

# CHE384T Pset 4: Molecular dynamics

## Solutions

October 10, 2024

Included is the code that was used in this problem set. Due to the fact that several modifications were made to each of the files, the relevant files of code are listed specifically to each problem where appropriate. All runs and plots of graphs are located in the corresponding PS4-\* .py.

## 1 Exercise 1: One Atom on the Surface

To simulate one atom on the surface, we use the Verlet algorithm and a surface potential of the form

$$\phi(x, y) = A \sin(\pi x) \sin(\pi y) \quad (1)$$

Since we deal with one atom, we input the energy. No periodic boundary conditions are used. For this exercise, we use the given and constructed files

1. `initsurf.py` → initialization of positions and velocities.
2. `phisurf.py` → contains surface potential.
3. `fsurf.py` → contains corresponding force of `phisurf.py`.
4. `MDSurf.py` → contains implementation of Verlet Algorithm for one atom on 2D surface.
5. `averaging.py` → gives thermodynamic averages of energies, temperature, and pressure.

### 1.1 Time step for numerical error

We first find the appropriate time step that preserves conservation of energy. In Figure ??, we show two different time step intervals and the corresponding energy. We use  $A = E_{in} = 1$ .

To determine whether our chosen parameters result in the conservation in energy, we examine the fluctuations of total energy. Using the criteria that  $\frac{E_{max} - E_{min}}{\langle E \rangle} 10^{-4}$ , we choose  $dt = 0.006$  in reduced units.

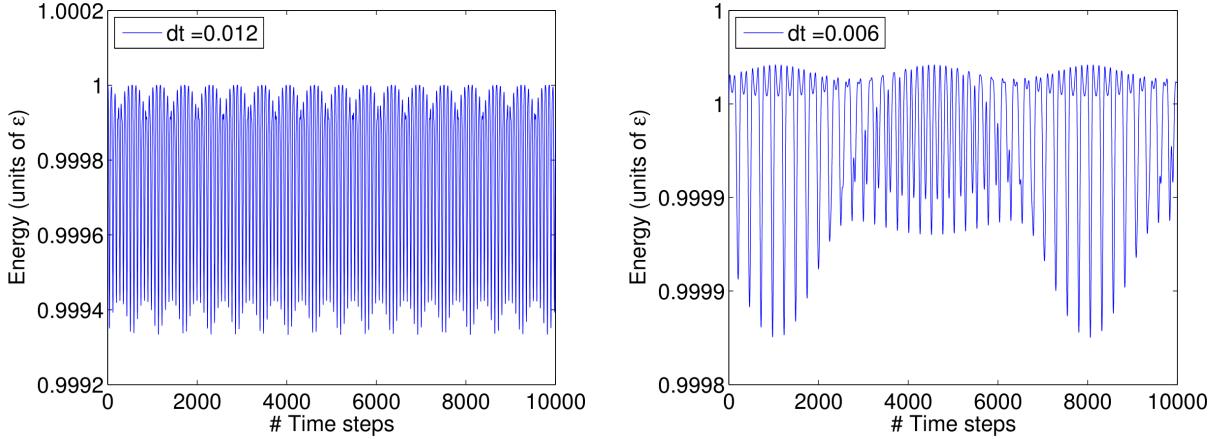


Figure 1: Convergence of fluctuations in total energy for varying time steps in reduced units, (a)  $dt = 0.012$  and (b)  $dt = 0.006$

## 1.2 Varying E

We next vary the inputed energy to examine the behavior between the total, potential, and kinetic energy. As seen in Figure ??, the total energy remains constant (fluctuations are small compared to the axes plotted). In contrast, the kinetic and potential energy fluctuate noticeably. Increases in kinetic energy correspond to decreases in potential energy, and vice versa, reflecting the harmonic oscillator like behavior of the particle on the surface. All runs were done with 5000 time steps.

## 1.3 $\langle u \rangle$ and $\langle K \rangle$

We additionally report the average potential and kinetic energies for the one atom system on a surface potential. The below numbers are for 20,000 steps,  $dt = 0.006$ ,  $A = 4$ , and  $E_{in} = 1$ . The averages are done not including the first 200 time steps to exclude any time for equilibration.

```
au = -1.0779
ak = 2.0778
ae = 0.9999 % for comparison to
```

The kinetic energy is directly proportional to the temperature, by the equipartition theorem. For a two dimensional system consisting of one atom,  $K = k_b T$ , where  $k_b$  is Boltzmann's constant.

## 1.4 Atom motion and energy, and corresponding trajectory

As the parameters for the potential surface remain constant, the fraction of potential energy of the total energy diminishes with increasing  $E_{in}$ . The strength of interaction with the

