Molecular Dynamics Simulations 2025 / Exercise 1 (SOLUTION)

1. Using lammps, create a diamond cell of Silicon 2x2x2 unit cells. To describe the interatomic interaction between particles in the system, use the Stillinger-Weber potential provided with the exercise instructions. Minimise it's positions, generate velocities using a gaussian distribution at 300 K, seed 123456 and preserving the linear and rotational momentum. Relax the system at 300 K in NPT for 30 ps and constant tau. Then relax it for another 30 ps using the NVT ensemble. Save thermo output every 100 steps. Plot the energy, pressure, and temperature as a function of time. In addition, for the last 30 ps of the simulation and using, compute the velocity autocorrelation function of the system (you can find help for doing it in lammps here `https://docs.lammps.org/compute_vacf.html`). The average can be sampled every 0.5 ps.

Below is the input script for lammps. It creates the atoms, sets the different parameters of the simulation, minimises it, samples velocities, and evolves it in NPT and NVT ensembles. The vacf is computed every 0.1 ps, but that can be filtered in the post-processing. Note that we use placeholders for the variables that we are gonna play with in the second part of the exercise.

```
log log_lammps_${N}_${dt}_${tau_T}

units metal
timestep 0.001
boundary p p p

lattice         diamond 5.43
region          box block 0 ${N} 0 ${N} 0 ${N}
create_box      1 box
create_atoms    1 box

timestep ${dt}

mass 1 28.0855
labelmap atom 1 Si

pair_style sw
pair_coeff * * Si.sw Si

min_style cg
minimize 1.0e-7 1.0e-8 100 1000
```

```
22
23 velocity all create ${temp} 4928459 mom yes rot yes dist gaussian
24
25 variable t equal time
26
27 compute 1 all vacf
28
29 thermo 100
30 thermo_style    custom step time dt temp pe ke press vol c_1[4]
31 thermo_modify   line one flush yes format 1 "ec %8lu" flush yes
32
33 fix myhalt all halt 100 v_t >= 30.0 error continue
34
35 fix NPT all npt temp ${temp} ${temp} ${tau_T} iso 0.0 0.0 1.0
36 run 10000000
37 unfix myhalt
38 unfix NPT
39
40 fix myhalt all halt 100 v_t >= 60.0 error continue
41 fix NVT all nvt temp ${temp} ${temp} ${tau_T}
42 run 10000000
43
44 write_data cell.data_${temp}
```

Using the script we obtained the figure 1 which shows the autocorrelation series of the velocities. Since we are working with a solid we can see that the function fluctuates around 0 periodically.

