

FYS2160 Lab1: Gas thermodynamics

D. K. Dysthe, Vetle A. Vikenes, C.A. Lütken and A. L. Read

September 21, 2023

Abstract

The ideal gas is a theoretical idealisation that is valid for gases at low pressure and density. In this exercise you will do experiments to probe the range of validity of the **ideal gas model** and the **degrees of freedom** of the gases. You have two types of instruments at your disposal to do experiments,

- **Kundt's tube** that measures the **speed of sound** in three physical gases:
 - air
 - CO₂
 - argon
- **Lammps MD simulator** that can be used to measure the **compressibility** and the **heat capacity** in three model gases:
 - atomic LJ
 - diatomic LJ with spring bond
 - diatomic LJ with stiff bond

The objective of this lab is to find answers to the following questions:

- When can we consider a real gas to be ideal?
- In what range of pressures, densities and temperatures are the deviations from ideality negligible? (What is "negligible"?)
- In which sense can molecular gases be considered ideal gases?
- Can we model the deviation from ideal behaviour?
- Do the MD model predictions agree with experiments?

You can explore this "unstructured problem formulation" in the way you want to. If you find this too difficult, we have formulated some more structured "tasks". In both cases: in order to answer the questions well you will have to analyse your results as you go and refine your physical and numerical experiments.

Practicalities

- You need to read this document and prepare yourself before the lab
- You need to bring your own laptop with the following programs installed and working:
 - Lammmps
 - Ovito
 - Matlab or Python
 - scripts/pieces of code to do linear regression and analysis of accuracy on data
- The lab lasts a whole day from 08:15-16:00. During this time you will
 - do physical experiments
 - do Lammmps simulations
 - analyse, interpret and discuss data
 - discuss physics
 - write a report
- The report should be handed in by Sun 23:59

About the report

Document your work and what you learn in one single report including both experiments and simulations. We suggest that while you work you will document in your report: your planning, why you do the experiments you do, your reasoning when obtaining the results, change of plans, new results and final conclusions. This will make the document something between a lab notebook and a report.

We suggest writing the report in L^AT_EX using Overleaf, but you may also use Jupyter notebook, Matlab livescript or a word processing system like Word. The report you deliver on Canvas should be a single PDF file. Students working in a group may deliver identical reports. Students working together should identify their partners on the front page of their report.

Before you leave the lab you have to show your report to a supervisor. The supervisor will decide if your report is close enough to be finished. The report must be handed in no later than 1 day after the lab is finished.

1 The ideal gas and degrees of freedom

The simplest model of an atom is an indivisible mathematical point (*rigid ball*) without “personality” (i.e., no other physically measurable attributes). The simplest model of a molecule is rigid ball atoms which are connected by “sticks” with no structure. This *ball&stick* molecule can move and rotate in space, but that is all. A better model is to replace the sticks with springs, so that the molecular bond also can vibrate (*ball&spring* model). These two molecular models come in variants: quantum mechanical (QM) and classical. The QM models have discrete energy levels of rotation and vibration. These discrete levels, ε_i , are occupied according to the Boltzmann probability $P_i = \frac{1}{Z} e^{-\varepsilon_i/kT}$, where Z is the partition function. This means that at temperatures $T \ll \varepsilon_i/k$ there will be no rotation/vibration. In the classical model there will always be some rotation and vibration.

1.1 Heat capacity of ideal gases

For ideal gases we have a very direct bridge between micro- and macro-physics, via the *heat capacity*:

$$C_V = \left(\frac{\partial U}{\partial T} \right)_V \quad (1)$$

$$C_p = \left(\frac{\partial U}{\partial T} \right)_P + P \left(\frac{\partial V}{\partial T} \right)_P \quad (2)$$

In the *ideal gas* model air molecules (ball, ball&stick, ball&spring) do not interact and the only contributions to the heat capacity is from each molecule separately. Therefore, the only information that thermodynamics retains about this “mathematical” gas is how many *thermodynamic degrees of freedom* f the molecules have, and how heavy they are. The more degrees of freedom the molecules have, the more heat they can store.

The heat the molecules can store is measured by the molar heat capacities $c_p = C_p/n_{\text{mol}}$ (constant pressure; *isobaric* process) and $c_V = C_V/n_{\text{mol}}$ (constant volume; *isochoric* process), where n_{mol} is the number of moles used to measure the extensive heat capacity C_p and C_V . n_{mol} is the amount of matter measured in the SI unit *mol*. The number of moles of gas molecules is $n_{\text{mol}} = m/M_{\text{mol}} = N/N_A$, where m is the mass of the gas, M_{mol} is the mass of one mole of the gas (the *molar mass*), N is the number of molecules in the gas, and N_A is Avogadro’s number.

In an ideal gas at normal temperature every degree of freedom contributes $R/2$ to c_V :

$$c_V = f \frac{R}{2}, \quad c_p = (f + 2) \frac{R}{2},$$

where R is the *molar gas constant* (a.k.a. the *universal* or *ideal* gas constant)

$$c_p - c_V = R = 8.3144598(48) \text{ J/(K} \cdot \text{mol)}.$$

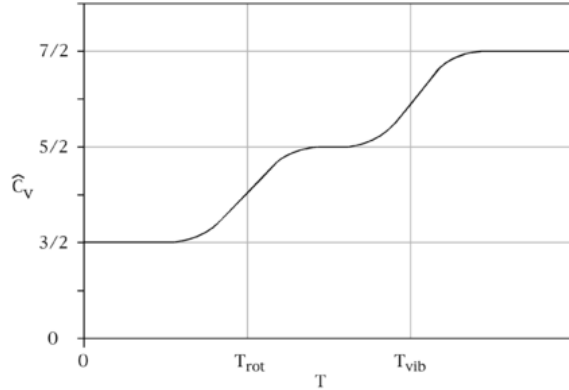


Figure 1: Idealised behaviour of heat capacity of simple molecular gases as function of temperature, $\hat{c}_V = \frac{C_V(T)}{n_{mol}R} = \frac{c_V(T)}{R}$. The temperatures T_{rot} and T_{vib} is where the quantized rotational and vibrational states are excited. Which type of molecules can be assumed to follow such an idealised behaviour?

The *adiabatic index* (a.k.a. the *heat capacity ratio*, the *ratio of specific heats*, *Laplace's coefficient*, or the *isentropic expansion factor*) for an ideal gas is

$$\boxed{\gamma = \frac{c_p}{c_V} = \frac{f+2}{f}}. \quad (3)$$

This ratio determines the macroscopic *adiabatic equations* for an ideal gas, which assert that pV^γ , $TV^{\gamma-1}$, and $Tp^{1/\gamma-1}$ are constants.

1.2 Speed of sound of ideal gases

From wave mechanics we know that the speed of sound c depends on the density ρ of the material (gas) and the *adiabatic compression modulus* K : [5]

$$\begin{aligned} K &= -V \frac{dp}{dV} = \gamma p \implies \\ c &= \sqrt{\frac{K}{\rho}} = \sqrt{\frac{\gamma p}{\rho}} = \sqrt{\frac{(f+2)p}{f\rho}}, \end{aligned}$$

where we have used that $K = \gamma p$ follows from the adiabatic equation $p \propto V^{-\gamma}$ and Eq. (3).

Example: The density of air is $\rho_0 \approx 1.29 \text{ kg/m}^3$ at $T_0 = 0^\circ\text{C}$ at sea level, where the air pressure is $p_0 = 1$ atmosphere $\approx 1.0125 \times 10^5 \text{ Pa}$. If air is an ideal gas, then $f = f_{rom} = 3 + 3 - 1 = 5$ gives the speed of sound $c_0 \approx 331.5 \text{ m/s}$, which is in good agreement with the experimental value. [2]

From $\rho = m/V = nM_{\text{mol}}/V$ and $pV = nRT$ we get a thermodynamic equation for c in an ideal gas:

$$c_{\text{id}}(T) = \sqrt{\frac{(f+2)RT}{fM_{\text{mol}}}}, \quad (4)$$

where T is the *absolute temperature* (measured in K).

