



NANOSCALE MATERIAL MODELING

WEEK 5

Ilija A. Gjerapić, S4437586; i.a.gjerapic@student.rug.nl; @github
Supervisor: prof. dr. A. Giuntoli, prof. dr. J. Slawinska

Contents

1 Assignment 1: What is a Particle?	2
1.1 Role of looping structure	2
1.2 Extraction of Potential of Mean Force	2
1.3 Course Graining	3
2 Assignment 2: Just Jamming	5
3 Assignment 3: Patch up	7
4 Assignment 4: Look Ma, no hands!	9

1 Assignment 1: What is a Particle?

1.1 Role of looping structure

The looping structure is responsible for moving the two fullerene molecules in discrete steps of 0.3 Å and 0.1 Å for large and small distances, respectively. After the two fullerene molecules are moved closer together, the system is run under nvt ensemble at 300K for 2.5 ps with the positions and forces on each atom being sampled every 0.25 ps. For each distance step, 20 iterations are conducted providing a total of 40 iterations. The LAMMPS code responsible for the larger steps is provided in Listing 1

```
1 label          loop
2 variable       loop_count equal 20
3 variable       a loop ${loop_count} # iterator ranging from 1 to 20, inclusive.
4
5 variable       large_shift equal 0.3 # moves molecules towards each other by 2*shift
6 displace_atoms c60a move ${large_shift} 0 0 units box
7 displace_atoms c60b move -${large_shift} 0 0 units box
8
9 dump          2 all custom 500 lmp_data/dump.data${a} id type xu yu zu fx fy fz
10 dump_modify   2 sort id
11 run 5000
12 undump 2
13 next          a # goes to next a value, but stops after a=20.
14 jump          SELF loop # jumps to the label "loop" at the start of this code block
```

Listing 1: The LAMMPS code for moving the two fullerene molecules closer together for larger distances in discrete steps of 0.3 Å over 20 iterations. After the initial movement of atom positions, the system is run under nvt ensemble at 300K for 2.5 ps with the positions and force on each atom being sampled every 0.25 ps.

1.2 Extraction of Potential of Mean Force

The force and potential energy between the two molecules was extracted using MDAnalysis. To achieve this, the COM on each molecule and the total force on each molecule was determined for every sampled point in the simulation. The total force of a single molecule for each frame was then projected on the vector connecting the COMs of the two molecules. These steps are outlined in Listing 2. Only the forces on a single molecule were calculated as the other molecule feels an equal and opposite reaction in accordance with Newton's Laws of motion.

```
1 # Loop over the frames and load the positions and forces
2 for i, traj in enumerate(u.trajectory):
3     c60a = u.atoms.fragments[0]
4     c60b = u.atoms.fragments[-1]
5
6     c60a_forces[i] = sum(c60a.forces.copy())
7     c60b_forces[i] = sum(c60b.forces.copy())
8     c60a_COM[i] = c60a.center_of_mass()
9     c60b_COM[i] = c60b.center_of_mass()
10
11 # vector pointing from c60a COM to c60b COM
12 vect_connect = c60a_COM - c60b_COM
13
14 # projection of forces on molecule a onto vector connecting of COMs for each distance
15 distances = np.linalg.norm(vect_connect, axis=1)
16 force_projection = np.einsum('ij,ij->i', c60a_forces, vect_connect) / distances
17
```

Listing 2: The python code used to extract the total force of a molecule projected onto the vector connecting the two COMs of the molecules. The molecules were identified using `u.atoms.fragments` to group atoms based on if they are reachable by transversing bonds.

The calculated force projections were then averaged over 50 distance bins using the code outlined in Listing 3.

```
1 # Setting up bins
2 min_dist = min(distances)
3 max_dist = max(distances)
4 num_bins = 50
5
6 bin_edges = np.linspace(min_dist, max_dist, num_bins)
7 bin_centers = 0.5 * (bin_edges[:-1] + bin_edges[1:])
8
```

```

9 # averaging forces on molecule A
10 avg_forces = np.zeros_like(bin_centers)
11 for i in range(len(bin_centers)):
12     mask = (distances >= bin_edges[i]) & (distances <= bin_edges[i + 1])
13     avg_forces[i] = np.average(force_projection[mask])
14
15 # removing nan values
16 valid_indices = ~np.isnan(avg_forces)
17 avg_forces_valid = avg_forces[valid_indices]
18 bin_centers_valid = bin_centers[valid_indices]

```

Listing 3: The python code used to average out the force projections

The potential energy was then extracted from these forces:

$$V = - \int_O^\infty \mathbf{F} \cdot d\mathbf{r} = \int_\infty^O \mathbf{F} \cdot d\mathbf{r} \quad (1.1)$$

using the `scipy.integrate.trapezoid` method as shown in Listing 4

```

1 # determining Potential Mean Field
2 potential = np.zeros(len(bin_centers_valid))
3 for i in range(len(bin_centers_valid) - 1, 0, -1):
4     potential[i-1] = potential[i] + trapezoid(avg_forces_valid[i-1:i+1],
5                                              x=bin_centers_valid[i-1:i+1])
5

```

Listing 4: The python code used to determine the PMF. Integration starts from larger distances to smaller to allow for complete integration of the forces. Actual integration is achieved using the `scipy.integrate.trapezoid` function.