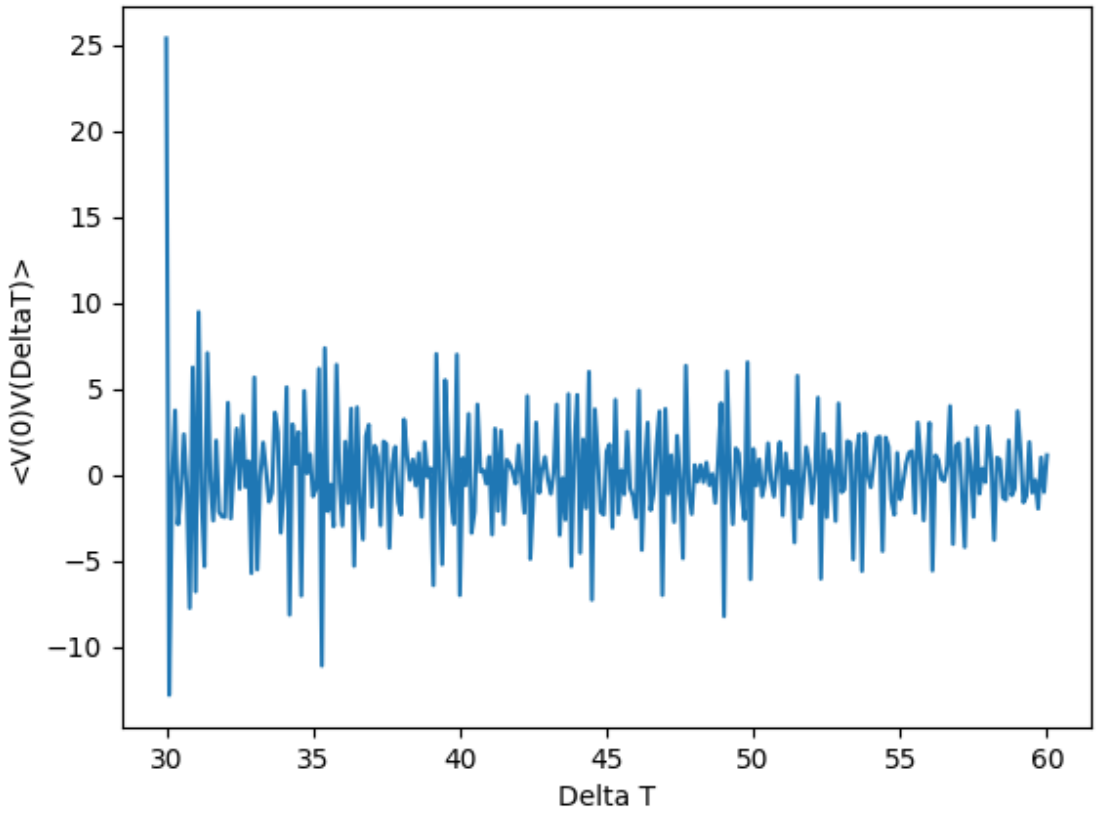

Figure 1: VACF of the solid during the last 30 ps, sampled every 0.5 ps.

2. Using the previous script convert all the previous parameters to placeholder variables, which can be parsed from the command line, and test the following:

- Check timesteps: 0.0001, 0.0005, 0.001, 0.003, 0.01 ps. Plot energy vs timestep for each. Comment on what you observe.
- Monitor the temperature as a function of time for different temperature relaxation constant. Do you see any difference?
- Monitor pressure as a function of time for different simulation cell sizes (N=1, N=2, N=4, N=8). What conclusions do you draw?

Based on the previous plots, what are the optimal parameters of your simulation? Reason your choice.

The use of placeholders for the variables, allow us to set the relevant parameters to be played with on runtime. Below is an example bash script that will in turn check for the different parameters, namely N, dt and tau.

```
#!/bin/bash

NP=2
INPUT="in.script"

# --- Baseline values ---
BASE_dt=0.001
BASE_tau_T=0.1
BASE_N=2
BASE_T=300

for dt in 0.0001 0.0005 0.001 0.003 0.01; do
        mpirun -np $NP lmp_mpi -in "$INPUT" -v dt "$dt" -v tau_T "$BASE_tau_T" -v
            N "$BASE_N" -v temp "$BASE_T"
done

for tau_T in 0.01 0.1 1.0; do
  mpirun -np $NP lmp_mpi -in "$INPUT" -v dt "$BASE_dt" -v tau_T "$tau_T" -v N "
      $BASE_N" -v temp "$BASE_T"
done

for N in 2 3 4 5 6 8; do
  mpirun -np $NP lmp_mpi -in "$INPUT" -v dt "$BASE_dt" -v tau_T "$BASE_tau_T" -v N
       "$N" -v temp "$BASE_T"
done

echo "All single-variable scans completed."
```