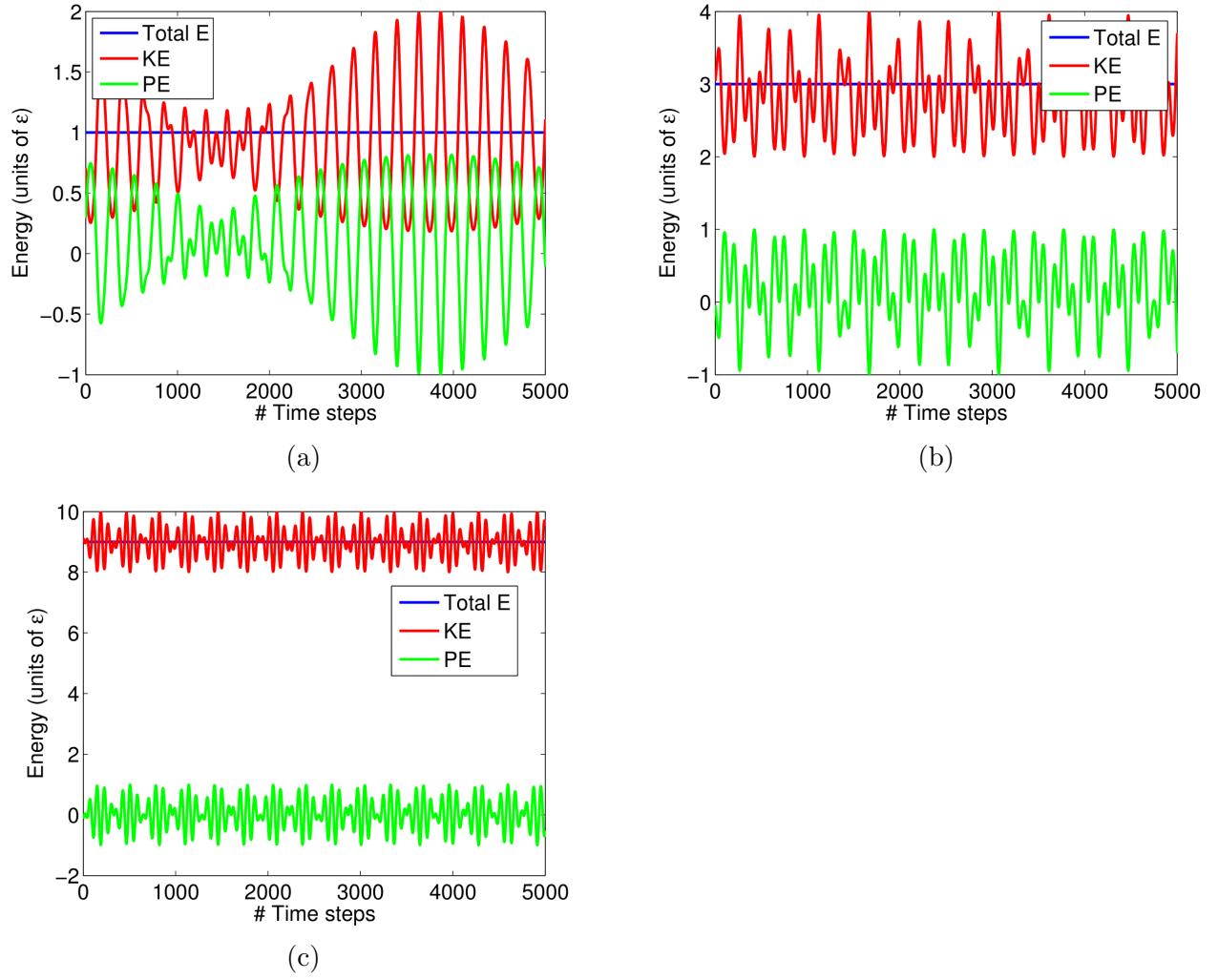
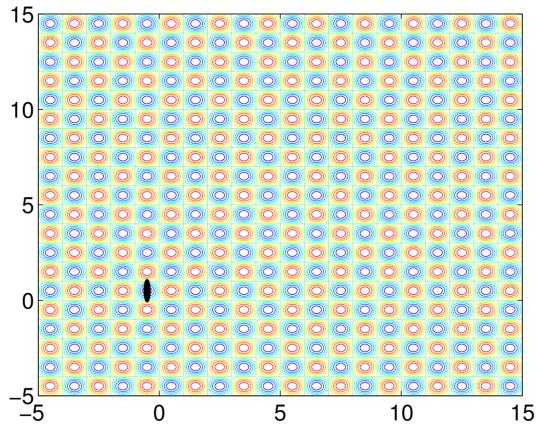


Figure 2: Tracking the relation between total, potential, and kinetic energy of one atom on a surface potential for (a)  $E_{in} = 1$ , (b)  $E_{in} = 3$ , and (c)  $E_{in} = 10$  with 5,000 time steps; units of energy are in reduced units.

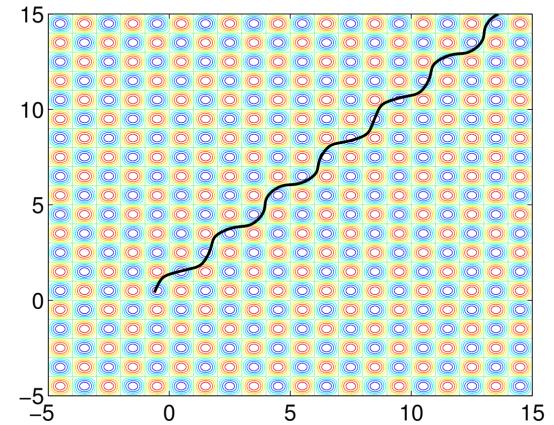
surface potential is directly proportional to  $A$ . For convenience, we choose  $A = 1$ . The atom is initialized to be in a well near the origin.

For sufficiently high  $E_{in} > A$ , we would expect that on average the particle has enough kinetic energy to move nearly uninhibited by the surface potential. In contrast, for  $E_{in} < A$ , the particle has less available kinetic energy, and thus would have a much shorter net trajectory. This is indeed what is observed in Figure ???. For comparison,  $E_{in} \approx A$  is also plotted, and exhibits a trajectory intermediate to the two extreme aforementioned cases.

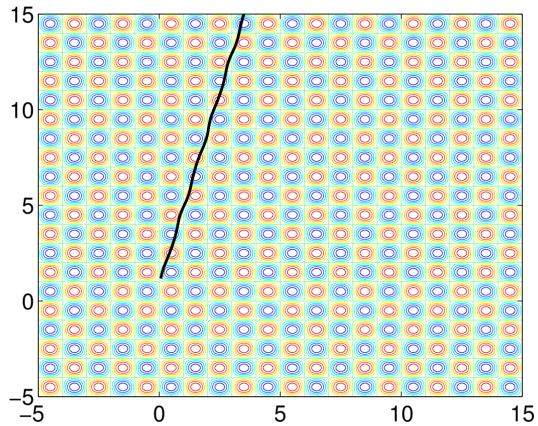
For  $E_{in} = 0.1 < A$ , the particle essentially stays in the well it was initialized in and is essentially stationary, as it does not have sufficient kinetic energy to escape the well. In contrast, the atom with  $E_{in} = 5 > A$  is able to pass through the potential with the initial velocity with little problem, as if potential were a small perturbation. The intermediate case shows a combination of both behaviors, in which the trajectory of the the particle is clearly altered by the potential, but not sufficiently so that it halts its motion across the surface.



(a)



(b)



(c)

Figure 3: Trajectory of one atom on contour of surface potential for (a)  $E_{in} = 0.1$ , (b)  $E_{in} = 1$ , and (c)  $E_{in} = 5$  for  $A = 1$  with 10,000 time steps; units of energy are in reduced units.

## 2 Exercise 2: Multiple Atoms on the Surface

Now we add additional atoms and interactions between them using the Lennard-Jones potential. For this exercise, the following files were used

1. `initsurf2.py` → initialization of positions of  $n$  atoms; atoms are randomly placed in a well and then slightly displaced randomly in that well; initialization occurs such that no more than one atom sits in a well.
2. `MBinit.py` → initialization of velocities for  $n$  atoms in Maxwell-Boltzmann distribution centered around zero based off of desired temperature input.
3. `phisurf.py` → contains surface potential.
4. `fsurf.py` → contains corresponding force of `phisurf.py`.
5. `fLJ2.py` → contains calculation of forces for 2D Lennard-Jones system; also outputs corresponding potential energy.
6. `MDSurfLJ.py` → contains implementation of Verlet Algorithm for one atom on 2D surface and interaction using Lennard-Jones potential.
7. `averaging.py` → gives thermodynamic averages of energies, temperature, and pressure.