1.3 Course Graining

The extracted mean force and potential of mean force is shown in Figure 1.1(a) and Figure 1.1(b), respectively. The Lennard-Jones potential

$$V_{\text{LJ}} = 4\epsilon \left[\left(\frac{\sigma}{r} \right)^{12} - \left(\frac{\sigma}{r} \right)^6 \right], \quad (1.2)$$

where ϵ is the well depth, and σ the zero-crossing distance along with the Morse potential

$$V_{\text{Morse}} = D_e \left(e^{-2a(r-r_0)} - 2e^{-a(r-r_0)} \right), \quad a = \sqrt{\frac{k}{2D_e}}, \quad (1.3)$$

were D_e is the well depth, k the force constant of the bond, and r_0 the equilibrium bond distance were fitted to the PMF as shown in Figure 1.1(b). The non-bonded interactions are best described with a Morse potential.

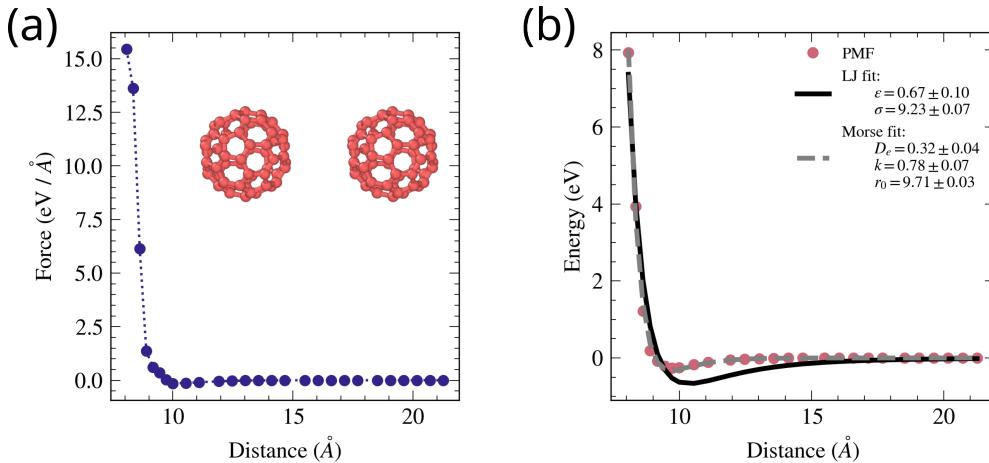


Figure 1.1: The averaged force (a) and the potential of the mean force (b) calculated for two fullerene molecules. The Lennard-Jones (black solid line) and the Morse (grey dashed line) potentials were fitted to the PMF using `scipy.optimize.curve_fit`. The Morse potential most accurately describes the PMF. The inset of (a) is an ovito snapshot of the two fullerene molecules.

A simulation utilizing the coarse-grained particles would require the mass, the fitting parameters, and the diameter of the particles. For the case of fullerene, the mass is 750 g/mol, the diameter 7.1 Å(determined from the average R_g from the previous simulations), and the fitting parameters are shown in Figure 1.1. An implementation of these course-grained particles is shown in Listing 5. The difference between a CG particle and an all-atom fullerene is showed in Figure 1.2(a), while the result of Listing 5 is showed in Figure 1.2(b).

```

1 dimension      3
2 units         metal
3 atom_style    full
4 boundary      P P P
5
6 # Variables
7 variable       N equal 200      # number of particles
8 variable       eta equal 0.3    # packing fraction
9 variable       temp equal 300.0   # temperature
10
11 variable      sigma equal 7.1   # diameter from avg R_g of one molecule
12 variable      particle_volume equal (4.0/3.0)*PI*(0.5*${sigma})^3
13 variable      total_volume equal ${N}*${particle_volume}
14 variable      lbox equal (${total_volume}/${eta})^(1.0/3.0)
15 variable      l equal 0.5*${lbox}
16 variable      rand_seed equal 5468945
17
18 region        box block -${l} ${l} -${l} ${l} -${l} ${l}
19 create_box    1 box
20 create_atoms  1 random ${N} ${rand_seed} box
21
22 mass          1 750
23 pair_style    morse      17.5
24 pair_coeff    1 1 0.32  1.104  9.71  # D_e,  a,  r_0
25
26 minimize     0.0 1.0e-8 1000 100000
27
28 write_data   relaxed_CG.data

```

Listing 5: An example implementation of the start of a simulation of our coarse-grained fullerene molecules. The result is seen in Figure 1.2(b).

