SI2530 Computational Physics: Molecular dynamics simulation of a Lennard-Jones Fluid

October 15, 2010

1 Background

In this computer exercise you will use a quite simple molecular dynamics program to simulate a Lennard-Jones fluid in the Gibbs ensemble (constant N, P and T) and compare the results with experimental data for argon (Ar).

The Lennard-Jones fluid is a very simple system. It is defined by two independent parameters, σ and ϵ via the Lennard-Jones potential,

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

The attractive term is the van der Waals' interaction and its functional form ($1/r^6$) comes from quantum-mechanical perturbation calculations for non-polar, neutral atoms with a spherically symmetrical electron shell, i.e noble gas atoms. The functional form of the repulsive part is not known exactly but is approximated by either an exponential or an inverse power. The number 12 in the Lennard-Jones potential is used for computational efficiency since $1/r^{12}$ is the square of $1/r^6$.

1.1 Reduced units

It's important to use a suitable set of units, e.g. to minimize the effects of round-off errors. In this lab you will use what is called *reduced units*, which are dimensionless. They are derived from the potential parameters, ϵ and σ, the mass M of a single atom and Boltzmann's constant, k_B. This means that all energies are given in units of ϵ and all lengths in units of σ. The reduced temperature is given by k_BT/ϵ and so on (see Table 1).

Question 1: What is the appropriate scaling form for the reduced pressure? What is the scaling form for the reduced velocity? Fill in the missing expressions in Table 1.

	Reduced units	Relation to SI
Length	r^*	$r\sigma^{-1}$
Energy	E^*	$E\epsilon^{-1}$
Temperature	T^*	$k_BT\epsilon^{-1}$
Time	t^*	$t\sigma^{-1}\sqrt{\epsilon/M}$
Mass	m^*	mM^{-1}
Force	F^*	$F\sigma\epsilon^{-1}$
Density	ρ^*	$N\sigma^3V^{-1}$
Pressure	P^*	
Velocity	v^*	

Table 1: Reduced units

2 How to do it

2.1 Preparations

Download the file `labmdar.tar` from the course home page[1] and untar it with the command `tar xvf labmd.tar`. Check out the file `README` to find out more about the different files you have now unpacked.

Now it is time to compile the program. Type `make` to compile the program (if you like you may check out the file `Makefile` for the actual compiler commands).

2.2 Running the program

The program `md` needs information from two files: a parameter file and a coordinate file. The parameter file `param.dat` contains a number of different parameters; number of MD-steps, size of the time step, values for temperature and pressure, frequencies for writing output to file and so on. Note that some of the values are intentionally set to zero – you will change them later on. The time step you can probably leave as it stands, 0.001 in reduced units which is about 2 fs (for argon). The temperature coupling algorithm is called *Berendsen coupling.* Each time step the velocities are scaled with a factor that is calculated from the relative difference between the computed temperature and a reference temperature, TREF which you assign a value in `param.dat`. You also need a value for the time constant, TAUT, which determines how fast the scaling should be; the smaller the time constant the faster the simulated temperature goes towards the reference temperature. If TAUT is set to zero, no coupling is applied. There is one major drawback from this algorithm; it does not produce a true canonical ensemble. It works well for many purposes for instance to calculate averages. But it does not necessarily give the correct fluctuations: This makes it problematic if you for example were to calculate C_V from the fluctuations in energy in the canonical ensemble. Obviously the result would depend upon the choice of time constant in this case. There are better ways of dealing with constant temperature, but it is somewhat outside the scope of this exercise. The pressure coupling algorithm is similar to the temperature coupling and it has the same name. It uses one more parameter, COMPR, which is the isothermal compressibility.

[1] http://courses.theophys.kth.se/SI2530

The coordinate file contains positions and velocities for 500 atoms on an *fcc* lattice. The columns are: x, y, z, v_x, v_y, v_z. The first two lines are the number of atoms and simulation box side length respectively. As you can see all velocities are initially zero which means that the first time you simulate the system you need to give each atom a random velocity corresponding to the desired temperature. This is controlled by the parameters GENVEL and SEED in the parameter file.

The program produces three output files: A coordinate file with the end structure (positions and velocities), an energy file containing various quantities as functions of time (see Table 2) and a trajectory file containing all positions and velocities as a function of time. The parameters EOUT and XVOUT controls how often these files are written.

