ.in` and `heatcapN2.in` provided for running LAMMPS simulations. The scripts specify an experiment where the model gas is thermalized for some timesteps and then in the last run some energy is added to the system at each timestep. We are interested in how much the system temperature rises when heating. From this you can calculate the heat capacity and compare to the experiments and ideal gas theory.

Try to understand what the input file specifies about the simulations: Number of particles N , density ρ , volume V , temperature, T , initial configuration, what is kept constant (NVE, NVT, NPT), heat flux, atom/molecule model, dump output and thermodynamic output.

Do some test runs and open `dump.lammpstrj` with a visualizer like Ovito to get an idea of what is going on. Try to vary the number of particles, density, starting temperature and energy flux and note how these changes affect the simulations. Find a density-temperature phase diagram for your system and note where in the phase diagram all of your simulations are performed.

Write a simple analysis script in Matlab or Python that uses the thermodynamic data in `log.lammps` and trajectory information in `dump.lammpstrj`.

Task 1

The heat capacities at constant volume, C_V and constant pressure, C_P , are defined in the Schroeder textbook on pp. 28-29. Use the equipartition theorem

$$U = N f k_B T$$

and ideal gas equation of state

$$PV = N k_B T$$

where N is the number of molecules and f the degrees of freedom of the molecules, to find an expression for the difference $C_P - C_V$ for an ideal gas.

Task 2

Run simulations on the Lennard-Jones system. From the data in the logfile and trajectory files determine C_V with uncertainties. Compare the value to the ideal gas value and comment. Determine the number of degrees of freedom of the gas particles.

Task 4

Can you determine the temperature T and pressure P of a dilute gas from the velocity data of the trajectory? Do they agree with the T and P from the thermodynamic logfile?

Task 5

For a gas density try temperatures from the triple point temperature to 10 times the critical temperature. Does C_V change with temperature?

Task 6

For a temperature above the critical temperature try densities from a solid to a dilute gas. Does C_V change with density?

Task 7

How can you do an MD experiment to measure C_P instead of C_V ? For a dilute gas density and the triple point density calculate C_P . Compare C_P and C_V . Does the difference correspond to the theoretical relation you derived?

Task 8

Now perform a simulation to determine C_V for the two N_2 models. Comparing C_V to ideal gas C_V : how many degrees of freedom do these N_2 models have? How does this compare to the number of degrees of freedom that you derived from sound velocity measurements in the lab?

Task 9

Turn off the heat flux in the simulation of one of the models (choose any of the models). Use the fluctuations of kinetic energy in a NVE run to calculate C_V :

$$\langle \kappa^2 \rangle_{NVE} - \langle \kappa \rangle_{NVE}^2 = \frac{3k_B^2 T^2}{2N} \left(1 - \frac{3k_B}{2C_V} \right), \quad (1)$$

where κ is the kinetic energy. Alternatively, you may use an NVT run and this expression:

$$C_V = \frac{1}{k_B T^2} \langle E^2 \rangle_{NVT} - \langle E \rangle_{NVT}^2 \quad (2)$$

Do these agree with the non-equilibrium experiment?

Appendix A: Tools

1. LAMMPS

LAMMPS has been introduced in the lectures. For this project you will use two input files, `heatcapLJ.in` and `heatcapN2.in`, that only need small modifications. In addition the N_2 simulations need the file `N2.molecule` in the directory where the simulation is run.

2. Visualization

Using Ovito or some other program to visualize the motion of the particles during the simulation is very useful to understand how they move, if the particles behave as you expected, to see which phase they are in, etc. It is useful to use images from the visualization in the report.

3. Analysing data in logfiles and trajectory files

Thermodynamic data is written to the logfile `log.lammps` at regular intervals specified by `thermo X`, meaning “write thermodynamic data to logfile every X timestep”. Trajectory data of all atoms at every 100 timesteps is specified by the command

```
dump 1 all custom 100 dump.lammpstrj id x y z vx vy vz
```

in the input files. During this project you have to write your own analysis programs in Python or Matlab that take data from the log- and trajectory files and calculate heat capacities and degrees of freedom in different ways.

4. Python

A tool to read a logfile produced by LAMMPS into a simple python data structure with a `get()`-function providing the log data can be found at: <https://pypi.org/project/lammps-logfile/>

5. Matlab

Scripts to read logfile and dumpfile into Matlab can be found in

<https://www.uio.no/studier/emner/matnat/fys/FYS2160/h20/Molecular-Dynamics/>

The script `readLammps.m` is specific for the output defined in `heatcapLJ.in` and `heatcapN2.in`.

Appendix B: Particle models in MD

You will use molecular dynamics to study three different model substances. The models are defined by their interatomic interaction potentials and their mass m .

1. Atomic model

The atomic pair interactions in both the atomic and N_2 simulations are defined by the Lennard-Jones/cut potential:

$$U(r < r_c) = 4\epsilon \left(\left(\frac{\sigma}{r} \right)^{12} - \left(\frac{\sigma}{r} \right)^6 \right)$$

$$U(r \geq r_c) = 0,$$

that has a characteristic energy, ϵ , diameter σ and cutoff distance r_c . The cutoff distance is set to $r_c/\sigma = 2.5$ by

```
pair_style lj/cut 2.5
```

2. N_2 models

The N_2 molecule consists of two identical atoms connected by a bond. A bond can be modelled as quantum mechanical harmonic oscillator, but since the MD simulations are classical we will use either a spring or a rigid bond connecting the two atoms. The bond defined by

```
bond_style class2
bond_coeff 1 r0 K2 K3 K4
```

is

$$E = K_2(r - r_0)^2 + K_3(r - r_0)^3 + K_4(r - r_0)^4.$$

The other option is to fix the distance between the two nitrogen atoms with the command

```
fix shake all shake 0.0001 20 0 b 1
```

and the length of the bond is set to 1.0977 Å in the file `N2.molecule`.

The interactions between nitrogen atoms that are not bonded are defined by

```
pair_style lj/class2 10.0
pair_coeff * * 0.05980 3.8008
```

which means that $r_c = 10$ Å, $\epsilon = 0.0598$ kcal/mol and $\sigma = 3.8008$ Å.

Appendix C: “real” units

You may find the units of C_V from the definition of style *real* units (see <http://lammps.sandia.gov/doc/units.html>)

- mass = grams/mole
- distance = Angstroms
- time = femtoseconds
- energy = Kcal/mole
- velocity = Angstroms/femtosecond
- force = Kcal/mole-Angstrom
- torque = Kcal/mole
- temperature = Kelvin
- pressure = atmospheres
- dynamic viscosity = Poise

- charge = multiple of electron charge (1.0 is a proton)
- dipole = charge*Angstroms
- electric field = volts/Angstrom
- density = gram/cm³
- calorie = 4.184 J