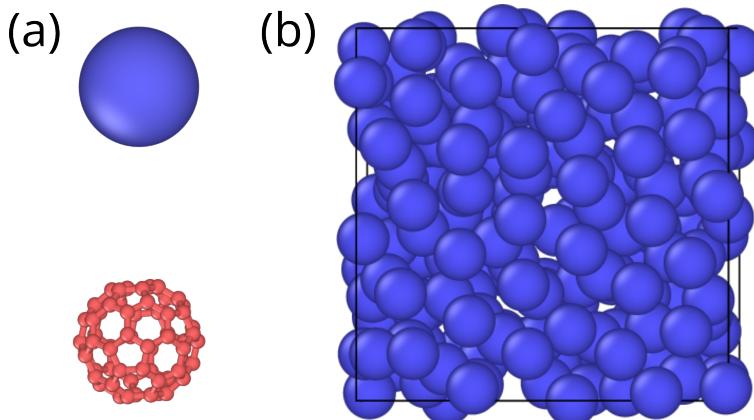


Figure 1.2: (a) The comparison between an all-atom fullerene molecule and its coarse-grained bead, from `fullerene_CG.data`. (b) The result of the 200 coarse-grained fullerene beads in a 30 v% box.

2 Assignment 2: Just Jamming

The input script was adjusted such that the packing fraction η and the interaction strength ϵ could be inputted as variables using `lmp -in in.3dlj -v ETA $eta -v EPSILON $epsilon`.

```

1 # Variables
2 variable      N equal 2048      # number of particles
3 variable      eta equal ${ETA}  # packing fraction
4 variable      temp equal 1.0    # temperature
5 variable      epsilon equal ${EPSILON}
6
7 # ...
8
9 pair_style      lj/cut 2.5
10 pair_coeff     1 1 ${epsilon} 1.0 2.5

```

For a fixed packing fraction of $\eta = 0.4$, the interaction strength was varied to be $\epsilon \in \{0.5, 0.2, 5.0\}$. All simulations reached equilibrium as shown in Figure 2.1.

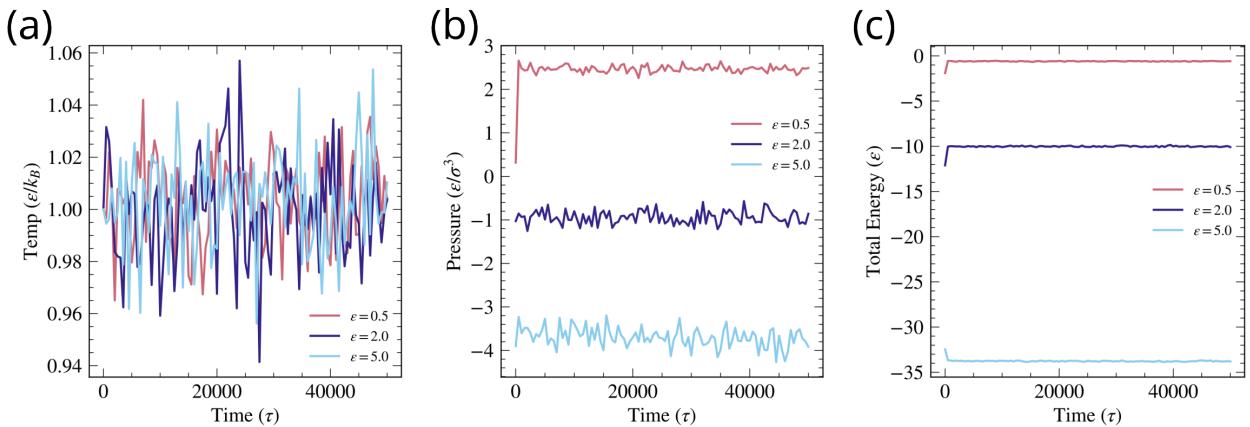


Figure 2.1: The (a) temperature, (b) pressure and (c) total energy against the simulation time for a LJ substance with packing fraction of $\eta = 0.4$ and interaction strengths $\epsilon = 0.5$ (red), 2.0 (dark blue), 5.0 (light blue). The plateaus in all thermodynamic properties demonstrate that each simulation reached equilibrium.

Figure 2.2 shows the radial distribution function and the mean square displacement (MSD) for an LJ substance with a packing fraction of 0.4. For interaction strengths of 0.5 and 2.0, the LJ substance is a liquid, while the LJ substance is crystalline with $\epsilon = 5.0$. The liquid behavior is supported by the less intense and broad peaks that quickly decay in the RDF. For the crystalline phase with $\epsilon = 0.5$, the peaks in the RDF are much more intense and sharper. Moreover, the MSDs for $\epsilon = 0.5, 2.0$ both show a ballistic region followed by diffusion. The crystalline phase of $\epsilon = 0.5$ is supported by the plateau in the MSD after $10^2 \tau$.

