Project on Monte-Carlo and molecular dynamics simulations

Consider a one dimensional atomic chain consisting of N=1000 atoms interacting with each other by a Lennard-Jonnes potential:

$$u(r) = 4\varepsilon\left[\left(\frac{\sigma}{r}\right)^{12} - \left(\frac{\sigma}{r}\right)^{6}\right] \quad (1)$$

where the parameters ε and σ are given and equal to unity, i.e., ε=1.0, σ=1.0. Using the two atomistic simulation methods Monte-Carlo and molecular dynamics find the coefficient of thermal expansion for the chain.

1. Monte Carlo

Following the general procedure described in the class and using Matlab (or any other programming language) write a Monte Carlo (MC) program for studying thermal expansion of the molecular chain. To complete this task follow the steps outlined bellow:

a) Generate the initial chain configuration by assigning positions to all particles in the system. For example define the variable $x_o=2^{1/6}$ = 1.122462 and then initialize the components of the vector x(1000) holding the positions of all the particles. A reasonable choice for the initial position would to set all the particles equidistant along Ox axis such as:

x(1)=0.0, x(2)=x_0, x(3)=2x_0, …, x(1000)=999x_0. [The general relation is x(i)=(i-1)x_0, with i =1,…,1000]

b) Set the values for the variable controlling the Monte-Carlo displacement Δ. You may try values such as: Δ=0.015, Δ=0.01, Δ=0.005, Δ=0.001. Hint: The appropriate value of Δ to use in the simulations is the one that ensure an acceptance rate of about 50%. Moreover, the larger the temperature of the system the larger the values of Δ you may use.

c) Select the temperature of the system. You will perform simulations for five different temperatures: T_1=0.0, T_2=0.1ε/k, T_3=0.2ε/k, T_4=0.3ε/k and T_5=0.4ε/k . k is the Boltzmann's constant.

d) Start writing the actual Monte-Carl code. At this point you have selected the temperature T (one of the five values suggested) and Δ (I would suggest to try Δ=0.01):

1. Select a particle at random; that is generate an integer number, i, between 1 and N. You may use the following procedure: i = INT(rand()*N)+1. Here rand() generates a random number between 0 and 1
2. Calculate the energy of this particle U_i using $U_i = \sum_j^N u(|x(i)-x(j)|)$. When calculating this you might consider only the particles j, that are closer than r_0=2.5σ to particle i.

3. Give a small random displacement to the particle i. For example choose a trial move for the particle i from the position x(i) to a new position x'(i)=x(i)+(rand()-0.5)*Δ.
4. Again calculate the energy of particle i in this new position U_i' using $U'_i = \sum_j^N u(|x'(i) - x(j)|)$. Notice that all the other particles have the same coordinates as at the previous step.
5. Apply the Metropolis algorithm to decide whether or not the move of particle i is accepted. Calculate the energy change $\Delta U = U'_i - U_i$.
   a) If T=0K (zero temperature case) then:
      If (ΔU < 0) then accept x'(i) as the new position for particle i
      otherwise reject the trial move of particle i.
   b) If T≠0K,
      If (ΔU ≤ 0) then accept x'(i) as the new position for particle i .
      If (ΔU > 0) then generate a random number ζ∈(0,1) and if ζ < exp(-ΔU/kT)
      then accept x'(i) as the new position for particle i
      otherwise reject the trial move of particle i.

6. Go to point 1. and select randomly a new trial particle and continue with steps 1 through 5. Perform approximately 500000 MC steps, this means each particle is subject to MC trial move for about 500 times.
7. During the last, say 50000 MC steps evaluate the length of the chain at every 1000 MC steps, L=x(1000)-x(1) and evaluate the average <L>.

e) After running the Monte-Carlo code you just wrote for the five different temperatures you will obtain five different lengths L_1 , L_2, L_3, L_4 and L_5. By plotting L vs T estimate the coefficient of thermal expansion $\alpha_{MC}=\Delta L/\Delta T$ for this one-dimensional system in which the interaction is modeled by Lennard-Jones potential.

2. Molecular dynamics

Following the general procedure described in the class and using Matlab (or any other programming language) write a molecular dynamics (MD) program for studying thermal expansion of the molecular chain. To complete this task follow the steps outlined bellow:

a) Select the temperature of the system. You will perform simulations for four different temperatures: $T_1=0.1\varepsilon/k$, $T_2=0.2\varepsilon/k$, $T_3=0.3\varepsilon/k$ and $T_4=0.4\varepsilon/k$. k is the Boltzmann's constant. Use the reduced units for the atomic mass m=1.0.

b) Similar to the previous MC approach generate the initial chain configuration by assigning positions to all particles in the system. For example define the variable $x_o=2^{1/6}$ = 1.122462 and then initialize the components of the vector x(1000) holding the positions of all the particles. x(1)=0.0, x(2)=x_0, x(3)=2x_0, …, x(1000)=999x_0. [The general relation is x(i)=(i-1)x_0, with i =1,…,1000]

c) Generate the initial velocities for all the particles in the system such the kinetic energy is consistent with the initial temperature. First generate random velocities in the interval $(0, 2v_T)$ where $v_T = kT/m$. That is set $v(i)=rand()*2*v_T$ for $i=1,\ldots,1000$

d) Set the value for the variable controlling the molecular-dynamics integration time-step $\Delta t=0.0001$.

e) Start writing the actual molecular-dynamics code. At this point you have selected the temperature T (one of the four values suggested) .

1. Compute the forces on each particle in the system. For the Lennard-Jones interaction potential given by equation (1) the force f(r) acting on particle i due the presence of particle and j is given by:

$$f(r) = -\frac{du(r)}{dr}\frac{(x(i)-x(j))}{r} = \frac{48\varepsilon}{r}\left[\left(\frac{\sigma}{r}\right)^{12} - 0.5\left(\frac{\sigma}{r}\right)^{6}\right]\frac{(x(i)-x(j))}{r} \qquad (2)$$

where $r=|x(i)-x(j)|$ is the distance between particles i and j. When calculating the total force on particle i, sum all of the interaction forces with particles closer than $r_0=2.5\sigma$

2. Using the Velocity-Verlet algorithm integrate the equation of motion for each particle in the system. That is, find the new positions x(i) and new velocities at time $t+\Delta t$ for all of the particles in the system.

3. Every 10 MD time-steps evaluate the actual temperature in the system and rescale the velocities of all of the particles in the system such that the new temperature is the desired temperature for the simulation. To perform this task calculate the actual temperature T' given by:

$$T' = \sum_{i=1}^{N} \frac{mv(i)^2}{3Nk} \qquad (3)$$

then rescale the velocities of all the particles in the system using:

$$v(i) \rightarrow \sqrt{\frac{T}{T'}}\, v(i) \qquad . \qquad (4)$$

4. Repeat steps 1. through 3. for a total of Nsteps=100000 MD steps.

5. During the last, 10000 MD steps evaluate the length of the chain at every 500 MD steps, L=x(1000)-x(1) and evaluate the average <L>.

e) After running the MD code for the four different temperatures you will obtain four different lengths L_1 , L_2, L_3 and L_4. By plotting L vs T estimate the coefficient of thermal expansion $\alpha_{MD}=\Delta L/\Delta T$ for this one-dimensional system in which the interaction is modeled by Lennard-Jones potential.

3. Discussions and Conclusions

Write a report discussing all the steps required to accomplish this project. Comment on the values of coefficients of thermal expansion obtained for the Lennard-Jones chain by using the Monte-Carlo and molecular dynamics simulation methods.