Notice that the molar mass M_{mol} determines the speed of sound: the lighter the gas, the faster sound waves move through it. Compare for example the speed of sound in helium, which at room temperature (20°C) is more than 1000 m/s (cf. inset in Fig. 13), with the speed of sound in air. An exception from this rule is neon, which is a bit heavier than ammonia and water: $M_{\text{mol}}(\text{NH}_3) = 17.03 \text{ g/mol} < M_{\text{mol}}(\text{H}_2\text{O}) = 18.02 \text{ g/mol} < M_{\text{mol}}(\text{Ne}) = 20.12 \text{ g/mol}$. The reason is that neon has fewer degrees of freedom, which in this case is enough to overcome the small difference in molecular masses.

2 Real gases

For an ideal gas the *equation of state* is

$$pV = NkT = n_{\text{mol}}RT.$$

The fact that most substances can change phase from gas to liquid and solid shows that the ideal gas model is very limited in its applicability. We are interested in how applicable the ideal gas model is for substances we know as gases in our daily lives: air (mainly N_2 and O_2), CO_2 and argon.

2.1 Virial expansion

Rearranging the ideal gas equation of state in terms of the compressibility factor, $Z = \frac{pV}{NkT} = 1$, we can express deviations from the ideal gas model in terms of the virial expansion.

$$Z = \frac{pV}{NkT} = 1 + B\rho + C\rho^2 + \dots, \quad (5)$$

where $\rho = N/V$ and B and C are the virial coefficients. This virial expansion is a macroscopic model equation of state that takes into account that the molecules have finite size and interact. A positive second virial coefficient signifies that the finite size of the molecules and repulsive interactions between them dominate (see Figure 2), a negative B signifies that the attractive interactions between the molecules dominate.

2.2 Heat capacity

In real gases the internal energy has contribution from both kinetic energy of the molecules, U_k (as for ideal gases) and potential energy from interaction between

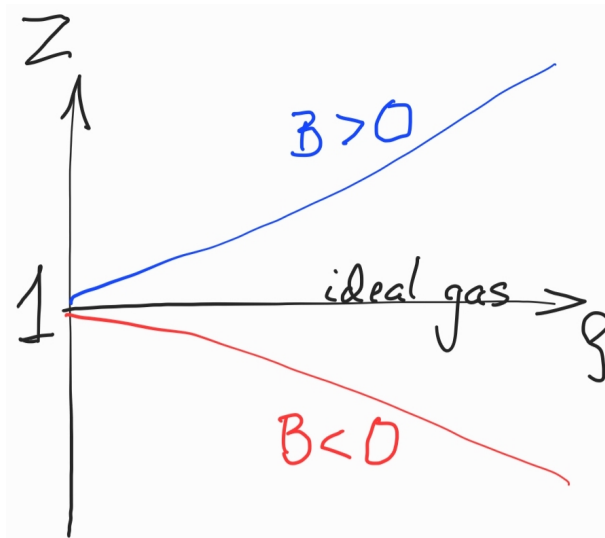


Figure 2: Non-ideal gases have compressibility factors that deviate from 1. The second virial coefficient B is a measure of positive or negative deviation from ideality.

molecules U_p , $U = U_k + U_p$. The Lennard-Jones interaction energy, $U_{ij}(r_{ij})$, in equation (8) is a model of such interaction energies with a positive energy for distances $r_{ij} < \sigma$ and negative energies for distances $r_{ij} > \sigma$. From the definition of the heat capacities (1) and (2) it is clear that the addition of U_p will change the heat capacities. An estimate of the correction to the heat capacity is:

$$\hat{c}_V = \frac{C_V}{\rho n_{\text{mol}}} \approx \hat{c}_{V,\text{idealgas}} - 2\rho T \frac{\partial B}{\partial T} \quad (6)$$

3 Experiments using sound waves

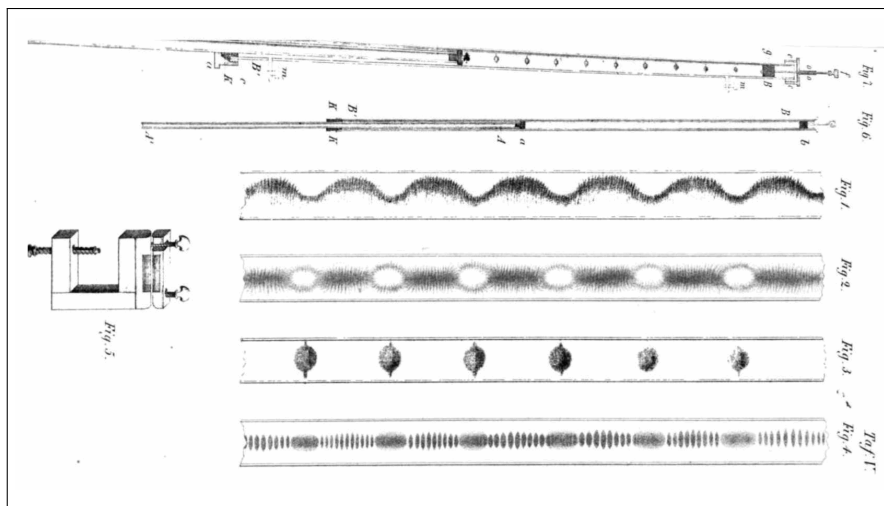


Figure 3: Original illustration from the article by August Kundt in *Annalen der Physik* in 1866, which shows standing waves inside *Kundt's tube*. [1] We shall here repeat his experiment with modern equipment, with one of the objectives being to test thermodynamic gas theory.

Sound is a longitudinal pressure wave, and the speed of this wave depends on temperature, pressure, and other thermodynamic quantities. We are here going to use Kundt's tube to measure the speed of sound (cf. Fig. 3), use this to investigate how the heat capacity $c_*(s, M_{\text{mol}}, T)$ depends on molecular structure and temperature, and compare and contrast this with the theory of ideal gases.

3.1 Experimental determination of the speed of sound by finding resonances in the tube

We can determine the speed of sound in a gas by identifying the resonance frequencies of standing waves inside a tube, since we know that the velocity c of a wave always is given by $c = \lambda \nu$, where λ is the wavelength. From wave mechanics we know that a standing wave in a closed tube with resonance frequency ν_n has wavelength $\lambda_n = 2L/n$, where L is the length of the tube and n is an integer ($n = 1, 2, 3, \dots$). Combining these results we obtain the linear function

$$\nu_n = an + b, \quad a = \frac{c}{2L}, \quad (7)$$

where the speed of sound is determined by the slope a . To get a better linear fit for the slope we leave the value of b undetermined. [3]

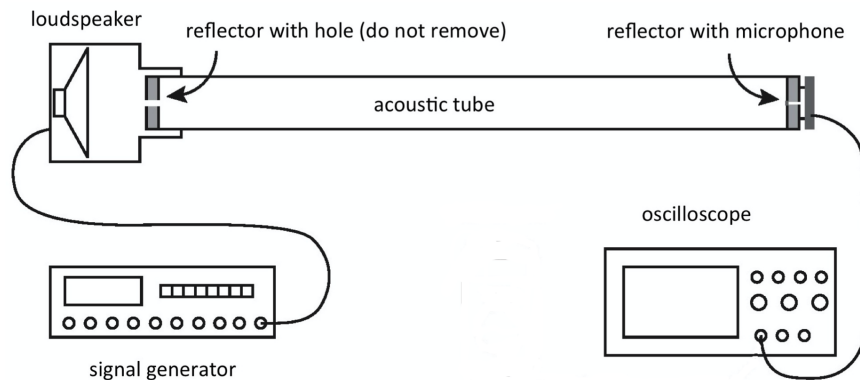


Figure 4: Sketch of the device used to measure the speed of sound in a gas (air, argon, or CO_2 , in this lab). The temperature inside the tube is monitored by a tiny thermistor (not shown here) that does not obstruct the sound waves. Two of the tubes are wrapped with a heating cable and one of these with insulation (not shown here) so that the gas can be heated (to at most 70°C). OBS: The recommend heating voltage (or current) is shown on the stand holding the tube.

The uncertainty δc of your best estimate $\bar{c} = 2\bar{L}\bar{a}$ of the speed of sound is obtained by using the “Pythagorean method” (described in the appendix) for calculating how the uncertainties of a and L propagate through the function $c(a, L) = 2La$. The value of L and δL is given on each tube (they are not all the same).

3.2 How to carry out the experiments

How the temperature is measured and how to calculate the temperature is described in section C

We are going to use standing waves to measure the speed of sound, with an apparatus sketched in the diagram shown in Fig. 4.