The crystalline phase for $\eta = 0.4, \epsilon = 5.0$ and liquid phases for $\eta = 0.4, \epsilon = 0.2, 0.5$ are qualitatively confirmed by using the common neighbor analysis modification in ovito as shown in Figure 2.3. This shows that the crystalline structure is primarily FCC, with some HPC contributions.

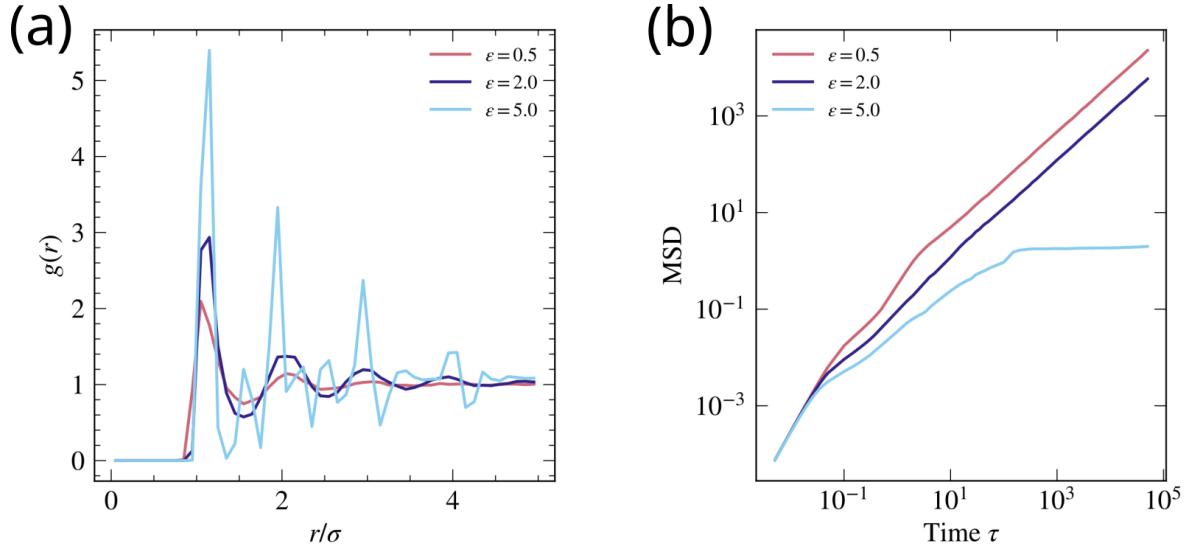


Figure 2.2: The (a) RDF and (b) MSD for a LJ substance with packing fraction of $\eta = 0.4$ and interaction strengths $\epsilon = 0.5$ (red), 2.0 (dark blue), 5.0 (light blue). For interaction strengths of 0.5, 0.2 the LJ substance is a liquid, but is crystalline for an interaction strength of 5.0.

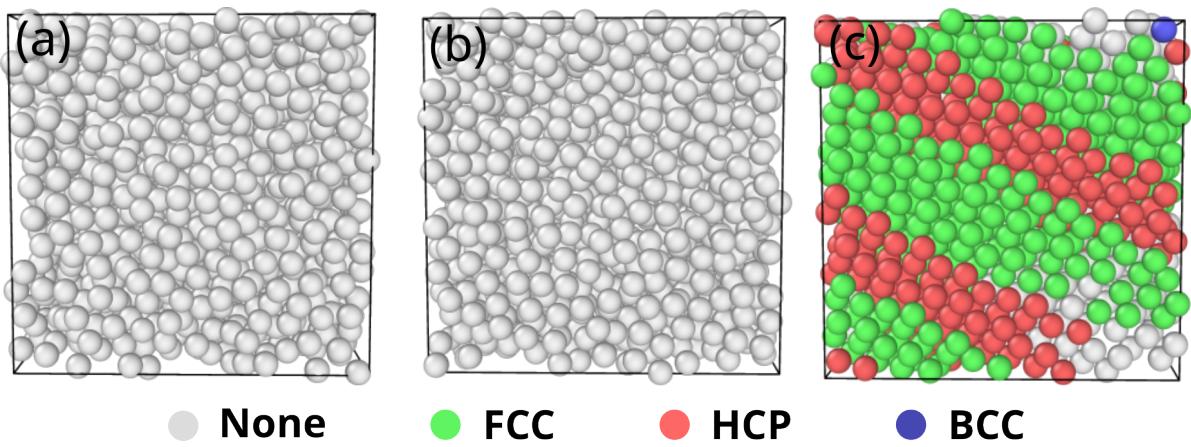


Figure 2.3: The common neighbor analysis for the final configuration for an LJ substance at temperature 1.0 with a packing fraction 0.4 and interaction strength (a) $\epsilon = 0.5$, (b) $\epsilon = 2.0$ (c) $\epsilon = 5.0$. The LJ substance forms a crystalline structure of mixed HPC and FCC packing for an interaction strength of 5.0, while for interaction strengths of 0.5 and 2.0, no crystalline structures are observed.