Figure ?? gives an example initial configuration for 10 atoms, as implemented by `initsurf2.py`.

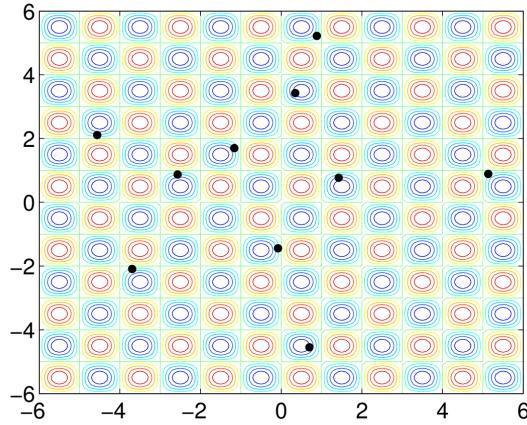


Figure 4: Example initial position configuration for modeling multiple atoms on 2D surface potential with Lennard-Jones interactions

Since we invoke the minimum image convention, we must rescale the positions of atoms that have trajectories outside the  $6 \times 6$  simulation cell with lattice constant of unity we have chosen. This is implemented in `PS4-* .py`, which determines how much the trajectory of each atom folds upon itself in the simulation cell. That is, if the trajectory of the atom leaves the top of the cell, it is observed to continue from the bottom of the cell.

In Figure ?? we plot the trajectories for a system of 4 atoms over 3,000 time steps for  $dt$

$= 0.006$ . A similar behavior is observed as with the one atom system. Instead of inputting energy, we input temperature, which determines the statistical distribution of velocities of the atoms in the system. When  $T_{in}$  is comparable to  $A$ , there is observed clustering of the atoms, as seen by the fact that few trajectories in the upper half of the simulation cell, but also some interaction and spread in motion. Likewise, when the kinetic energy, proportional to  $T_{in}$  is much smaller than  $A$ , there is insufficient kinetic energy, and so the atoms remain in their respective wells; the surface potential dominates the interactions. Finally, when  $T_{in}$  is larger than  $A$ , there is more available kinetic energy. Limited clustering is observed, but the particles act more like non-interacting particles with high kinetic energies, as seen by the large spread of trajectories in the plot.

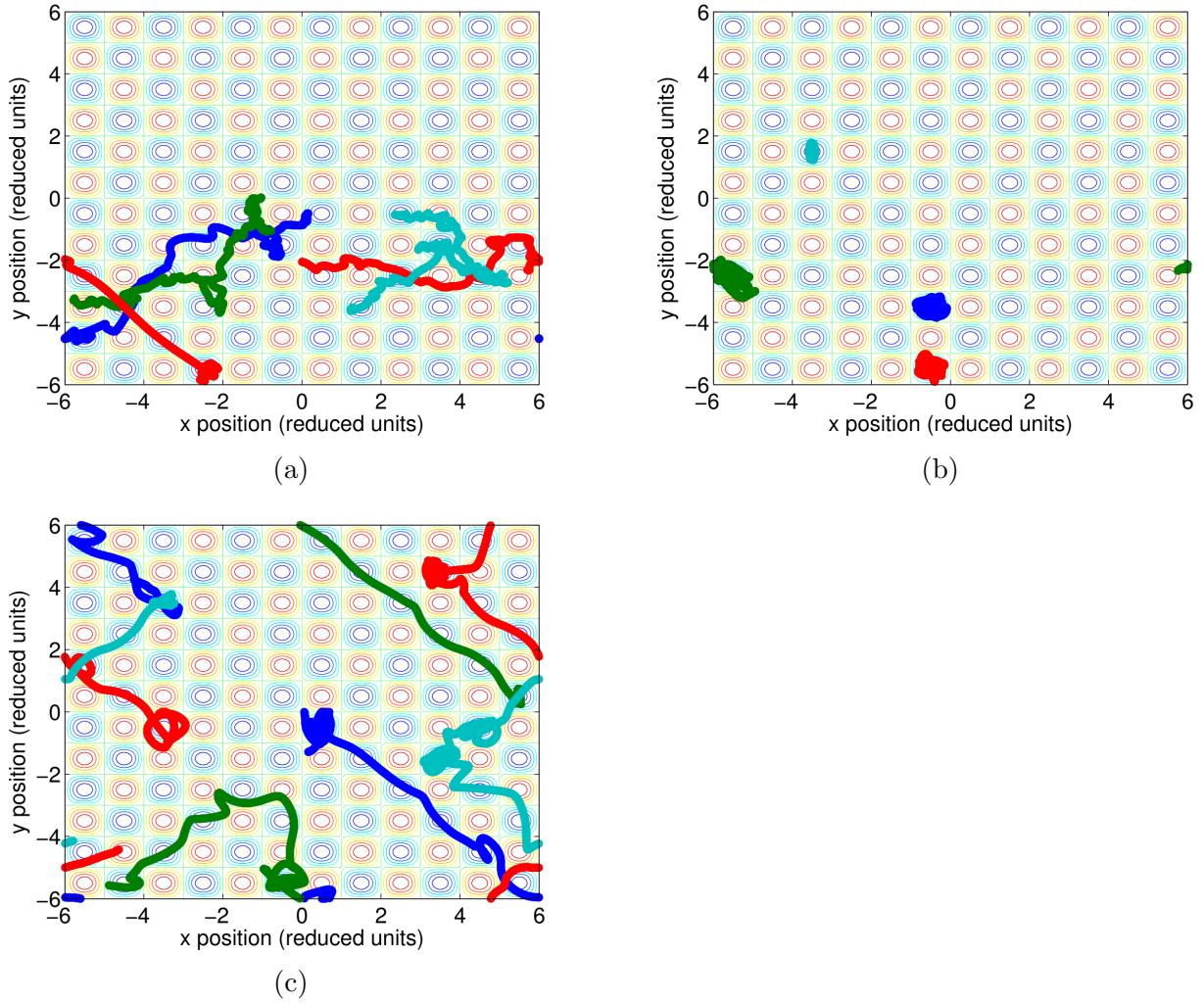


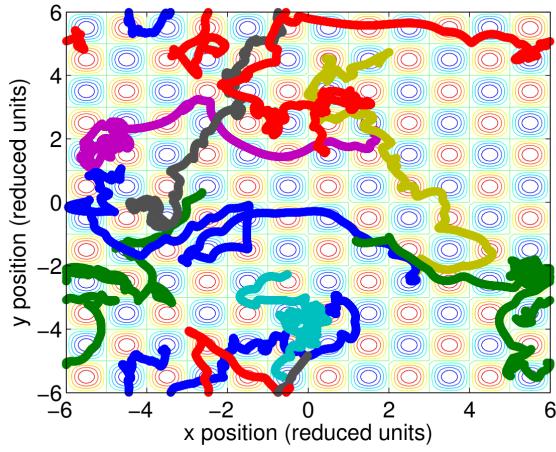
Figure 5: Trajectory for four atoms on contour of surface potential for (a)  $T_{in} = A = 0.1$ , (b)  $T_{in} = 0.1, A = 1$ , and (c)  $T_{in} = 1.2, A = 1$  for 3,000 time steps; all quantities are in reduced units.