The gas is contained inside a long tube (with a specified internal length L with uncertainty $\delta L = \pm 1.5 \text{ mm}$, measured with a laser), which is plugged at both ends with massive metal disks. One of the plugs has a small hole in the center that emits sound waves from a loudspeaker attached to the outside of the plug. The speaker is driven by an alternating harmonic current delivered by a signal generator, which has a number of knobs on the right hand side where the amplitude (signal strength) can be adjusted so that the sound detector does not “clip” the signal. An input signal of $\approx 35 \text{ mV}$ (RMS, measured by the oscilloscope) has been found to give robust results.

The plug at the other end of the tube is equipped with a miniature microphone, which is connected to a battery driven amplifier attached to the outside

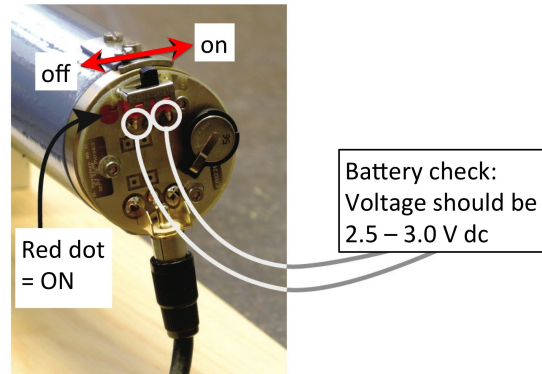


Figure 5: Photograph of the circuit board with on/off switch for the microphone amplifier, which runs on a small battery (round disk on the right hand side) that should have a nominal voltage of at least 2.3 V.

of the plug (cf. Fig. 5). The signal from this amplifier, which is proportional to the pressure in the gas at the microphone, is sent to an oscilloscope. The oscilloscope, a PicoScope, is controlled from a lab-PC via USB. Double-click on the desktop icon `PicoScope-Kundts-K%.pssettings` (with $\% = 1, 2, 3,$ or 4) to start the PicoScope with settings that are tuned to a 35 mV input signal. Make sure that both the input and output signals are unclipped harmonics (sines) before you proceed. If there is no harmonic output signal (the "B" or red trace on the PicoScope), then use a multimeter to check the health of the battery for the amplifier (cf. Fig. 5).

Our task is to identify resonance frequencies where the signal is much stronger than neighboring frequencies. The advanced signal generator can deliver frequencies with a precision of 10^{-3} Hz, but we cannot determine the maximum peaks on the oscilloscope with anything like this precision. Estimate the uncertainty in your readings.

There will be four groups analysing the speed of sound in four different tubes:
 [ATTENTION: You are not allowed to fill any gas other than air by yourself!]

K1: contains argon or CO_2 at $T = T_{\text{room}}$

K2: contains air at $T = T_{\text{room}}$

K3: contains air at $T \simeq 70^\circ\text{C}$

K4: contains air at $T \simeq 50^\circ\text{C}$

All results from K1 - K4 will be shared, so that you can compare and contrast them. After completing the next exercise you will compare these experimental results with theoretical expectations for ideal gases.

Structured formulation of tasks

1. Find all resonance frequencies in K1 - K4 in the frequency interval from about 200 Hz up to about 2 kHz. You may find a resonance below 200 Hz, but that one is so uncertain that it is better to use higher frequencies. Each reading should be as accurate as you can manage with about 30 seconds of “fine-tuning” for each resonance.
2. Estimate (roughly) the uncertainty in each frequency measurement. Since it is difficult to find the longest wave (how long?) it is better to plot differences. This eliminates any systematic mislabeling of the data, i.e., use $\Delta\nu_k$ from the previous exercise.
3. Find the best linear fit to the data, and use Eq. (7) to find the speed of sound.
4. What is the most important contribution to the uncertainty? You may find the PYTHON code in the appendix useful. It will return the least squares fit to the data, including the uncertainty in the slope. Verify that including more points shrinks this uncertainty.
5. Determine the thermodynamic degrees of freedom of the gases from the experiments and compare to your theoretical prediction.
6. Compare experiments to simulation and theory. Remember to use the uncertainties of your measurements when comparing!
7. Extra: Use the principle of corresponding states to determine which T^* and ρ^* your experimental conditions of CO_2 correspond to. Using Lammmps, determine the compressibility factor Z at this state. Use the compressibility factor to correct the theoretical estimate of the speed of sound c_s . Does this improve the agreement between experiment and theory?

4 Molecular dynamics simulations

In order to better understand the relation between microscopic, molecular parameters and macroscopic properties like compressibility and heat capacity we can use Molecular Dynamics (MD) to study molecular models. The simplest model is to represent the molecule as a spherical particle with a finite diameter σ and attractive energy ϵ .

4.1 Lennard-Jones

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

$$\begin{aligned} U(r < r_c) &= 4\epsilon \left(\left(\frac{\sigma}{r} \right)^{12} - \left(\frac{\sigma}{r} \right)^6 \right) \\ U(r \geq r_c) &= 0, \end{aligned} \quad (8)$$

that has a characteristic energy, ϵ , diameter σ and cutoff distance r_c . The cutoff distance is set to $r_c/\sigma = 2.5$ in this pair interaction definition for Lammmps:

```
pair_style lj/cut 2.5
```

4.2 Molecular models with atoms and bonds

Since the LJ model does not have any internal degrees of freedom we have to use a molecular model to get a better representation of the heat capacity. The diatomic Lennard-Jones 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. In Lammmps 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 atoms with the command

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

and the length of the bond is set to 0.7σ in the file `diat.molecule`.

4.3 Simple MD experiments

There is a range of MD experiments that you can do, but we will only mention two simple experiments. Both experiments require that you use thermodynamic

data from the file `log.lammps`, enter them into Python or Matlab and do linear regression of some theoretical function to the data. You may yourself choose at what temperature you will start the experiments:

```
variable T equal 1
```

Here $k_B T/\epsilon = 1$. And if your computer is fast you may want to increase the number of atoms/molecules

```
variable N equal 100
```

You may have to increase the run length (for example `run 100000`) in order to get stable measurements.

4.3.1 Compressibility

If you do a number of simulations at the same temperature, but vary the density, you can record the pressure as function of density to calculate the compressibility. You may vary the density by changing the variable `rho` from 0.002 to something else in

```
variable rho equal 0.002
```

4.4 Heat capacity

Both the atomic (`heatcapLJ.in`) and molecular (`heatcapLJdiat.in`) input files are prepared to do an experiment where a certain amount of energy, `eFlux`, is added to the system in each time step.

```
fix heat all heat 1 ${eFlux} region simbox
```

From the temperature and energy measurement you may then calculate the heat capacity of the system. Remember to set `eFlux` to zero when you do not want to add energy during the simulation.

5 Structured formulation of tasks

Read the Lammps input scripts `heatcaplj.in` and `heatcapljdiat.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,

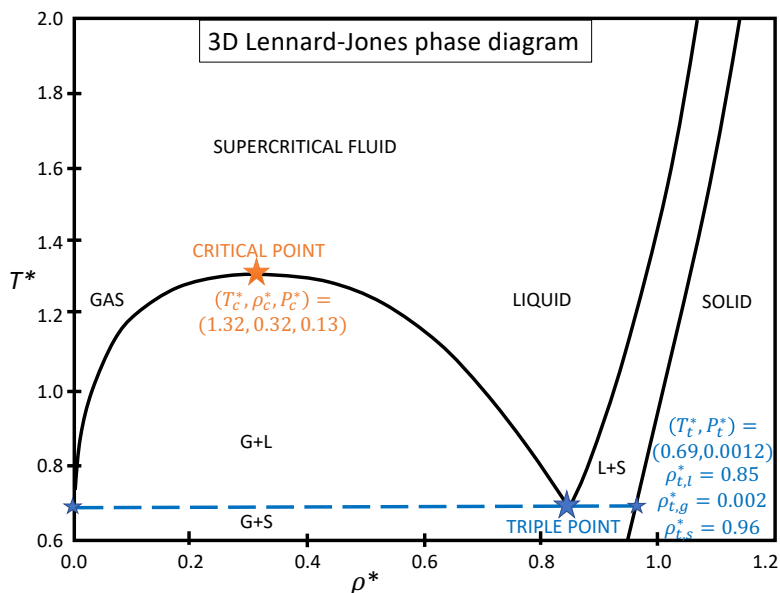


Figure 6: Phase diagram of the LJ model.