3 Assignment 3: Patch up

Two initial data files with patchy particles were built with one consisting of 25% particles having 2 patches (N_2) and 75% having 4 patches (N_4) and the other having equal number of N_2 and N_4 particles. The naming convention for each system is $N_2.Y$, where Y is the fraction of particles that are N_2 . Figure 3.1 shows the visualization of the two different patchy particles considered.

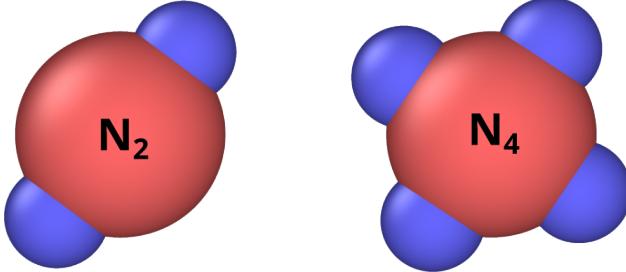


Figure 3.1: Ovito images of the patchy particles considered.

Each system has a volume fraction $\phi = 0.2$ following [1]. Moreover, for each system, the simulation was run at a temperature of $T = 0.5$ and $T = 2.0$. The lower bound was chosen as Zhang *et al.*, showed that N_4 -like particles form sheets below $k_B T / \epsilon = 1.0$, while N_2 -like particles form long chains at $k_B T / \epsilon = 0.5$. On the other hand, Zhang *et al.* showed N_4 particles form a disordered state above $k_B T / \epsilon = 1.67$ and N_2 above $k_B T / \epsilon = 1.67$ [1]. The evidence that each system reached equilibrium before analyzing the final frame is shown in Figure 3.2 and Figure 3.3 for N_2 -0.25 and N_2 -0.5, respectively.

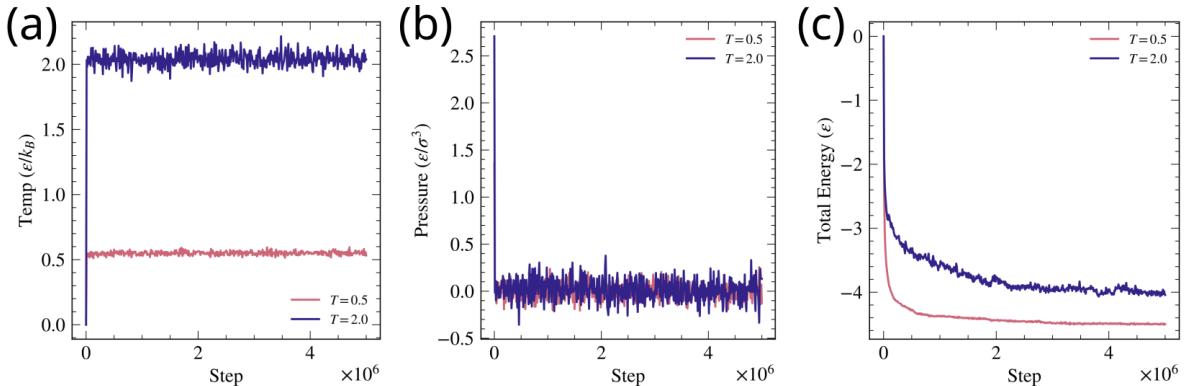


Figure 3.2: The thermodynamic properties (a) temperature, (b) pressure and (c) Total energy of the N_2 -0.25 system overtime at $k_B T / \epsilon = 0.5$ (red) and $k_B T / \epsilon = 2.0$ (blue).

The results of the simulations are shown in Figure 3.4(a-d). In all systems clusters of atoms are formed, but of different sizes and general shapes.

To quantitatively characterize the size of the clusters formed, the number of constitute atoms is considered. To avoid clusters consisting of only two patchy particles, a cluster is considered to be formed when more than eight atoms can be traversed with a cutoff distance of 1.5. This results in the smallest clusters consisting of three patchy particles as N_2 and N_4 consist of three and five atoms, respectively. The final ensemble averaged cluster size for each system is show in Figure 3.4(e).