### 3 Exercise 3: Even More Atoms on the surface

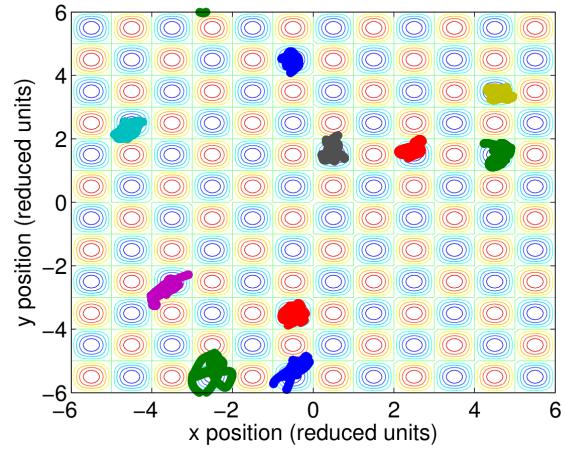
We repeat our above analysis for a different density of atoms on the surface. Here we choose to model a system of 10 atoms on our  $6 \times 6$  simulation cell with lattice constant of unity.

When  $T_{in}$  is comparable to A, there is observed but limited clustering of the atoms. That is, there is a fair spread of trajectories and interaction between atoms, as seen by the squiggly trajectories. Indeed, unlike the four atom case, there is much more wiggle to the trajectories indicating the increased influence of atomic interactions from the Lennard-Jones potential. For the case when the kinetic energy, proportional to  $T_{in}$  is much smaller than A, there is insufficient kinetic energy, and so the atoms remain in their respective wells; the surface potential dominates the interactions, as was found in the four atom case. There is more variation to the motion of the atoms in their respective wells, as there is more variation in the initial velocities possible. Finally, when  $T_{in}$  is larger than A, there is more available kinetic energy. Limited clustering is observed, but the particles act more like non-interacting particles with high kinetic energies, as seen by the large spread of trajectories in the plot and that some atoms are able to traverse quite far in the simulation cell without being deflected significantly.

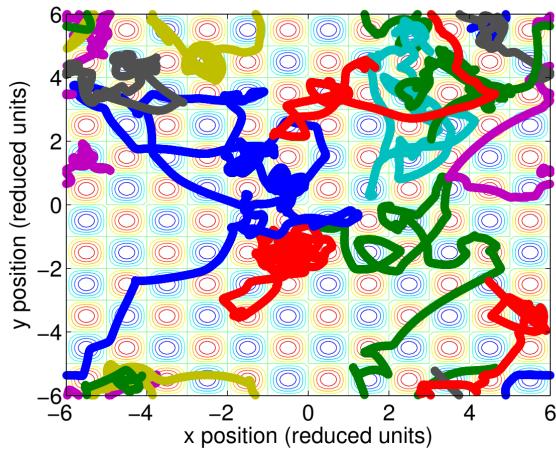
It is worth noting that it is possible for such a small density of atoms, that extremely large velocities may be initialized, especially as the choice of initial total momentum of the system is zero.



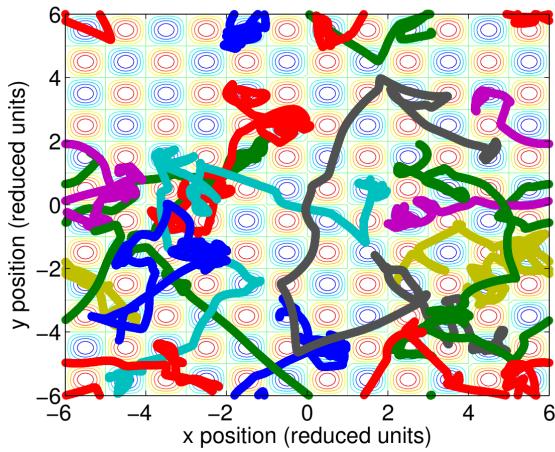
(a)



(b)



(c)



(d)

Figure 6: Trajectory for 10 atoms on contour of surface potential for (a)  $T_{in} = A = 0.1$ , (b)  $T_{in} = 0.1, A = 1$ , (c)  $T_{in} = 1.2, A = 1$  and (d)  $T_{in} = 1.5, A = 1$  for 3,000 time steps; all quantities are in reduced units.

## 4 Exercise 4: Bulk FCC- Proper Time Step

Now we turn to the bulk material. We choose to simulate the parameters of the Ne atom, which has Lennard-Jones parameters  $\sigma = 2.74\text{\AA}$ ,  $\epsilon = 0.0031 \text{ eV}$  and molar mass of 20.1797 g/mol. To get the mass per atom, we divide the molar mass by Avogadro's constant.

a) assuming a time step of  $5 \cdot 10^{-15} \text{ s}$ , our predicted time step in reduced units is given by

$$dt_{red} = \frac{5 \cdot 10^{-15}}{t_o} \quad (2)$$

where  $t_o$  is

$$t_o = \sqrt{\frac{m_{atom}\sigma^2}{\epsilon}}. \quad (3)$$

From these parameters, we calculate  $dt_{calc} = 0.0022$ .

b) to calculate the theoretical time step, we use the fact that the period of an atom is  $\frac{2\pi}{\omega}$ , where  $\omega = \sqrt{\frac{k}{m}}$ . It can be shown that the spring constant,  $k$ , is  $k = 377\frac{\epsilon}{\sigma^2}$ . Assuming 50 time steps per period, we find the theoretical reduced time constant,

$$dt_{th} = \frac{2\pi}{\sqrt{\frac{k}{m}}} \frac{1}{t_o} = 0.0065, \quad (4)$$

which is around three times larger than the one from Lennard-Jones empirical parameters.

## 5 Exercise 5: Bulk FCC- Confirming proper time step

For this exercise, we use the following code:

1. `initLJMD.py` → initialization of positions of  $n$  atoms; atoms are chosen to sit on *fcc* lattice sites.
2. `init3dMB.py` → initialization of velocities for  $n$  atoms in Maxwell-Boltzmann distribution centered around zero based off of desired temperature input for 3D system.
3. `fLJsum.py` → contains calculation of forces for 3D Lennard-Jones system; also outputs corresponding potential energy and pressure; uses minimum image convention.
4. `MDLJ.py` → contains implementation of Verlet Algorithm for 3D bulk system with interaction using Lennard-Jones potential.
5. `averaging.py` → gives thermodynamic averages of energies, temperature, and pressure.