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

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

Solution:

I would like my system to be in the gas phase for all temperatures, thus I choose half the triple point gas density, $\rho^ = 0.001$ and I start the simulation slightly below the triple point temperature, $T^* = 0.6$. First time I run the simulation I*

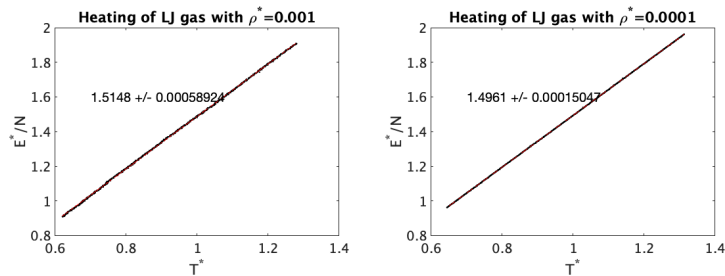


Figure 7: First simulations yielded a slope of $3/2$, thus number of degrees of freedom $f = 3/2$.

just use the "eFlux=1" that was in the input file. The first run I try 100000 timesteps. Observing the output to screen I see that the temperature only changed by 0.68. First simulation yielded a slope of almost $3/2$, thus number of degrees of freedom $f = 3/2$. The very small standard deviation only says that the line is very straight. The value of the slope is 1 % too high compared to an ideal gas. Is this because the gas is too dense? I ran the simulation again with $\rho^ = 0.0001$ and then the deviation was only 0.3%*

I used a simple Matlab script to do this:

```

fs=20;
logdata = readlog('log.lammps');
lastarray=str2num(logdata.data{4});
%Step Temp E_pair E_mol TotEng Press
T=lastarray(:,2);
E=lastarray(:,5);
figure(1)
X=[ones(size(T)) T];
[ b,stdb ] = linregr( X,E );
Ehat=X*b;
plot(T,E,'.k',T,Ehat,'--r')
text(0.7,1.6,[num2str(b(2)),' +/- ',num2str(stdb(2))],'FontSize',fs)

%%%%% pretty plotting
ax2=gca;
ax2.FontSize=fs;
ax2.FontName='TimesRoman';
xlabel('T^*')
ylabel('E^*/N')
title('Heating of LJ gas with \rho^*=0.0001')

```

Now I got curious about what it will be in the liquid phase. So I started a simulation at the triple point density $\rho^ = 0.85$ and a temperature slightly above the triple point temperature, $T^* = 0.8$. I ran the simulation for 500000 timesteps*

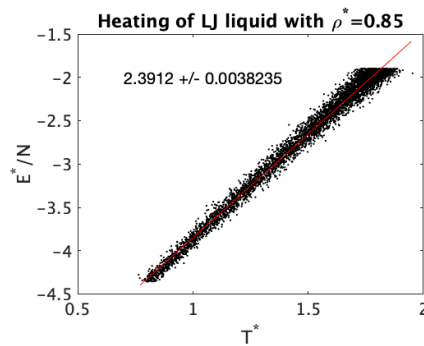


Figure 8: The liquid phase is clearly not an ideal gas!!

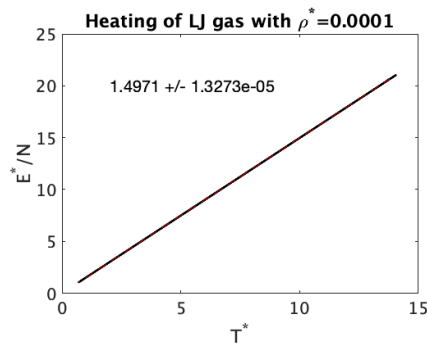


Figure 9: From triple point temperature to very high supercritical.

to be sure to get above the critical temperature. The simulation is now much slower because forces need to be calculated between many more atoms at each timestep. Figure 8 shows that it is clearly not an ideal gas since the slope is now 2.4!

Task 2

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

Solution:

Figure 9 shows that the slope does not change at all, even going to very high temperatures. This is not surprising because there are no extra degrees of freedom that get excited.

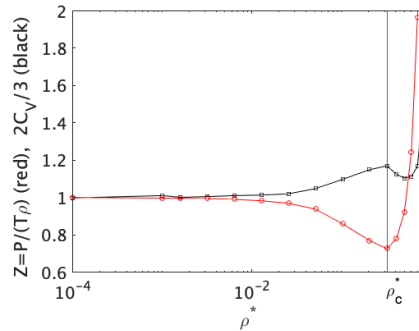


Figure 10: Density dependence of the compressibility factor and the heat capacity divided by the ideal gas value $3/2$.

Task 3

For a temperature above the critical temperature try densities from a dilute gas to the triple point. Do C_V and Z change with density?

Solution:

Now I need to do many short simulations. I have already done one at triple point density which was very different from the gas. I will start all simulations at $T^ = 1.4$ and try logarithmic change in density: $\rho = 0.0001, 0.001, 0.0016, 0.0032, 0.0064, 0.0128, 0.0256, 0.0512, 0.1024, 0.2048, 0.32, 0.4096, 0.5, 0.6, 0.7, 0.85$.*

In figure 10 one observes that both the curves have local extrema at the critical density! In addition, the compressibility curve shows that at intermediate densities the supercritical fluid is dominated by attractive interactions, whereas above $\rho = 0.5$ the repulsive interactions dominate.

Task 4

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?

Solution:

Figure 11 shows that the slope is 1.74 and does not change with temperature. Since it is classical I expected all degrees of freedom to be excited all the time. Then I expected $C_v = 7/2 = 3.5$. Visualization in Ovito shows that the molecules are spinning like crazy from the start. Then they are moving faster and faster in their translational trajectories. Maybe they do not collide enough to exchange energy between the degrees of freedom? Maybe the heating rate (eFlux=10) is

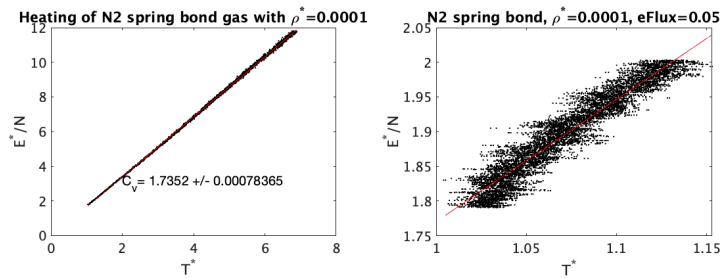


Figure 11: Heat capacity experiment for N2 model with spring and eFlux=10 on the left side and eFlux=0.1 on the right side

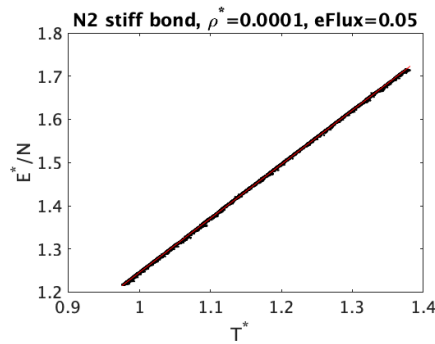


Figure 12: 2 million timestep run with stiff N2 molecule and eFlux=0.05. Slope is 1.25.

way too fast? I'm trying again with eFlux=0.1. The slope is still 1.75.

To end off I started a run with rigid N2 molecules, eFlux=0.05 and 2 million timesteps. The very strange result is that the slope is 1.25, not 2.5. Ahh, one needs to multiply by 2 for the N2 models, because the energy is per atom, not per molecule!! So that means $C_v^/N = f/2 = 5/2$ as expected. Then the slope 1.75 gives $C_v^*/N = f/2 = 7/2$ for the flexible N2.*

A Models and experiments

Real materials consist of atoms, which are made of electrons, protons and neutrons, which are made of quarks and gluons, which are How can we do physics when matter is so complicated? The answer is that we can model the “cosmic onion” one layer at a time. Physics is the art of simplification, i.e., of ignoring those details that are irrelevant for what one has chosen to model.

This is possible because most of the details that are important (relevant) for microphysics are unimportant (irrelevant) for macrophysics. Consider the Solar system. Compared to a planet you are microscopic, and completely irrelevant for the planet’s trajectory through space and time. Planetary trajectories can be determined to very good accuracy by modelling the Sun and planets as points obeying Newtonian mechanics.