The shape of the cluster is quantified by the "acircularity" c ,

$$c = \lambda_1^2 - \lambda_2^2, \quad \lambda_1 > \lambda_2, \quad (3.1)$$

where $\lambda_{1,2}$ are the eigenvalues of the 2D gyration tensor. The acircularity is my 2D analog to the acylindricity. The acircularity of a cluster is exactly zero if the particles are homogeneously distributed in the x-y plane (e.g square or circular sheets), and gives higher values for more linear clusters. A comparison of the final ensemble averaged acircularity for each system is shown in Figure 3.4(f)

For both system compositions, larger chain-like clusters are formed at $k_B T / \epsilon = 0.5$, while smaller sheet clusters are formed at $k_B T / \epsilon = 2.0$.

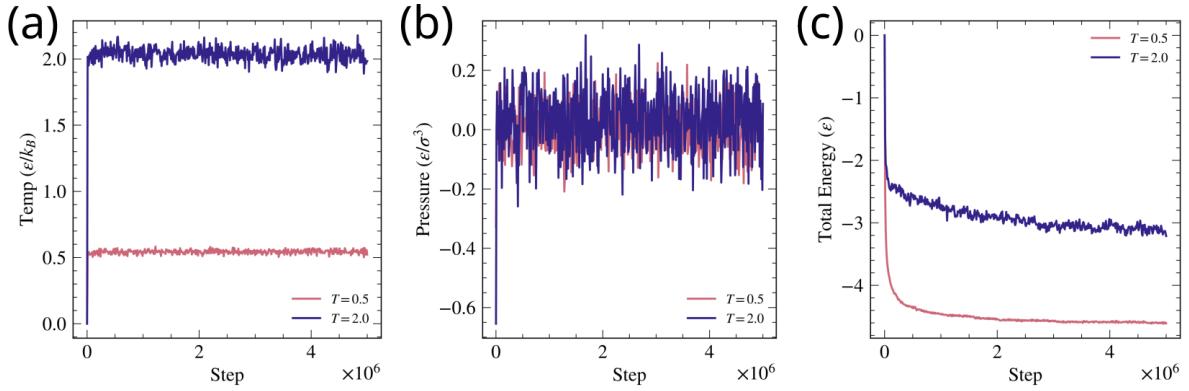


Figure 3.3: The thermodynamic properties (a) temperature, (b) pressure and (c) Total energy of the $N_2\text{-}0.5$ system overtime at $k_BT/\epsilon = 0.5$ (red) and $k_BT/\epsilon = 2.0$ (blue).