We start from the reduced time step calculated from the empirical Lennard-Jones parameters,  $dt = 0.0022$ . We find it to be a moderate estimate, with fluctuations of around  $5 \cdot 10^{-4}$ .

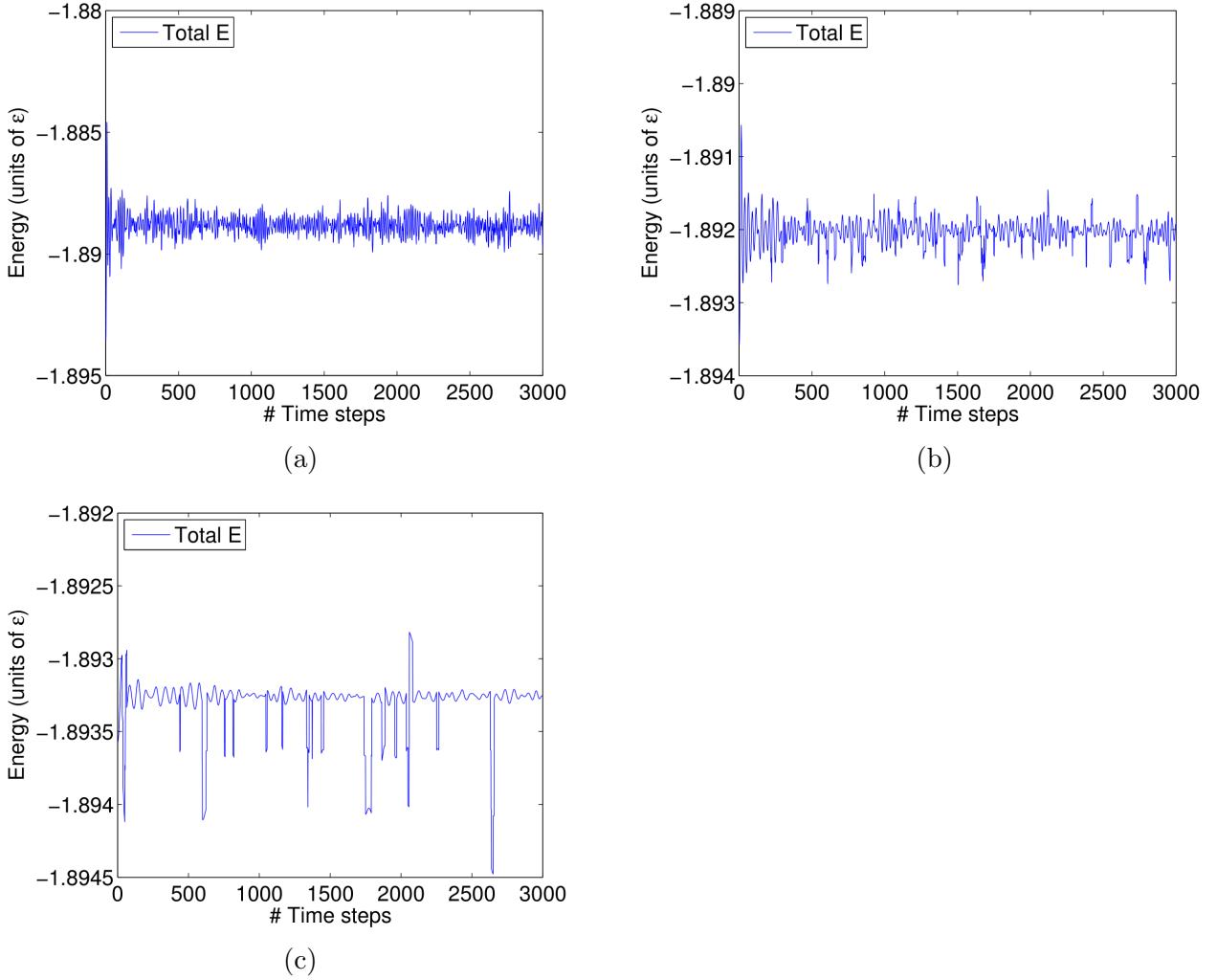


Figure 7: Convergence of fluctuations in total energy for varying time steps in reduced units, (a)  $dt = 0.004$ , (b)  $dt = 0.002$ , (c)  $dt = 0.001$

A larger  $dt = 0.004$  resulted in approximately factor of two larger range of fluctuations. A further reduction of  $dt = 0.001$  does not result in any less fluctuations of the total energy. We therefore keep the original  $dt = 0.0022$ , which has fluctuations within the acceptable order of magnitude for numerical variation. Going to smaller time steps may reach fluctuations more aligned with  $10^{-4}$  but would require more steps to simulate longer time scales.

Figure ?? illustrates the fluctuations in energy for the various  $dt$  time steps tested.

## 6 Exercise 6: Bulk FCC- Solid Phase

We choose parameters based on the phase diagram for Lennard-Jones materials that result in a solid phase. These parameters are

```

reduced density, \rho* = 1.4142
reduced temperature, T* = 1.0

```

The corresponding equilibration is presented in Figure ??.

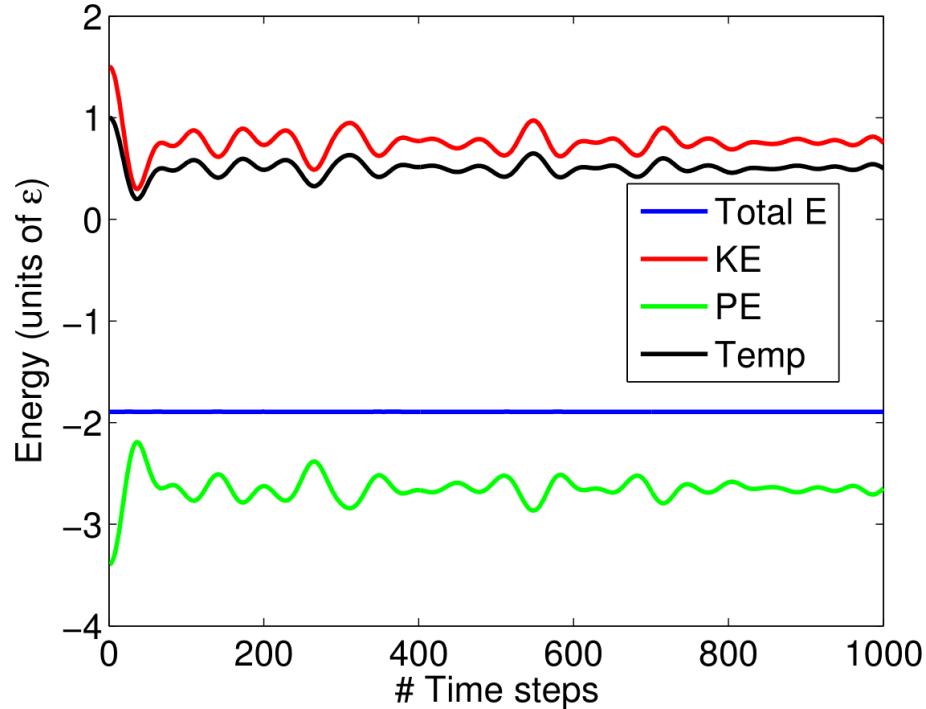


Figure 8: Equilibration of solid FCC Lennard-Jones phase for  $\rho^* = 1.414, t^* = 1.0$ .