It is not unusual that macroscopic concepts have no microscopic meaning. An atom has neither pressure nor temperature. This phenomenon, that the whole (collective behaviour) is more than (or at least different from) the sum of its parts, is called *emergence*. That the “collective” (the gas) forgets the “personality of its individuals” (microscopic details of the molecules) is called *universality*. Without these concepts we cannot understand physics or any other natural science. *The prime example of this is thermodynamics and statistical mechanics.*

One of the purposes of this lab is to encourage you to reflect on what is important, and what is not, in thermodynamic gas theory. Our first task is find out how to model the molecules in a gas: what is relevant, and what is not? We wish to find out how thermodynamic variables like pressure and temperature capture the collective macroscopic behaviour of the myriad of microscopic constituents (atoms or molecules).

A.1 Comparing theoretical models with experiments

A comparison of experimental data and theoretical results is meaningless unless you have a “stick” to measure the distance between them. This measuring stick is the “error” or “uncertainty” of your measurement!

It makes no difference whether the theoretical model is analytical or numerical. You may be able to solve a sufficiently simple model analytically and thereby obtain exact theoretical values of observables, but this has no value unless you have experimental data with error bars that can be used measure how well the model simulates reality.

In the absence of “the untimely intrusion of reality” (experiments), no matter how hard you work on your model this will only teach you something about the model, nothing at all about the real world.

Furthermore, since all models have limited validity, it is not sufficient to only rely on the experimental data that led to the construction of the model in the first place: it may break down at any time, so you must keep checking the

model with new experiments adapted to your needs.

In other words, in physics an experimental number without units and error bars is worthless, and a theoretical number detached from reality is equally worthless.

B Counting degrees of freedom

We must distinguish between the number of mechanical degrees of freedom (f_{mech}) and the number of thermodynamic degrees of freedom (f), because they usually do not coincide at high temperature.

If the temperature T is significantly lower than the *characteristic temperature* $\Theta \simeq 1000K$ where the atoms in a molecule start to vibrate, then the molecule will behave like a rigid body. Three numbers are needed to specify the location of the center of mass (3 translational degrees of freedom). In addition there are at most three rotations of the molecule that can store energy, but if the molecule has one or more axes of rotational symmetry ($s > 0$), then these rotations cannot store energy, and the number of rotational degrees of freedom relevant for thermodynamics is $f_{rot} = 3 - s$. For $T \ll \Theta$ the number of “rigid” degrees of freedom is therefore given by

$$\boxed{f_{rig} = 3 + f_{rot} = 6 - s}, \quad (9)$$

where s is the number of rotational symmetries of the molecule. Every rigid degree of freedom contributes $R/2$ to the heat capacity. So, for rigid molecules (i.e., at low temperature[4]) the number of atoms does not matter, only which shape the molecule has (and its total mass).

Confusing vibes

This section is not relevant for this lab, but is intended to clarify a topic of much confusion that you may encounter elsewhere.

If $T \gtrsim \Theta \simeq 1000K$ we must include other degrees of freedom. The total number of *mechanical* degrees of freedom for n atoms is always $f_{mech} = f_{rig} + f_{vib} = 3n$, because we need three coordinates to determine the position of each of the n points, no matter how they move. The number of mechanical vibrational degrees of freedom is therefore $f_{vib} = 3n - 6 + s$. Each of these can be modelled by replacing the rigid rods between pairs of atoms with springs, i.e., harmonic oscillators.

Each vibration mode contributes an amount R to the heat capacity, so we can write the total heat capacity as $c_V = fR/2$, where we will call $f = f_{rig} + 2f_{vib} = 6(n - 1) + s$ the number of *thermodynamic* degrees of freedom. At high temperature the number of thermodynamic degrees of freedom does *not* equal the number of mechanical degrees of freedom if $n \geq 2$: $f = f_{mech} = 3$ for $n = 1$, but $f > f_{mech} = 3n \geq 6$ for $n \geq 2$.

If these concepts are confused, as sometimes happens even in textbooks, then the counting of relevant degrees of freedom will be wrong. Since a vibrational mode can not be excited at room temperature, in this lab there will be *no* confusion: $f = f_{rig} = 6 - s$ for $T_{room} \ll \Theta \simeq 1000K$.

The reason that we have to double the counting of vibrational modes is that an oscillator has both kinetic and potential energy. If we model the bond between two atoms with a spring of a given stiffness ν_k (which is determined by how strong the bond is), then the molecule can store potential energy proportional to ν_k in the spring when it is stretched or compressed. At sufficiently high temperature each spring can store equal amounts of kinetic and potential energy (the *equipartition theorem*), so each vibration contributes twice as much as translations and rotations to the capacity of the gas to store energy (the heat capacity c_V).

We define Θ_k ($k = 1, 2, \dots, f_{vib}$) to be the *characteristic temperature* that must be exceeded in order to excite the vibration mode labeled by k . The value of $\Theta_k \propto \nu_k/k_B$ is determined by the spring constant ν_k . For normal molecules Θ_k is over a thousand degrees, and by “room temperature” we mean $T_{room} \ll \Theta_k$ (for all k).

C Thermistor physics

The purpose of this part of the lab is to emphasize that the apparatus (sensors) we use for measurements also are physical systems. They are therefore only useful to the extent that we understand their physics. Often we use tables and graphs to convert the independent variable we actually measure to the dependent variable we need.

A good example that we have encountered before is the *Hall effect*, which appears when an electrical current in a solid encounters a magnetic field. This is an interesting phenomenon that we studied in FYS1120: *Electro-magnetism* to get a better understanding of both electromagnetism and the (quantum mechanical) band structure of semiconductors. Having understood the physics of this phenomenon, we can then use it to make devices that measure magnetic fields with great precision, by measuring the Hall potential transverse to the current. Such *Hall probes* are now widely used, and so cheap and tiny that you probably have a handful in your phone.

To give a quantitative comparison of our sound data with thermodynamic gas theory we must be able to measure the temperature accurately. We do this by measuring the electrical resistance R_t of a particular type of semiconductor called a *thermistor*. The “thermistor function”, the basic form of which can be derived from thermodynamics,

$$\boxed{T_C(r) \approx 25 - 24 \ln r}, \quad (10)$$

where $r = R_t/(10^5 \Omega)$ and R_t is the Ohmic resistance of the thermistor,[6] is

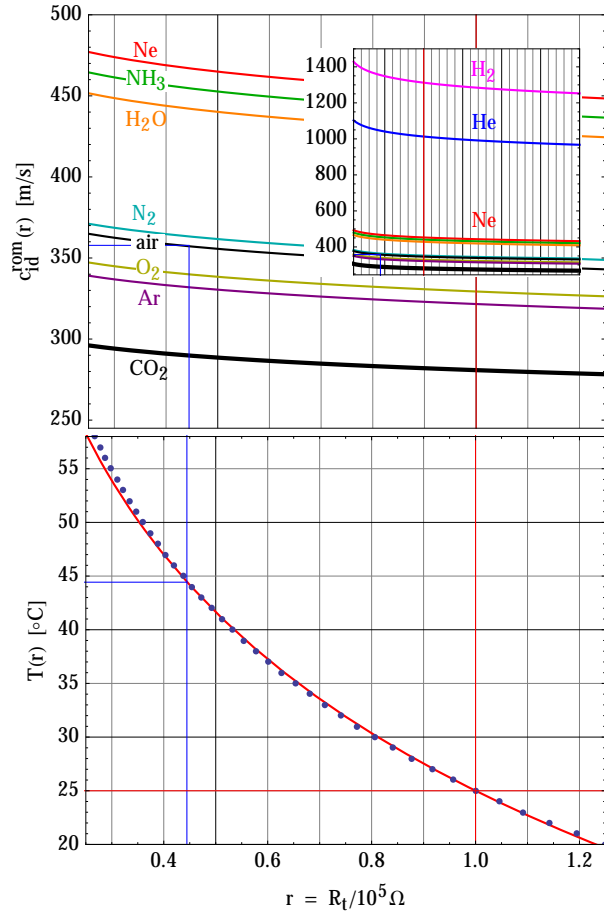


Figure 13: By measuring the electrical resistance $R_t[\Omega]$ in a thermistor, we can read off the temperature $T_C [^\circ C]$ from this diagram. Bottom: The red graph is our approximate (empirical) thermistor function $T_C(r) \approx 25 - 24 \ln r$ [$r = R_t/(10^5 \Omega)$], which has been fitted to the manufacturer's table (blue dots).

plotted in Fig. 13. Notice that it gives the temperature in *Celsius* ($^{\circ}C$), not in *Kelvin* (K)!