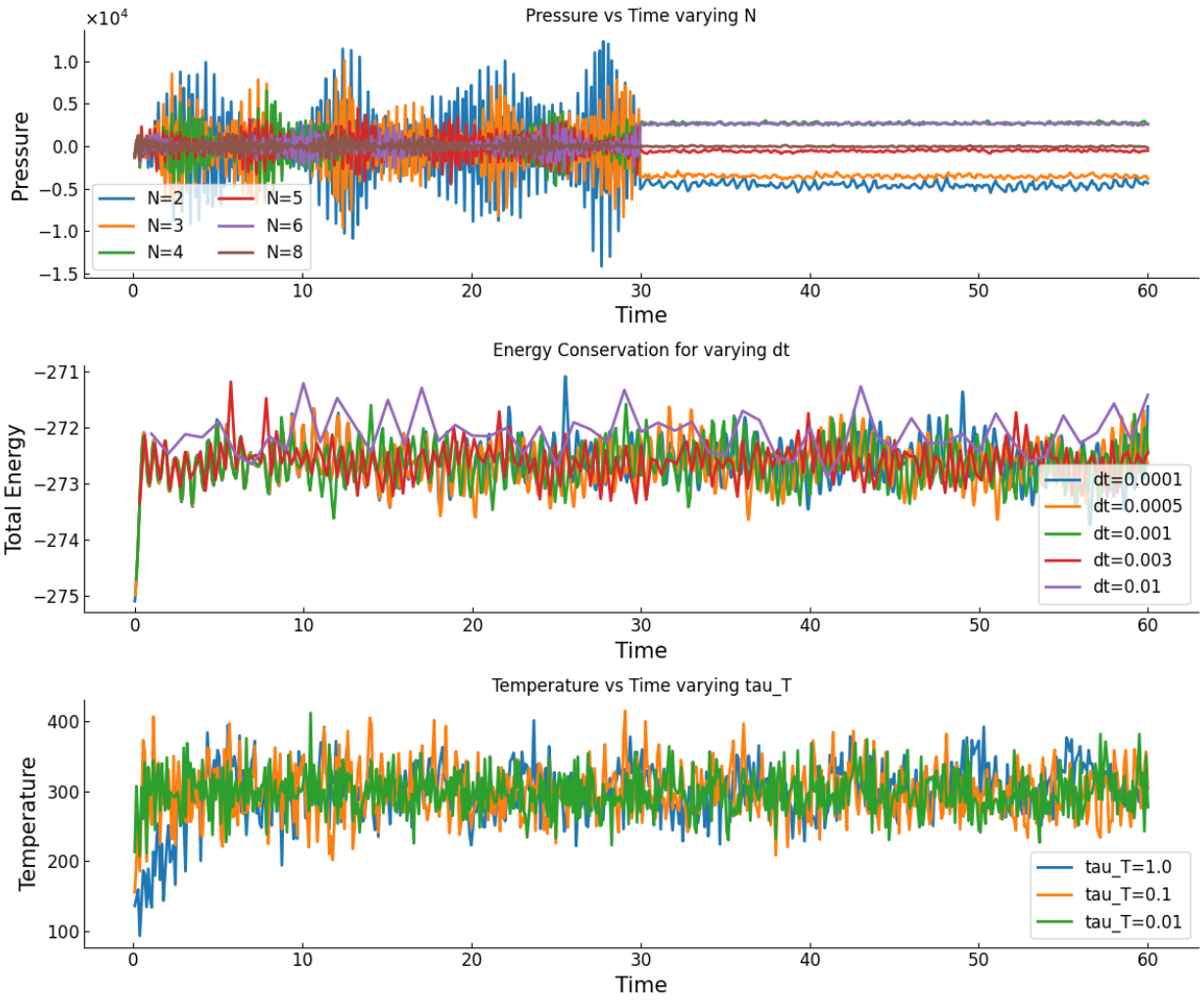

Figure 2: Results for different parameters.

In figure 2 we can see the results obtained for different parameters:

- The box size seems to condition the oscillations of the pressure function. That makes sense, since the bigger the box is the more robust to phonon wawes the simulation will be. Therefore the larger the better. Above N=6 it seems that the oscillations are atenuated.
- The timestep is crucial in a simulation. Here you can see that the smallest timesteps give the same trajectory in the energy coodinate, which means that at some point (maybe 1e-3) the timestep becomes redundant. The highest timestep produces to abrupt oscillations and therefore should be avoided. It

is recommended for silicon atoms to use timesteps between 1e-3 to 3e-3 ps.