It takes around 50 time steps for the run to equilibrate.

The average of thermodynamic quantities (without the equilibration period) are (in reduced units)

```

potential energy, U* = -2.6517
kinetic energy, K*      = 0.7583
energy, E*            = -1.8935
temperature, T*        = 0.5055
pressure, P*           = 64.5382

```

## 7 Exercise 7: Making the code more efficient

We rewrite the code in order to separately equilibrate the system, instead of having to for each molecular dynamics simulation.

We present the code used to invoke this new scheme.

```

% for Neon
eps = 0.0031; % eV
sigma = 2.74; % angstroms

nc = 3; % number repeated cells
tin = 1; % input temperature
% for solid
ao = sigma*2/sqrt(2); % convert from nearest neighbor distance to cubic lattice constant
n = 4*nc^3; % number of atoms
vol = (ao*nc)^3; % volume, not in reduced units
volred = (ao/sigma*nc)^3;
density = n/volred; % in reduced units
nsteps = 1e3; % number time steps
dt = 0.001; % time step

% equilibration
[n,x,y,z,vx,vy,vz] = initLJMD(nc,tin) ;
[un,kn,en,tn,pn,a,x,y,z,vx,vy,vz]= MDLJi(density,200,dt,n,x,y,z,vx,vy,vz);

% simulation from equilibrated parameters
[un,kn,en,tn,pn,a,x,y,z,vx,vy,vz]= MDLJi(density,nstems,dt,n,x,y,z,vx,vy,vz);

% averaging time sequence values
xmin = 1;
xmax = length(un);
[au,ak,ae,at,ap] = averaging(un,kn,en,tn,pn,xmin,xmax)

```

The modified `MDLKj.py` file has the initialization removed and input/output parameters altered to reflect this.

A set of averages with this newly modified molecular dynamics simulation.

```

potential energy = -2.6496
kinetic energy = 0.7570
total energy = -1.8926
temperature = 0.5046
pressure = 64.5482

```

These averages are not drastically different from the previously calculated averages and are not outside the range of numerical variation observed in general. Thus, the number of steps for these runs (i.e., 1000) allowed to run are sufficiently long enough to avoid the issue of drift in calculations.

## 8 Exercise 8: Bulk FCC- Liquid phase

We repeat the analysis as we did for solid phase Lennard-Jonesium for parameters that simulates liquid Lennard-Jonesium. For these runs, based on the phase diagram provided, we choose  $\rho^* = 0.4, T^* = 1$ . Figure ?? presents the evolution of the kinetic, potential, and total energy over time steps.

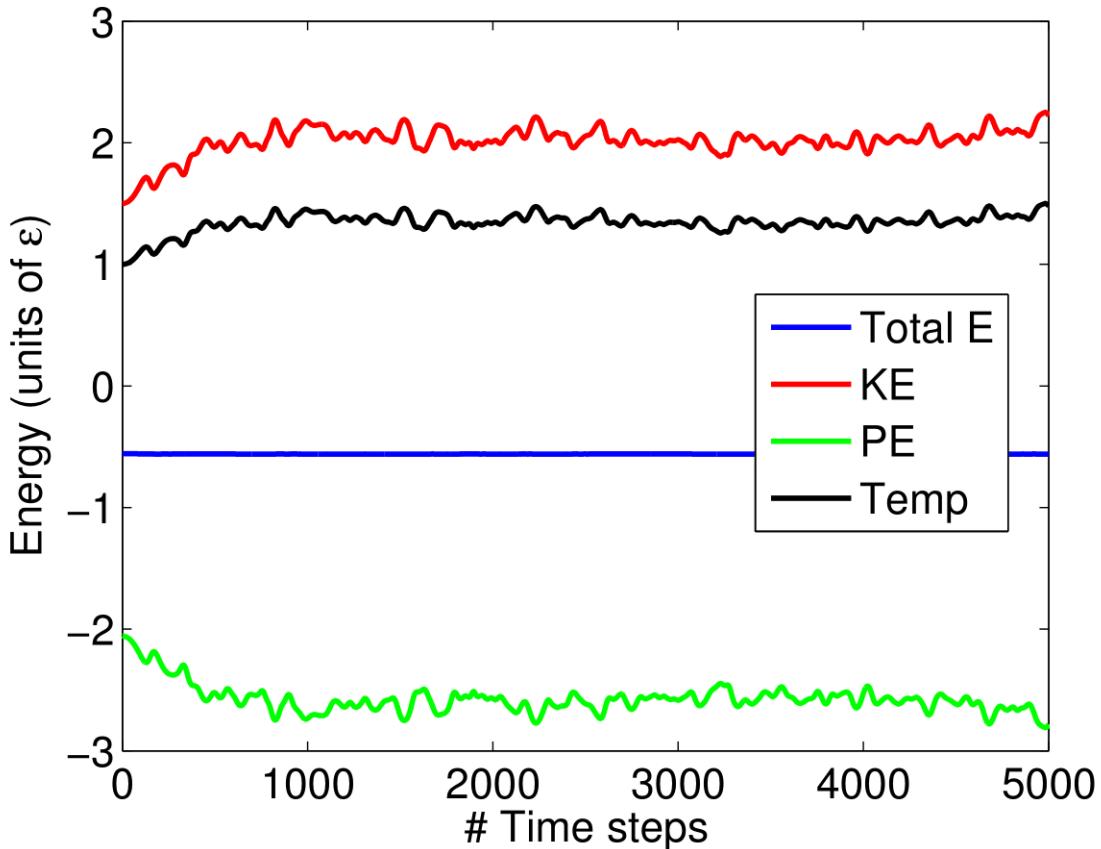


Figure 9: Tracking relation between kinetic, potential, and total energy for liquid phase of bulk FCC Lennard-Jonesium

Equilibration takes around 1000 time steps before equilibration is reached.