We see that this empirical formula (best fit to manufacturer’s data over a small range of temperatures) deviates slightly from tabulated values at high temperatures, but since it fits very well (with the uncertainties of our measurements) in the temperature range we are going to study, it is sufficient for our purposes.[6]

C.1 Linear models

In this lab the objective is to use a little knowledge from wave mechanics and a few measurements to construct a data list consisting of pairs of numbers, which can be thought of as points in a plane. Your task is to use this list to find the most probable value (the best estimate) \bar{c} of the speed of sound in a gas, and the uncertainty δc of this estimate.

The simplest way to estimate \bar{c} is to use a ruler. This is a slightly vague but very graphic way to illustrate how a line is fitted to a set of data. After plotting the data points on a plane a transparent ruler is placed on top of the paper in such a way that the data points are spread out “as evenly as possible” on both sides of the edge of the ruler. Intuition dictates that this is the “best fit”. Linear regression is one way to make this intuition precise. All we need is a simple way to measure how “evenly” the points are spread out.

Notice that you are using the whole data set, and therefore all available information, when you shift and twist the ruler, and this is clearly a necessary requirement for a good fit. Notice also that only in rare cases does a data point sit right on the line, and it is usually not a good idea to “connect the dots”, since this may be misleading as it does not combine the data set in a physically meaningful way.

A ruler is a good way to get a rough idea of the fitted line, but in reality we also use a computer to make this procedure quantitative. It tries out “all possible” lines $y = ax + b$ by changing the slope a and intercept b (constrained to a finite number by some built in numerical resolution). For each choice of line the sum of the squares of the (vertical) distances of the data points to the line is calculated. By definition, the winner (i.e., the “best fit”) is the line with the smallest sum of squares. The slope \bar{a} of this line contains the information about the best estimate of the speed of sound. (In other experiments we may also be interested in the best estimate \bar{b} of the intercept, but not here.)

The spread of the data points around the line gives us the standard deviation. If you do not already have a favourite application that fits a line and calculates the uncertainty of this estimate automatically, you may wish to use these two lines of PYTHON.

```
x=[1,2,3,...,n]
y=[f1,f2,f3,...,fn]
```

```
from scipy import stats
stats.linregress(x,y)
```

It does not get any simpler than that.

Without a universal line-fitting tool you cannot do physics, so if this is not already hardwired into your brain you should have that done now!

In the final part of this lab you are going to compare your experimental results with the theory of ideal gases, which asserts that $c \propto \sqrt{T}$. You must therefore also estimate the most probable value \bar{T} of the average temperature inside the tube when you obtained the data. This comparison is meaningless unless you can estimate the uncertainty δc of the estimate \bar{c} , and the uncertainty δT of your estimate \bar{T} .

It has no meaning to say that two numbers are “near” each other unless you have a “measuring stick” to measure the distance between these numbers.

Is the estimated value $\bar{\pi} = 3.1415$ of the circumference to diameter ratio of any circle (obtained by wrapping wires around circles of many different sizes, say) “near” the exact value $\pi = 3.1415926535897932\dots$ (exact if you knew all the dots), even if there are infinitely many numbers between these two (always!) distinct numbers? *The measuring stick is the variance (standard deviation) of the estimated value, so if you misplace this stick you have nothing!* If the uncertainty in the estimate of $\bar{\pi}$ is $\delta\pi = \pm 0.001$, then $\bar{\pi}$ and π must be treated as the same number in physics, because we have no empirical information that allows us to say otherwise. If the uncertainty in the estimate $\bar{\pi}$ is $\delta\pi = \pm 0.0001$, then $\bar{\pi}$ and π should be treated as different numbers in physics, because we do have empirical evidence allowing us to say that it is very improbable that they actually are the same number. This conclusion is not absolutely certain, but absolutes have no place in science. Our ambition is to know how uncertain our knowledge is, not to find “absolute truth”.

The “uncertainty” in data that comes from unavoidable statistical variations (often called “errors”, unfortunately) can be made as small as you can afford, by collecting more data. How big must the deviation be before we can say that the data do not support the model? There is no right answer to this question, but at least in particle physics the convention is that if the discrepancy is more than 5 standard deviations (“sigma”) (the probability that this is a random statistical fluctuation is less than 1 in 3.5 million), then there is a real problem, and usually the model (theory) is in big trouble. However, sometimes the lack of agreement is caused by unknown *systematic errors* that often derive from an inadequate understanding of the measuring apparatus. If this is the case, then an improved experiment (rather than an improved theory), which actually measures what we think it is measuring, is what is needed. Systematic errors is the Achilles’ heel of any experiment, because there is no systematic way of identifying their sources.

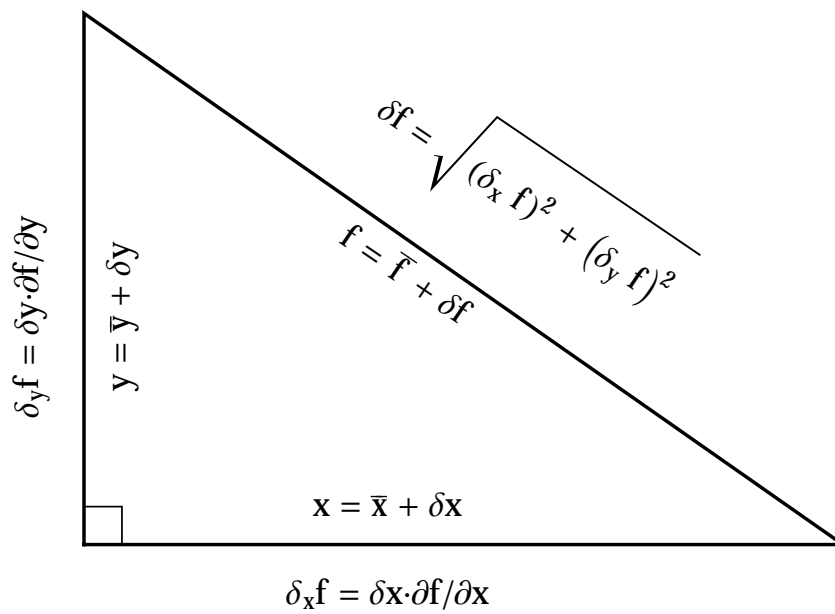


Figure 14: This “Pythagorean uncertainty triangle” is a mnemonic for addition of uncertainties.

C.2 Pythagorean error propagation

When the best line has been found we can calculate the “spread” δa (*variance, standard deviation*) of the slope \bar{a} of this line.

The standard deviation is defined in such a way that if you repeated the exact same experiment many times, then you would find

$a \in \langle \bar{a} - \delta a, \bar{a} + \delta a \rangle$ in ca. 68.27% of the experiments, $a \in \langle \bar{a} - 2\delta a, \bar{a} + 2\delta a \rangle$ in ca. 95.45% of the experiments, $a \in \langle \bar{a} - 3\delta a, \bar{a} + 3\delta a \rangle$ in ca. 99.73% of the experiments, $a \in \langle \bar{a} - 4\delta a, \bar{a} + 4\delta a \rangle$ in ca. 99.994% of the experiments, $a \in \langle \bar{a} - 5\delta a, \bar{a} + 5\delta a \rangle$ in ca. 99.9999% of the experiments, etc.

The best estimate of the speed of sound is $\bar{c} = 2\bar{L}\bar{a}$, but what is the uncertainty δc of this estimate? More generally: *what is the uncertainty in the value of a function $f(x_1, x_2, \dots)$ of one or more independent stochastic variables x_1, x_2, \dots , which each has an uncertainty $\delta x_1, \delta x_2, \dots$?*

Each uncertain variable x_k contributes to the uncertainty δf of f , but less than you might naively think. Heuristically, if all the measured values are independent, then they “pull in orthogonal directions”, and should therefore not be added linearly, which is the naive guess. That would give an excessively large estimate of δf .

