CVL867: Atomistic and multiscale modelling

Assignment 3

1) Write a function to calculate the potential energy between two particles interacting via Lennard jones potential given by following equation.

$$V_{\mathrm{LJ}}(r) = 4\varepsilon\left[\left(\frac{\sigma}{r}\right)^{12} - \left(\frac{\sigma}{r}\right)^{6}\right]$$

2) Using the above function write a function to calculate potential energy of the complete system taking into consideration the periodic boundaries and atom type information. Test the function on the following system, the configuration for which is provided.

Binary Lennard-Jones (LJ) $A_{80}B_{20}$ mixture

The system has two types of particles with composition $A_{80}B_{20}$ consisting of total N=100 particles in a cubic ensemble of size 4.3679 with periodic boundaries. The interaction between the particles is governed by

$$V_{\mathrm{LJ}}(r) = 4\varepsilon\left[\left(\frac{\sigma}{r}\right)^{12} - \left(\frac{\sigma}{r}\right)^{6}\right]$$

where 'r' refers to the distance between two particles, σ is the distance at which inter-particle potential energy is minimum and ε refers to the depth of the potential well. The LJ parameters are ε_{AA} = 1.0, ε_{AB} = 1.5, ε_{BB} = 0.5, σ_{AA} = 1.0, σ_{AB} = 0.8 and σ_{BB} = 0.88. The mass for all particles is set to 1.0. The interaction cutoff, $r_c = 2.5\sigma$ and the time step dt = 0.003.

Use following command to load the configuration as numpy arrays
positions=numpy.load("LJ_A80B20_3D_100_pos.npy")
species =numpy.load("LJ_A80B20_3D_100_species.npy")

positions : An array of shape (100,3) containing coordinates of particles
species : An array of shape (100,) containing atom type ('0' for A and '1' for B)

The simulation box is a cube with sides along cartesian axes and length equal 4.3679 starting from origin.

3) Write a program to perform energy minimization. Use any energy minimization algorithm discussed in class. You can also use functions from any standard optimization libraries.

4) Write a function to assign velocities to the particles corresponding to a given temperature using gaussian distribution. Use this function to assign velocities corresponding to a temperature of T=2.0 to the given LJ configuration.

5) Write a function to calculate force and acceleration on each particle for the given LJ interaction and perform one step of NVE simulation, using velocity-verlet algorithm.

6) Using above function generate a trajectory of 1000 steps, with timestep=0.003. Save it as a numpy array.

Bonus:

7) Write a function to save the trajectory in Ovito's (About OVITO – OVITO – Open Visualization Tool) xyz format, and visualise the obtained trajectory using Ovito software.