Below we present the average values of thermodynamic quantities for the liquid phase over 5000 time steps for the choice of  $\rho^* = 0.4, T^* = 1$  for the code with initialization and equilibration included in the molecular dynamics simulation (i.e., using `MDLJ.py`, not `MDLJi.py`).

```

U*   = -2.5835
K*   = 2.0234
E*   = -0.5601
  
```

```

T*      = 1.3489
P*      = 0.2161

```

Using instead `MDLJi.py` for comparison, we find the following average values for 5000 time steps (excluding equilibration),

```

U*      = -2.6144
K*      = 2.0539
E*      = -0.5606
T*      = 1.3693
P*      = 0.2262

```

As with the solid phase, the differences between the two methods of initialization and equilibration are similar to the fluctuations observed between each molecular dynamics run and are on the order of  $10^{-2}$  in reduced units for the time step size chosen. Drift is similar magnitude to the solid phase and to the variation that occurs between molecular dynamics runs of the same method. Longer runs may reduce the drift in both the case of the solid and liquid, but the length of runs presented presents a distinction enough between the solid and liquid phases.

## 9 Exercise 9: Calculating thermodynamic capacities

Using the time series data already available, we are able to calculate fluctuations in quantities of thermodynamic capacities. We present the variance of the various thermodynamic quantities so far for the solid and liquid phase. In addition to using the files mentioned in Exercise 5, we also use the file `variance.py` for calculating the variance.

We also include the averages for the same run for comparison.

For the solid phase, using a density  $\rho* = 1.414$  and  $T* = 1$  for 5000 time steps, we find the following

```

U*      = -2.6539
K*      = 0.7549
E*      = -1.8990
T*      = 0.5032
P*      = 64.4770

```

$$\sigma_U^* = -0.0029, \sigma_K^* = -0.0029, \sigma_E^* = -1.0232e-07, \sigma_T^* = -0.0013, \sigma_P^* = -0.0857,$$

For the liquid phase, using a density  $\rho* = 0.4$  and  $T* = 1$  for 5000 time steps (excluding equilibration), we find the following

```

U*      = -2.6417
K*      = 2.0797
E*      = -0.5620
T*      = 1.3865
P*      = 0.2321

```

$$\sigma_U^* = -0.0042, \sigma_K^* = -0.0042, \sigma_E^* = -4.7321\text{e-}07, \sigma_T^* = -0.0019, \sigma_p^* = -0.0301,$$

As seen from the above data, the liquid phase undergoes more fluctuations in its thermodynamic parameters than the solid phase. This is consistent with the picture of thermal vibrations between a solid and liquid phase. Additionally, the liquid phase is less energetically favorable than the solid ( $E_l^* > E_s^*$ ), and consequently have species with larger kinetic energy, also consistent with the intuitive picture of atomic motion in a liquid versus solid phase.

## 10 Exercise 10: Radial Distribution Function

We can learn more about the structure of the phases we simulated in previous exercises using the radial distribution function, which provides information on the probability and atom is located a distance  $r$  away from a reference atom. For crystalline solids, the radial distribution function is composed of defined peaks where as in less structured phases, such as liquid, the curve is more continuous and eventually asymptotes to unity.

We implement the calculation of the radial distribution function in the simulation for molecular dynamics, specifically when forces are calculated.

1. `initLJMD.py` → initialization of positions of  $n$  atoms; atoms are chosen to sit on *fcc* lattice sites.
2. `init3dMB.py` → initialization of velocities for  $n$  atoms in Maxwell-Boltzmann distribution centered around zero based off of desired temperature input for 3D system.
3. `fLJsumg.py` → contains calculation of forces for 3D Lennard-Jones system that includes calculation of the radial distribution function
4. `MDLJg.py` → contains implementation of Verlet Algorithm for 3D bulk system with interaction using Lennard-Jones potential and includes calculation of radial distribution function.

Figure ?? present the radial distribution function for the solid and liquid phase of the simulation cell size chosen, using 50 bins and 5000 time steps at  $T^* = 1$ . The plots cutoff at the size of the simulation cell, which is chosen to be 3 repeated unit cells in each direction.

The radial distribution function gives clear indication of the phase the material is in for the parameters tested.

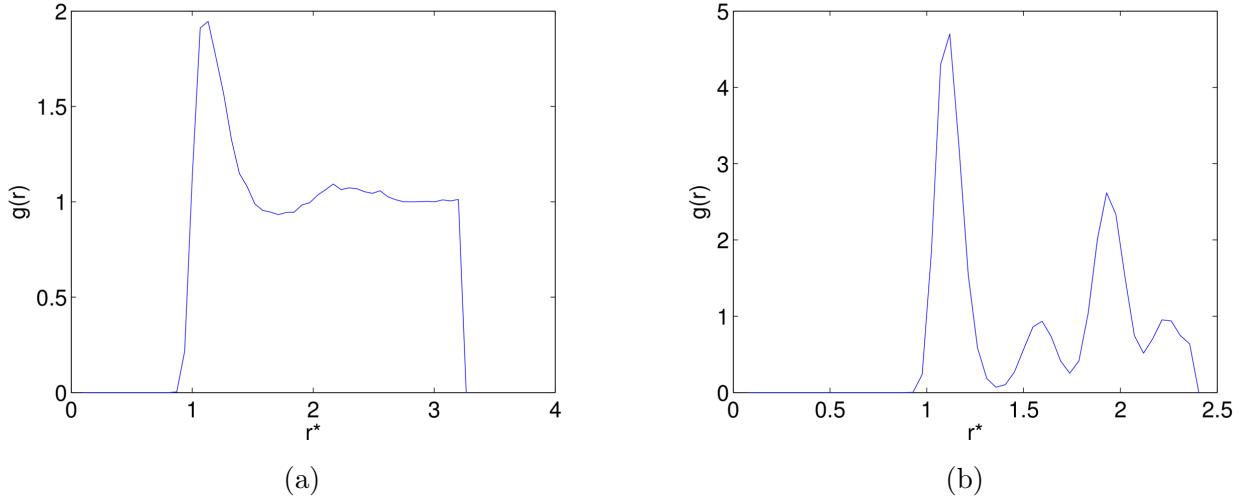


Figure 10: Radial distribution function for Lennard-Jonesium in (a) liquid phase,  $\rho^* = 0.4$  and (b) solid phase,  $\rho^* = 1$  at  $T^* = 1$

## 11 Exercise 11: Structural Order Parameter

We may also infer additional information from the structural order parameter. The structure order parameter for  $N$  atoms is defined as

$$\rho(\vec{k}) = \frac{1}{N} \sum_{i=1}^N \cos(\vec{k} \cdot \vec{r}_i) \quad (5)$$

For an *ccc* lattice,  $\vec{k} = \frac{2\pi}{a_{fcc}}(-1, 1, -1)$ , a reciprocal lattice vector, where  $a_{fcc}$  is the lattice constant of the unit cell, *not* the simulation cell size. However,  $\vec{r}_i$  are the position coordinates scaled to the simulation cell size. A perfectly *fcc* lattice will have  $\rho(\vec{k}) = 1$ , whereas a non-*fcc* phase will have  $\rho(\vec{k}) = 0$ . What this other phase is may be a liquid or a different crystalline phase.