The file names are controlled by a number of command line options. To find out what they are, type `md -h`. As you can see you can name your input and output whatever you like or stick to the default options.

Col. no.	Name	Quantity
1	STEP	Time step
2	TIME	Time
3	EPOT	Potential energy
4	EKIN	Kinetic energy
5	VIRIAL	Virial
6	BOXL	Simulation box side length
7	KPRES	Kinetic pressure
8	VPRES	Virial pressure
9	PRES	Total pressure ($p_{kin} + p_{vir} + p_{corr}$)
10	TEMP	Temperature

Table 2: The order of the columns in the energy output file.

2.3 Making the analysis

We have provided a sample Matlab script (`analyze.m`) that reads the energy file and plots the temperature as a function of time. You can use this script and modify it according to your needs. Instead of Matlab we will use the open source alternative Octave. The syntax of Octave is essentially equivalent to that of Matlab. To run a script using Octave you can either first start Octave by typing `octave` and then the name of the script or simply type just `octave --persist -q <scriptname>`.

Analyzing the trajectory is a bit trickier – we need it to calculate correlation functions. For that purpose we have provided a fortran program called `analyze`. This program takes one command line argument: the name of a trajectory file and produces three output files: `rdf.dat`, `r2t.dat` and `vacf.dat` which contains the pair distribution function, $g(r)$, the mean square displacement, $\langle(\mathbf{r}(t)-\mathbf{r}(0))^2\rangle$ and the velocity auto correlation function, $\langle\mathbf{v}(0)\mathbf{v}(t)\rangle$. The output can be plotted with Octave or Xmgrace, with the command `xmgrace`. Xmgrace is an excellent interactive plotting and analysis program. To use Xmgrace for plotting instead of Octave (which uses Gnuplot), save the figure data as a text file after generating it in Octave.

3 What to do

You will start by simulating the solid state. Choose a low temperature, 30–40 K (don't forget to use reduced units), and start a short simulation of about 1000 steps. Assign the system random velocities and use T-coupling but no p-coupling. Look at the temperature versus time.

Question 2: Does the thermostat work well? Plot and explain.

Make another short run, this time with the pressure coupled to a reference pressure of either 1 or 0.1 MPa and with the random velocity generator turned off. Watch the box side length which is printed to standard output during the simulation. Does it seem to stabilize? If not, run for another thousand steps. When you have your crystal in equilibrium, make a production run of a few thousand steps.

Question 3: Does the pair distribution function look like expected for a crystal? Plot and explain.

Now it is time to melt the crystal. In principle, you could just raise the temperature to about 90 K and start a simulation, but it seems that this would take too long. Instead, raise the temperature to about 200 K and make simulations of, say, 1000 steps at a time. Watch the box side length and the pair distribution function. When the system has melted, chose a temperature from the table in the appendix and edit the parameter file (T and p) and start equilibrating the system. This will take approximately 10000 steps. Look at the temperature, the pressure, the side of the box and the potential energy to make sure the system has reached equilibrium. Next, make a full production run of at least 10000 steps for the analysis.

Question 4: Does the pair distribution function look like expected for a liquid? Plot and explain.

The calculated pressure consists of three parts: the kinetic pressure (ideal gas pressure) which is positive, the virial pressure (which comes from the fact that the atoms interact) which is negative and a correction term which approximates the contribution from outside the cut-off. Assuming the pair distribution function, $g(r) = 1$ for $r > r_c$:

$$p_{corr} = -\frac{1}{6}\left(\frac{N}{V}\right)^2 \int_{r_c}^{\infty} r\frac{d}{dr}\left(-\frac{4\epsilon\sigma^6}{r^6}\right) 4\pi r^2 \, dr = -\frac{16\pi N^2 \epsilon\sigma^6}{3V^2 r_c^3}.$$

Question 5: What are the signs and relative sizes of the three terms?

The temperature and the pressure are coupled in the simulations to specified reference values. Plot T and p as functions of time.

Question 6: Do the coupling algorithms seem to work properly?

Calculate the number density.

Question 7: How does it compare to the experimental value?

Plot the velocity auto correlation function (vacf).

Question 8: What does its value at time zero represent? Describe the behavior of the vacf at short, intermediate and long times.