If there is only one variable, then the uncertainty in f is found by differentiation, $\delta f = |df/dx|\delta x$. If there are two or more variables each one contributes via partial derivatives, but these should be added “in quadrature”. For two variables a useful mnemonic is the “Pythagorean uncertainty triangle” shown in Fig. 14. The uncertainty δf of the best estimate $\bar{f} = f(\bar{x}, \bar{y})$ is given by the hypotenuse, which is smaller than the sum of the legs,

$$\delta f = \sqrt{(\delta_x f)^2 + (\delta_y f)^2} < \delta_x f + \delta_y f.$$

(More variables may be accommodated by an obvious generalization of this formula.) *Example:*

$$f(x, y) = xy \implies \frac{\delta f}{f} = \sqrt{\left(\frac{\delta x}{x}\right)^2 + \left(\frac{\delta y}{y}\right)^2}.$$

We see that it is the *relative uncertainties* $r_x = \delta x/x$, etc. that are relevant.

Since they are squared, a relative uncertainty that is significantly smaller than the others will not contribute much to the relative uncertainty of f . In this case we can drop one variable, and the equation simplifies to a much used form, $\delta f \approx y\delta x$. Fig. 15 shows how fast the contribution from the least significant variable, here $r_y = \delta y/y$, “dies” compared to $r_x > r_y$.

So, if you decide to use $\delta c \approx 2L\delta a$ to calculate the uncertainty in the speed of sound, then you must justify this by verifying that the relative uncertainty in a is much larger than the relative uncertainty in L .

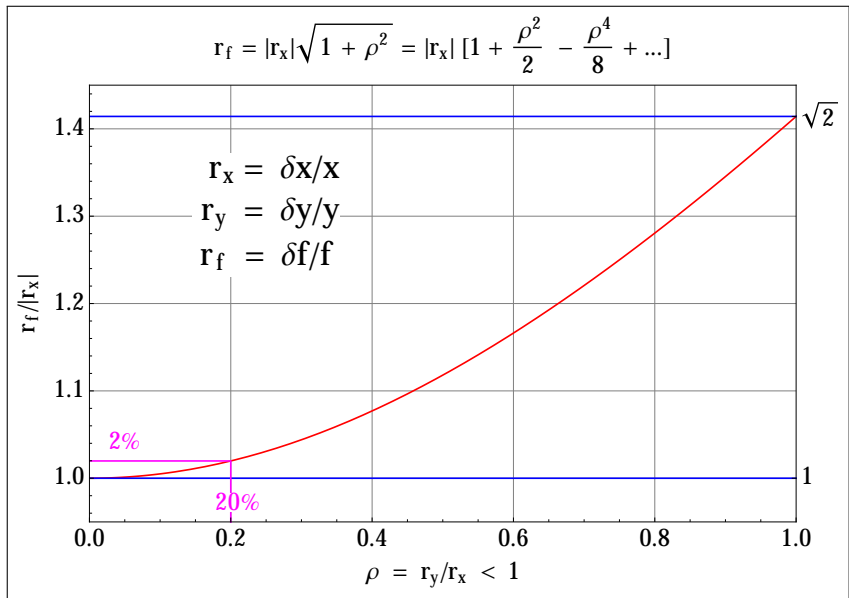


Figure 15: Relative uncertainty when $f = x \cdot y$ and $r_y < r_x$. Notice that if r_y is 20 % of r_x , then it contributes only 2 % to the relative uncertainty of f . If r_y is 10 % of r_x it contributes less 0.5 %.

C.3 Nonlinear models

The analytic process we have used here to estimate the speed of sound in a gas is typical. The method is the same for all linear functions, $f(x) = ax + b$. This is more general than it looks, because we can often swap a nonlinear function for a linear one by a change of variables. Some examples are:

- $f(x) = a/x + b$: define $z = 1/x$ and study instead $g(z) = az + b$.
- $f(x) = c \exp(ax + b)$: take the logarithm on both sides and study instead $g(x) = \ln f(x) = ax + \tilde{b}$, where the new constant is $\tilde{b} = b + \ln c$.
- $f(x) = c \ln(ax + b)$: exponentiate both sides and study instead $g(x) = \exp f(x)/c = ax + b$.
- $f(x) = (ax + b)^c$: take the root on both sides and study instead $g(x) = \sqrt[c]{f(x)} = ax + b$.

If it is the uncertainty δx of a variable x that is known (usually determined by a separate fitting) then you have to use the Pythagorean method, even if there is only one independent variable x , to find the uncertainty of any quantify that is a function of x . For example,

$$f(x) = ax + b \implies \delta f = \left| \frac{\partial f}{\partial x} \delta x \right| = a \delta x,$$

while (see above list),

$$\begin{aligned} f(x) = a/x + b &\implies \delta z = \left| \frac{\partial z}{\partial x} \delta x \right| = \frac{\delta x}{x^2} \\ &\implies \delta g = \left| \frac{\partial g}{\partial z} \delta z \right| = \frac{a}{x^2} \delta x. \end{aligned}$$

If it is the uncertainty δz of a transformed variable $z = z(x)$ that is known, usually by fitting a linear function $g(z) = az + b$, then you should use the Pythagorean method on $g(z)$, not $g(x)$: $\delta g = a\delta z$. The uncertainty in x is then $\delta x = |dz/dx|\delta z$. For example, if $z = 1/x$ then $\delta z = \delta x/x^2$ and $\delta x = \delta z/z^2$.

D Information and tips on using Lammops in Lab1

This appendix contains some practical information about the molecular dynamics simulations of Lab 1, with particular focus on LAMMPS. You will find practical information about how you should change the input files in order to run the experiments appropriately, and why these changes are made. In addition to that, you will find info on how to properly keep track of all the data, and how to analyze the output of LAMMPS using `python`.

- **Important:** Before doing anything at the lab, download the latest files from the website. This is crucial, as material may be updated frequently.

D.1 Notation and operating systems

Since I'm using Linux, most of this document is written according to this operating system's commands. If you use Windows or Mac, that's fine, but please modify your commands accordingly. Use the appropriate command for your software, e.g. `lmp` or `lmp_serial`.

D.1.1 Necessary software

Make sure LAMMPS is working on your computer. If you haven't tested LAMMPS already, you may use the script `myfirstmd.in` from the course web page. Move to the same directory as the file you downloaded and run the following from the terminal: `lmp_serial > myfirstmd.in`. If it runs without any warnings, and the two files `log.lammps` and `dump.lammpstrj` appear in your directory, everything should be working fine. If not, please notify the teaching assistant.

D.1.2 Keeping track of files

Every year there are several students who lose data by overwriting already existing output, and they lose track of what they have done in previous simulations by using a single input file for all tasks.

- **Input files:** Before modifying and running any simulations in LAMMPS, always make a copy of the `.in` file. This allows you to revisit and modify simulations easily. By *not* modifying a single `.in` file for each simulation, you prevent potential errors from propagating through your previous work unknowingly. Additionally, maintaining control over input files ensures greater reproducibility in your scientific work. If you ever doubt your results, having the old `.in` file allows you to rerun any simulation with ease. For example, in Task 2, we will create a file named `heatcap1j_task2.in`, which we use for that Task 2 only.

- **Output files:** Whenever you run a LAMMPS simulation, it generates two files, `log.lammps` and `dump.lammpstrj`. Subsequent simulations will overwrite these files if they have the same names. Given that you'll require outputs from various simulations in this lab, it's vital to prevent overwriting. To achieve this, **always rename the output files after each simulation**. For instance, after running the simulation for Task 2 with `lmp_serial > heatcaplj_task2.in`, rename the `log.lammps` file to `log.lammps_task2`. Explicitly naming both input and output files according to the task name makes tracking of which input corresponds to which output very easy.

D.1.3 Reading the logfiles

To read the logfile, and analyze the outputs, we will use the `lammps-logfile` package in Python. You can see basic examples of its usage here: <https://github.com/henriasv/lammps-logfile>. To install the package, simply run

```
pip install lammps-logfile
```

In section D.2 we provide an example for how to use this package.

D.1.4 Plotting

When plotting, remember that your data are discrete sets. If you're using `matplotlib` to plot e.g. `x` and `y` with `plt.plot(x,y)`, `matplotlib` will interpolate the data and draw connected lines between each point. This can be extremely misleading, and in worst case you may end up with a wrong interpretation and conclusion. In these labs you will be dealing with simulation outputs, and you should **always** plot the true data points explicitly. As long as you're doing this, you can also draw lines connecting these points. For the tasks where there are many data points, you can adjust the individual dots and connecting lines easily to avoid clutter. Here are three examples using `matplotlib`:

```
plt.plot(x,y, 'o', ms=1)
plt.plot(x,y, 'o-', ms=1)
plt.plot(x,y, 'o-', ms=1, lw=0.7)
```

In the first example we only draw single circles at the data points with `'o'`. With `ms=1` (`markersize`), we reduce the size of the circles (I don't know the default value, but it's larger than 1). In the second and third examples we draw lines connecting each point. The connecting lines in the latter example have a reduced line width (`lw`). The default is `lw=1`.