Figure ?? presents the structural order parameters for the solid and liquid phases including equilibration to see the evolution of the structure factor for 5000 time steps.

As seen in Figure ??, both the liquid and solid phases begin as *fcc* structures with  $\rho(\vec{k}) = 1$ . As the liquid equilibrates,  $\rho(\vec{k})$  drops and oscillates around zero, indicating that it has lost its original *fcc* ordering. In contrast, the solid phase retains a fairly consistent  $\rho(\vec{k}) \approx 1$ , indicating it retained its *fcc* structure. It agrees with what we expect and what is observed in Exercise 10.

The structural order parameter provides additional quantitative information to the extent that a phase retains its starting structure, and is easier to visualize as a function time. It is also specific to a particular ordering, depending on the choice of  $\vec{k}$ , and so provides crystal system specific information that is hidden in the radial distribution function.

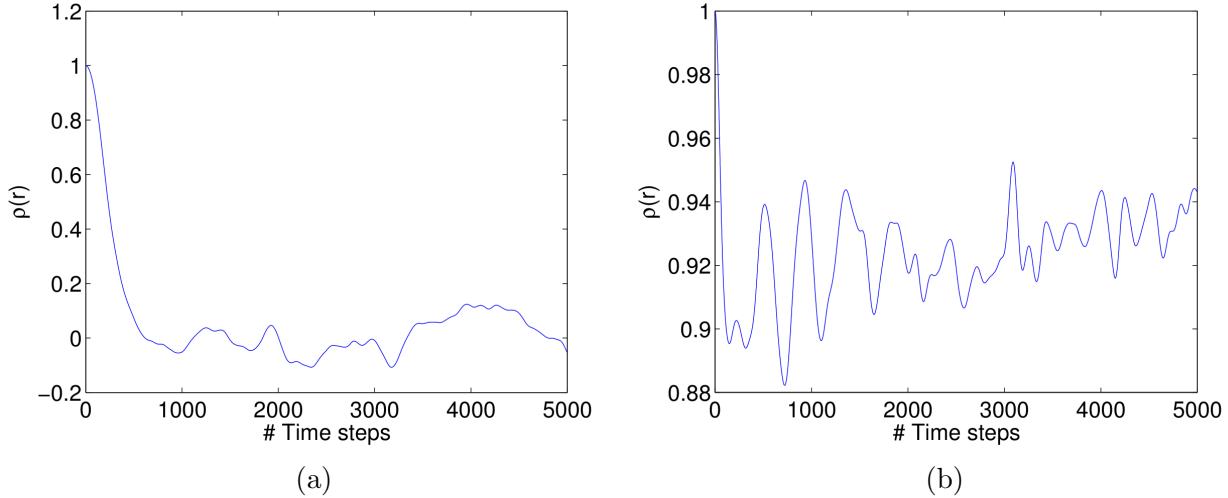


Figure 11: Structural order parameter for Lennard-Jonesium in (a) liquid phase,  $\rho^* = 0.4$  and (b) solid phase,  $\rho^* = 1$  at  $T^* = 1$

## 12 Exercise 12: Velocity Autocorrelation Function

We conclude with a calculation of the velocity autocorrelation function that is an average over all particles for the liquid and solid phase. We implement the velocity autocorrelation function in the less sophisticated way, so sizable fluctuations are expected. The velocity autocorrelation function is defined as

$$c_{vv} = \frac{m}{3k_b T} \langle \vec{v}(t) \cdot \vec{v}(0) \rangle \quad (6)$$

The corresponding velocity autocorrelation as a function of time step for the liquid and solid phases is plotted in Figure ??.

Autocorrelation functions measure the extent of correlation between two variables, specifically between two instances in time. With longer runs, the autocorrelation function is expected to decay to zero. This is indeed what is observed for both the liquid and solid phase. However, for this liquid phase, this decay is slow. This is rather unexpected as for a liquid, one would expect the many random collisions would cause a faster decay of correlation between velocities. The long tail decay that is observed suggests more complicated behavior. For the solid phase, one observes a negative autocorrelation at the onset of the simulation, which is behavior typically observed for denser systems at lower temperatures, corresponding to collisions with neighboring atoms that are like backscattering.

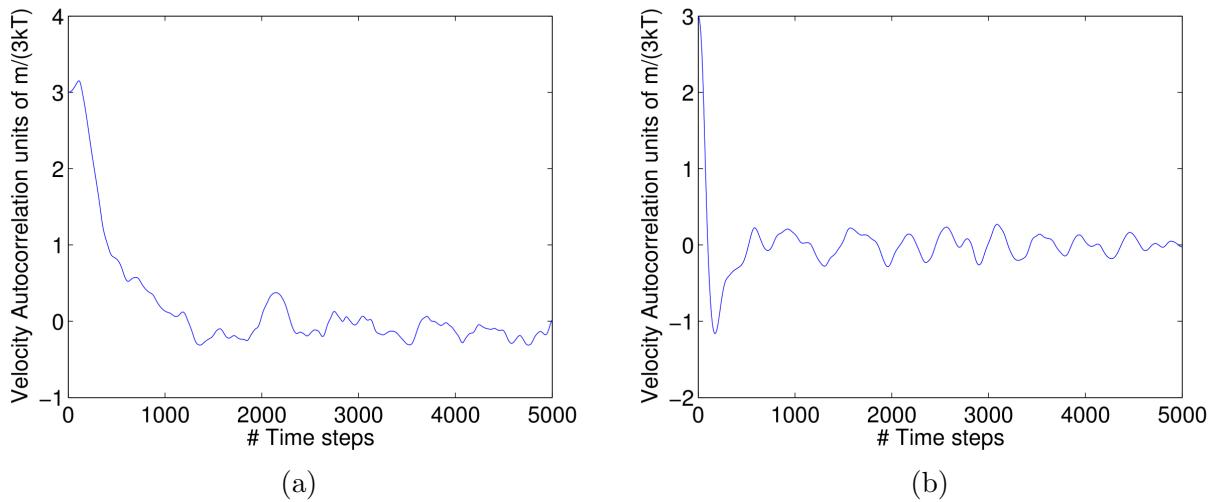


Figure 12: Velocity autocorrelation function for Lennard-Jonesium in (a) liquid phase,  $\rho^* = 0.4$  and (b) solid phase,  $\rho^* = 1$  at  $T^* = 1$