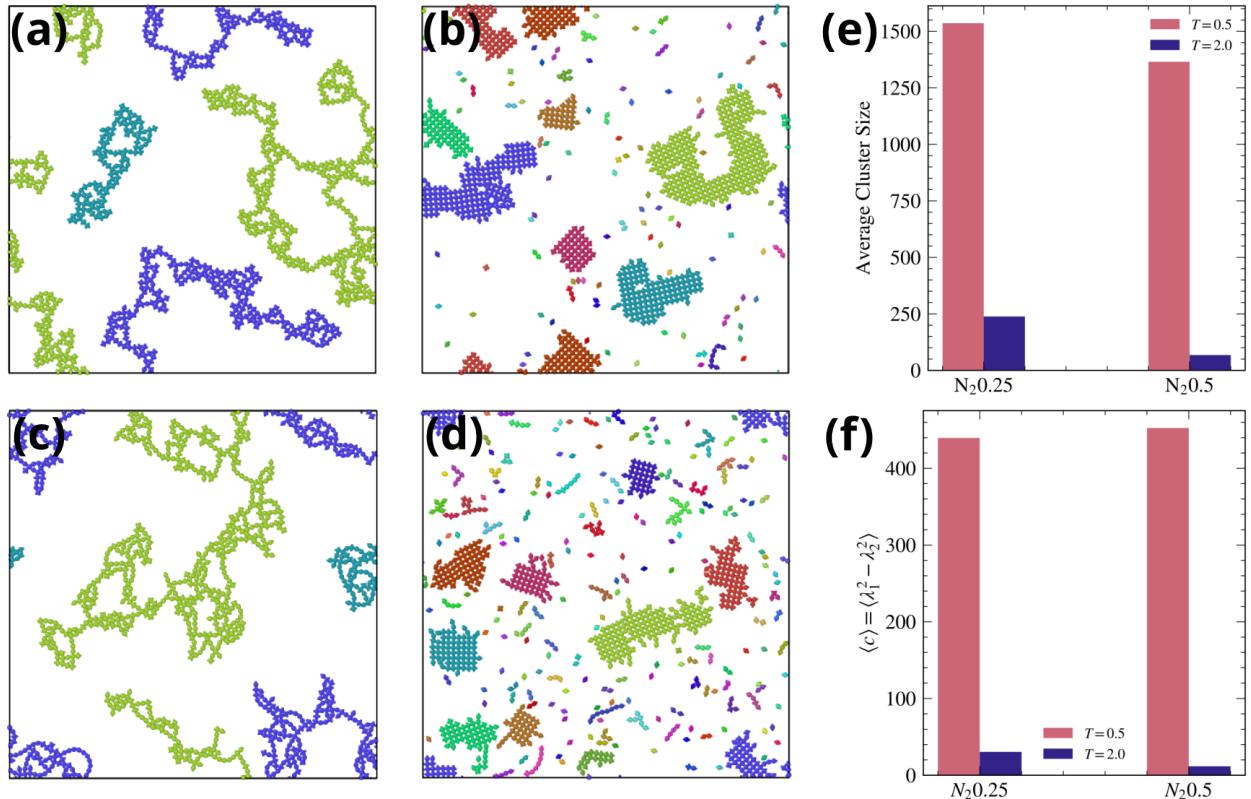


Figure 3.4: The final configurations of (a) $N_2\text{-}0.25$ $k_BT/\epsilon = 0.5$, (b) $N_2\text{-}0.25$ $k_BT/\epsilon = 2.0$, (c) $N_2\text{-}0.5$ $k_BT/\epsilon = 0.5$ and (d) $N_2\text{-}0.5$ $k_BT/\epsilon = 2.0$. The atoms are colored by their cluster, defined by a cutoff distance of 1.5. (e) The cluster size characterized by the number of constituent atoms. A cluster is considered to be formed when a cluster consists of more than eight atoms. (f) The shape of the clusters is characterized by the average "acircularity", $c = \lambda_1^2 - \lambda_2^2$, $\lambda_1 > \lambda_2$. For both compositions, larger chain-like clusters are formed at the lower temperature, while smaller sheets are formed at the higher temperature.

4 Assignment 4: Look Ma, no hands!

The temperature is stabilized using the `fix nvt/sllod` command. This command performs a constant NVT integration to update the atom positions and velocities for a simulation box that is changing size and/or shape. The NVT ensemble is implemented using the Nosé-Hoover temperature thermostat, which globally rescales the kinetic-energy of the particles such that the Nosé-Hoover non-Hamiltonian is conserved [2]. In our case, the global kinetic-energy is rescaled every 100 time steps.

To determine the flow-curve of colloids, the equilibrated structure of an LJ substance at temperature 1.0, packing fraction 0.58, $\epsilon = 1.0$ was subjected to shear rates of $\dot{\gamma} \in \{0.001, 0.005, 0.01, 0.05, 0.1, 0.2, 0.4, 0.5, 0.8, 1.0\}$. Figure 4.1 shows the initial configuration was in equilibrium before the shear was applied.

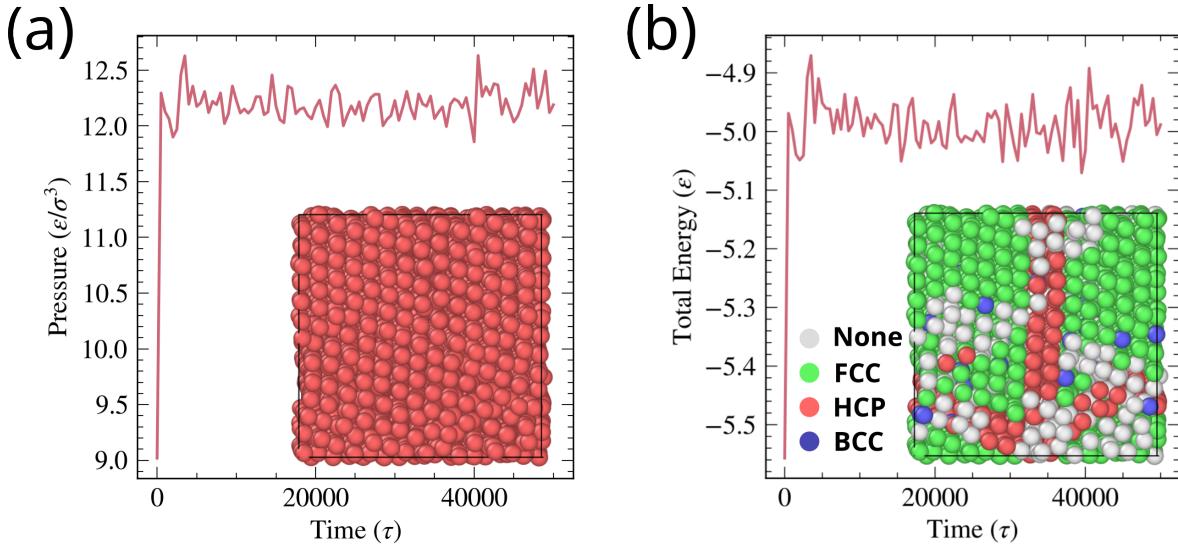


Figure 4.1: The time evolution of the (a) pressure and (b) total energy for the LJ substance with a packing fraction of 0.58 and interaction energy 1.0 at temperature 1.0. The inset in (a) shows the final configuration of the substance and the inset in (b) shows the same configuration but colored according to the common neighbor analysis by `ovito`. The final configuration was used before applying different shear rates.