D.2 Tasks in MD part of lab

In each of the tasks below, you are given information on what to adjust before each simulation. In your final report, you should include a brief explanation of

what you modify for each experiment, and *why* you modify it. For most of the tasks, you should be able to justify the modifications using the phase diagram given in the problem text.

Task 1

In this part, we will run a Lennard-Jones system to compute the C_V and the compressibility, $Z = P/(T\rho)$. Before doing anything, make a copy of the input file `heatcaplj.in` -> `heatcaplj_task2.in`. Now make the following modifications to `heatcaplj_task2.in`:

1. We want to ensure a gaseous phase, so we will reduce the density from 0.01 to 0.001.
Line 7: `variable rho equals 0.001`
2. To get a smoother output we increase the number of iterations of the simulation from 10000 to 100000.
Line 64: `run 100000`
Note: Don't change the other `run` parameters
3. Run the simulation: `lmp_serial > heatcaplj_task2.in`
4. Change name of the output: `log.lammps` -> `log.lammps_task2`

To analyze the data, we first see which physical quantities are available, using the `lammps-logfile` package.

```
import lammps_logfile

log = lammps_logfile.File("path/to/logfile/log.lammps_task2")
print(log.get_keywords())
```

The resulting output should be:

```
['E_mol', 'E_pair', 'Press', 'Step', 'Temp', 'TotEng']
```

To get temperature and energy arrays from the log-file, we simply load them with

```
T = log.get("Temp")
U = log.get("TotEng")
```

Note that in some simulations, LAMMPS may use a step to "calibrate" the system. It may therefore be a good idea to omit the first point of the data, by simply adding a `[1:]` after the parentheses.

Since we are considering an ideal gas in this exercise, there is a simple relationship between the temperature and energy of the system. Hence, you should be able to compute C_V and estimate the number of degrees of freedom, f , using only `scipy.stats.linregress`, i.e. without numerical differentiation.

Task 2

In this task, we do *not* want to run multiple simulations with increasing temperature. Instead, we will modify the input such that the temperature increases sufficiently within a single run.

1. Make a copy of the in-file from task2
`heatcaplj_task2.in -> heatcaplj_task4.in`
2. Change initial temperature from 1.0 to T_t^* . See figure 6.
Line 3: `variable T equal 0.????`
3. Change eFlux from 1 to 10, i.e. add ten times as much heat each timestep
Line 11: `variable eFlux equal 10`
4. Double the number of runs from last exercise to allow the temperature to increase sufficiently
Line 64: `run 200000`
5. Run the simulation and change the name of the outputs
`lmp_serial > heatcaplj_task4.in`
`log.lammps -> log.lammps_task4`

Repeat the analysis from task 2. Use the phase diagram to explain what you observe. How does the result change if you change the initial density?

Task 3

Now we are going to test increasing densities. For this, we will use the configuration of Task 2 as our starting point. Since we set the density at the beginning of the simulation, we have to perform one simulation per density we test. To get reasonable results, you need to at least test 10 different densities. Don't use $\rho \geq 0.85$. (See figure 6, $\rho_{t,l} = 0.85$) You therefore need ten different in-files, resulting in ten different out-files. To keep track of all the files, start by making a directory named `task5_infiles`, and copy the in-file from Task 2 (not Task 4) into that directory. Additionally, you should indicate the density index in the filename. For the first simulation, the setup is as follows:

1. `heatcaplj_task2.in -> task5_infiles/hatcaplj_task5_rho1.in`
2. Choose a temperature above the critical temperature
Line 3: `variable T equal 1.4`
3. Ensure dilute gas initially. Exact value not important
Line 7: `variable rho equal 0.001`
4. Line 64: `run 100000` (Double check, otherwise it will take a while to run everything.)

For N different values of ρ_i , the recipe is as follows (`cp` is the command for copying in Linux/Mac):

1. Run the first simulation
`lmp_serial > heatcaplj_task5_rho1`
2. Make a copy of the output file with appropriate name
`log.lammps log.task5_rho1`
3. Make a new input file for the second run
`heatcaplj_task5_rho1 heatcaplj_task5_rho2`
4. Change density of `heatcaplj_task5_rho2` to the new value
Line7: `variable rho equal ...`
5. Run the new configuration
`lmp_serial > heatcaplj_task5_rho2`
6. Rename the new output file
`cp log.lammps log.task5_task5_rho2`
7. Repeat until you have N input and output files, one for every density.

Renaming the output log-files is crucial in this exercise, as we need all N to get appropriate results. If your initial choice of ρ values don't give desirable results, you can update the existing `.in` files and repeat the simulations. Perhaps you only have to change a few samples. But, make sure to rename the output files if you redo any simulations.

Using the N output files, you can loop through them and compute C_V and Z as a function of ρ . Tips for easier comparison: Use your previous results to normalize C_V and Z such that both quantities start at 1.

Task 4

In this exercise, you will perform a similar simulation, but with N_2 molecules, rather than point particles. Download the `heatcapljdiat.in` file from the course web pages. Now, we will make two copies.

- **Modeling the bond between the two atoms as a rigid rod**

Make a copy of the original file:

```
heatcapljdiat.in > heatcapljdiat_rod.in  
Line 13: variable rigidbond equal 1
```

- **Modeling the bond between the two atoms as a spring**

Make a copy of the original file:

```
heatcapljdiat.in > heatcapljdiat_spring.in  
Line 13: variable rigidbond equal 0
```


The input files read information about the atoms from a separate file, `diat.molecule`. Download the file from the course page and place it in the same directory as the two `.in` files.

In order to determine the number of degrees of freedom for the molecule, it will be sufficient to study the two molecules in a gaseous phase, i.e. at a low, fixed density value (Why is that). Using what you've learned so far, you should be able to perform an appropriate simulation of the two molecules to answer the question. Does the value of f you compute coincide with what you expected for the two molecules?

Hint: *What units do LAMMPS use for the output?*

References

- [1] A. Kundt (1866). “Über eine neue Art Akustischer Staubfiguren und über die Anwendung derselben zur Bestimmung der Schallgeschwindigkeit in festen Körpern und Gasen”. *Annalen der Physik* (Leipzig: J. C. Poggendorff.) **127** (4).
- [2] The NIST Reference on Constants, Units, and Uncertainty. US National Institute of Standards and Technology. 2014 CODATA recommended values. OBS: SI units were recently redefined.
- [3] Rather than acquiescing to the theoretical bias $b = 0$. This is actually the least biased thing to do, because we have made many assumptions about the geometry of the apparatus to arrive at Eq. (7). Since it is the longest wavelengths that are most sensitive to global (geometric) features, we should expect low frequency data to deviate somewhat from the simple linear relation in Eq. (7), and they do, so b is a legitimate and necessary fitting parameter.
- [4] By “low” temperature we mean here $1K \ll T \ll 1000K$. In this case phase transitions are determined by classical thermal fluctuations. At really low temperatures ($T \ll 1K$) quantum phase transitions are possible that are driven by quantum fluctuations. These are of interest in future electronics, including quantum computers.
- [5] K is also called the *bulk modulus*, or *Young’s elasticity modulus* in three dimensions, since it parametrizes volumetric elasticity.
- [6] A log-linear fit of $T_C(r) = 25 - b \ln r$ to the factory table gives a slope $\bar{b} \approx 23.9548$. We will here use $\bar{b} \approx 24$, since this gives temperatures that deviate from the table by less than $\pm 0.05^\circ\text{C}$. The much more complicated standard empirical (Steinhart-Hart) equation $T_K(R) = 1/(a+b \ln R+c \ln^3 R)$ usually found in the literature requires three fitting parameters (a , b , and c), and the fit is no better over the small range of temperatures we are probing here.
- [7] This is not a purely academic exercise. To avoid a climate catastrophe we wish to store the greenhouse gas CO_2 inside the planetary crust. The thermodynamics of CO_2 and mixtures of CO_2 with other gases is therefore of considerable interest.