Two ways of calculating the diffusion coefficient D is from the slope of the mean square displacement and from the integral of the vacf;

$$\langle(\mathbf{r}(t) - \mathbf{r}(0))^2\rangle \to 6Dt + l \text{ (for large times)}$$

and

$$D = \frac{1}{3}\int_0^\infty \langle\mathbf{v}(0)\mathbf{v}(\tau)\rangle \, d\tau \quad . \tag{1}$$

Calculate D using both methods.

Question 9: How do the results compare?

Question 10: Prove equation 1 analytically in the limit $t \to \infty$.

Use that

$$D = \lim_{t\to\infty} \frac{\langle[\mathbf{r}(t) - \mathbf{r}(0)]^2\rangle}{6t} = \lim_{t\to\infty} \frac{\langle[\int_0^t dt'\mathbf{v}(t')]^2\rangle}{6t}$$

and furthermore, due to the stationarity of the equilibrium ensemble and the time reversibility of the classical equations of motions

$$\langle\mathbf{v}(t')\cdot\mathbf{v}(t'')\rangle = \langle\mathbf{v}(t'-t'')\cdot\mathbf{v}(0)\rangle = \langle\mathbf{v}(t''-t')\cdot\mathbf{v}(0)\rangle \quad .$$

The heat capacity at constant volume may, in the *microcanonical ensemble*, be calculated from the temperature fluctuations:

$$C_V/(Nk_B) = \frac{3/2}{1 - \frac{3}{2}N\frac{\langle(T-\langle T\rangle)^2\rangle}{\langle T\rangle^2}}$$

Make a microcanonical run (put the time constants for temperature and pressure scaling equal to zero in the input file), calculate C_V and compare this to experimental data.

Question 11: Give a hand waving argument why this heat capacity is larger than for an ideal gas?

In the microcanonical ensemble the total energy is constant.

Question 12: How good is the energy conservation in your simulation?

Finally, if you have time, you could do one of following:

1. Calculate C_V in a different manner; make two (or more) simulations at constant volume but slightly different temperatures and take the numerical derivative of the energy with respect to T.

2. Calculate C_p from two or more simulations with constant pressure but different temperatures as a numerical derivative. For an ideal gas $C_p = C_V + R$. Does this relation hold?

4 Write a report

Write a short report ($\sim$ 5 pages, including necessary figures) that describes what you have done and includes the answers to all the questions. Convert all numbers into suitable SI-units. You may write in pairs. Take the opportunity to ask for help also if you get stuck on the theoretical parts. Send your report (preferably as a pdf document) by e-mail to `anatoly@kth.se`.

Appendix I

Some experimental figures for liquid argon

All data is taken from CRC Handbook of Chemistry and Physics, 88:th edition.

Argon (Ar), atomic number 18, mass 39.948 u

Potential parameters

$$\begin{aligned} \sigma &= 3.40 \ \text{Å} \\ \epsilon &= 1.654 \cdot 10^{-21} \ \text{J} \end{aligned}$$

Thermodynamic properties

T	p	ρ	C_p	C_V
85	0.1	2.122	5.37	2.78
85	1.0	2.126	5.36	2.78
90	1.0	2.080	5.37	2.60
100	1.0	1.982	5.56	2.39

The columns are:

1. Temperature (K)
2. Pressure (MPa)
3. Number density ($\text{m}^{-3} \cdot 10^{28}$)
4. The heat capacity per atom at constant pressure in units of k_B
5. The heat capacity per atom at constant volume in units of k_B

An experimental value for the (self) diffusion constant is $2,43 \cdot 10^{-5}$ cm^2/sec at 90K and 1.374 g/cm^3 (Journal of Chemical Physics **36**:2710, (1962)).

Triple and critical point

The temperature, pressure and number density at the triple and critical point are:

$$T_t = 83.81 \text{ K} \ \ p_t = 0.6895 \cdot 10^5 \text{ Pa} \ \ \rho_t = 2.136 \cdot 10^{28} \text{ m}^{-3}$$

$$T_c = 150.66 \text{ K} \ \ p_c = 48.60 \cdot 10^5 \text{ Pa} \ \ \rho_c = 0.807 \cdot 10^{28} \text{ m}^{-3}$$