The viscosity was determined from the stress-strain curves. To determine the stress of the system due to the strain, the stress per atom was first calculated. The total stress was then computed as the sum of the xy component of the stress/atom over all atoms. The computed stress is given in units of pressure*volume, thus it is multiplied by the total particle volume. The implementation using LAMMPS is shown in Listing 6.

```

1 compute stressA all stress/atom NULL    # given in pressure*volume units
2 compute s4 all reduce sum c_stressA[4] # 4th element is xy component

```

Listing 6: The LAMMPS lines used to calculate the stress of the system.

The resulting stress-strain curves for the varius strain rates are shown in Figure ??(a). Further observation shows that for every strain rate, a steady-flow is obtained after a strain of 1%. Thus, the mean steady-state stress $\langle \sigma \rangle$ is given by the average of the stresses for strains of 1% and higher. The corresponding viscosity is given by

$$\eta = \langle \sigma \rangle / \dot{\gamma}, \quad (4.1)$$

where $\dot{\gamma}$ is the shear rate. The resulting flow curve is shown in Figure 4.2. A power law fit results in an exponent of $-0.96pm0.01$. A lower bound for the zero-shear viscosity is conservatively estimated to be $0.7\epsilon\tau/\sigma^3$ by taking the average of the viscosities corresponding to the two lowest shear rates.

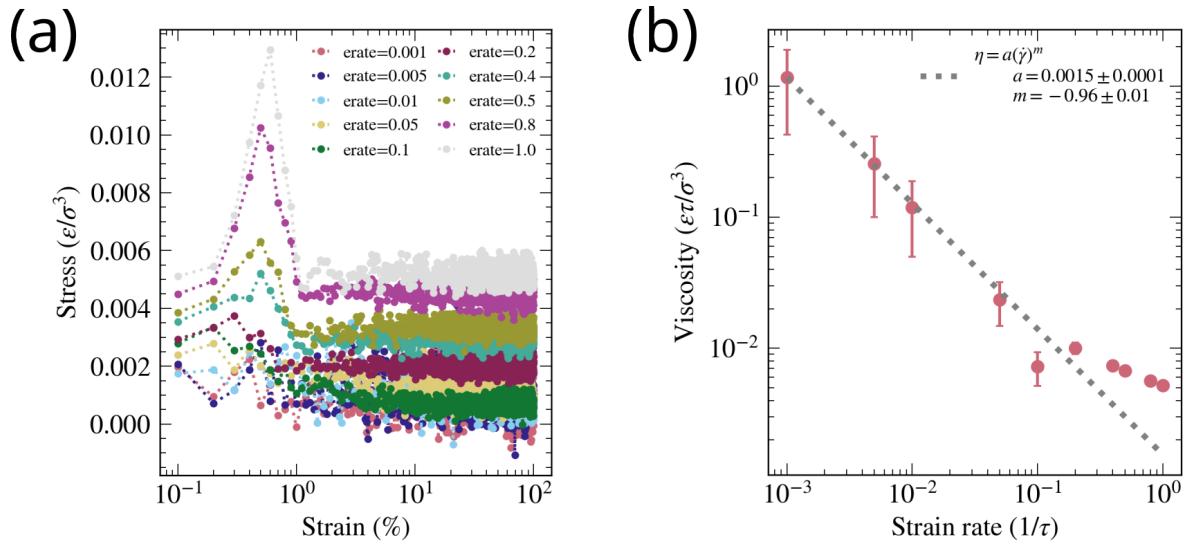


Figure 4.2: (a) The stress-strain curves for all strain-rates $\dot{\gamma}$ considered. For all strain-rates, a steady-flow is obtained after a strain of 1%. (b) The flow curve obtained by taking the viscosity η as the mean steady-state stress divided by the corresponding shear rate. The mean steady-state stress was determined as the average of the stresses for strains larger than 1%. A power law (grey dashed line) was fit with relative success. A lower bound of the zero-shear viscosity is determined to be $0.7\epsilon\tau/\sigma^3$ from the average of the two lowest shear-rates.

References

- (1) Zhang, Z.; Glotzer, S. C. Self-Assembly of Patchy Particles. *Nano Lett.* **2004**, *4*, 1407–1413, DOI: 10.1021/nl0493500.
- (2) Frenkel, D.; Smit, B., *Understanding Molecular Simulation : From Algorithms to Applications*, Third edition; Elsevier, Academic Press, an Imprint of Elsevier: London, United Kingdom, 2023.