- The relaxation constant of the thermostat determines the magnitude of the fluctuations. Too short constant means that the thermostat will oscilate widly, which is unphysical. Too long constant means that the system will take too long to equilibrate to the target temperature. In this case the system seems to be quite robust to the thermostat. The lammps manual recommends however using a relaxation constant of 100 timesteps = 0.1 ps for the temperature relaxation. For the pressure the recommendation is 1000 timesteps = 1.0 ps.

The parameters used for exercise 3 are actually a clue that these are the optimal parameters for the simulation.

3. For this part of the exercise use dt=1e-3 ps; τ_T = 0.1; τ_P = 1.0 and N = 6. Equilibrate your cell for a temperature range going from 200 K to 400 K, in steps of 10 K. Extract the length of your simulation cell (not the volume which scales as a cubic function) after equilibration. Plot the difference of the cell size and the size at 300 K against temperature (the curve should cross the 0 at 300 K). From your results extract the thermal expansion coefficient α at room temperature (300 K) by fitting the function $a(T) = a_0(1+\alpha(T-T_0))$
Again we use a bash script to loop over the temperatures going from 200 to 400 K in steps of 10.

```
#!/bin/bash

NP=4
INPUT="in.script"

for temp in {200..400..10}
do
  mpirun -np $NP lmp_mpi -in "$INPUT" -v dt 1e-3 -v tau_T 0.1 -v N 6 -v temp $temp
done
```

From that we fitted the length of the box against the temperature, according to the given function.

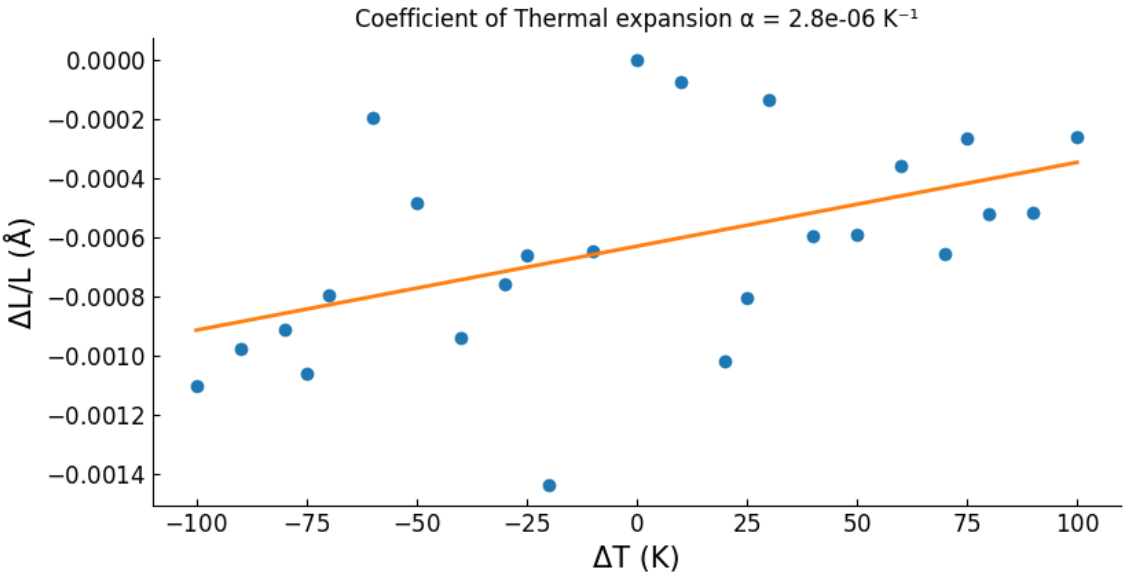

Figure 3: Relative expansion of the simulation cell as a function of the temperature difference

Although the fitting is not very accurate, the coefficient obtained is surprisingly close to the empirical value, 2.6e-6 /K.
Improving the accuracy of the obtained data points would require having a larger cell where the expansion can be observed accurately.