

7. Molecular Dynamics

One way to investigate the behaviour of a many particle system, such as a liquid, is to begin from the known intermolecular interactions and do a computer simulation of the dynamics. This approach involves tracking the behaviour of individual particles and is known as the molecular dynamics method. The method has been applied successfully to systems of several hundred to a million particles. However a knowledge of the individual particle trajectories is not helpful unless we can extract information relating to measurable properties, for example temperature, from the simulations. In this exercise you will set up a molecular dynamics simulation for a gas with a small number of molecules and use it to investigate the temperature and speed distribution of the system.

The Intermolecular Potential:

The first step is to specify the model system. For simplicity assume that the dynamics can be treated classically and that the molecules are spherical and chemically inert. Assume that the force between any pair of molecules depends only on the separation between them. In this case the potential energy, U , is a sum of two particle interactions:

$$U = u(r_{12}) + u(r_{13}) + \cdots + u(r_{23}) + \cdots = \sum_{i=1}^{N-1} \sum_{j=i+1}^N u(r_{ij}) \quad (1)$$

Here each $u(r_{ij})$ depends only on the magnitude of the distance between the particles and the exact form depends on the nature of the system. For a liquid such as argon this potential has two contributions, a strong repulsion for small values of r and a weak attraction at large values of r . The repulsion is a consequence of the Pauli exclusion principle which forces electrons to be in different quantum states. This repulsion distorts the electron clouds of two particles so that they do not overlap. The dominant weak attraction at larger r is due to the mutual polarisation of each particle and is known as the van der Waals force. One of the most common forms of the potential between two particles is the Lennard-Jones potential:

$$u(r) = 4\epsilon \left[\left(\frac{\sigma}{r} \right)^{12} - \left(\frac{\sigma}{r} \right)^6 \right] \quad (2)$$

The $\left(\frac{1}{r}\right)^{12}$ form of the repulsion is chosen for convenience only. The Lennard-Jones potential is parameterised by a length, σ , and an energy, ϵ . Values of $\epsilon = 1.65 \times 10^{-21}$ J and $\sigma = 3.4 \times 10^{-10}$ m give good agreement with the experimental values for liquid argon.

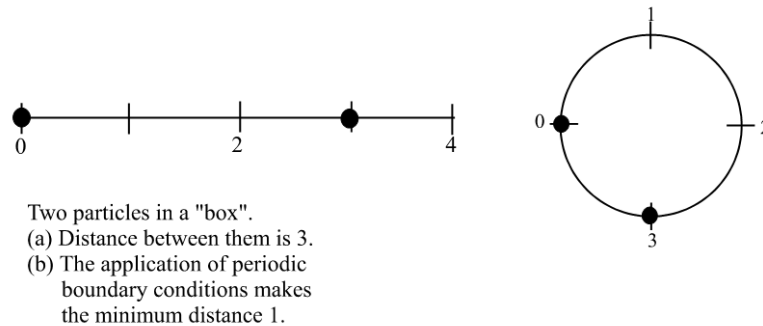
A:

Write a short program to plot the Lennard-Jones potential. Use the values for the parameters σ and ϵ given above. What is the significance of the two parameters? Determine an expression for the corresponding force and plot the magnitude of this force as well. At what value of r does the force equal zero?

Boundary Conditions

A useful simulation must incorporate all of the relevant features of the system of interest. For small simulations a significant fraction of the particles will be near the walls of the system and hence the behaviour of the system will be dominated by surface effects rather than the properties of the system itself. The most common way of minimising these surface effects is to use periodic boundary conditions. First think about a “one-dimensional box” of N particles that are constrained to

move along a straight line of length L . The ends of the line serve as imaginary walls. The application of periodic boundary conditions is equivalent to thinking of the line as a circle. The distance between particles is then measured along the circumference of the circle and therefore the maximum separation between the particles is $L/2$. In two-dimensions the application of periodic boundary conditions joins the opposite edges of a “box” so that the box becomes the surface of a torus.



The motivation for using periodic boundary conditions is that it results in a simulation in which there is no boundary and in which every point within the cell is equivalent. Imagine a set of N particles in a two-dimensional cell, the periodic boundary condition implies that this cell is duplicated an infinite number of times to fill two-dimensional space. Each imagine cell contains the original particles in the same relative positions (Fig 2).

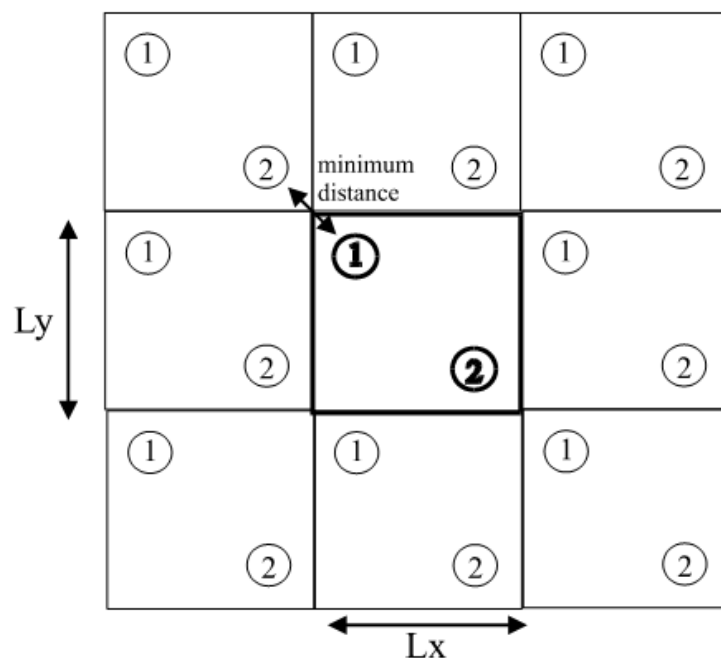


Figure 1: MD

As a particle moves in the central cell, its periodic images move in the same way so that only the motion in the central cell need be followed. When a particle enters or leaves the central cell, this is accompanied by an image of that particle leaving or entering a neighbouring cell through the opposite face. The total force on a given particle i is that due to the combination of the actions of the forces of every other particle within the central cell and also with all the contributions from

the image particles. In general there are an infinite number of contributions. However, for short-range interactions it is sufficient to only consider the contribution from the nearest image. This approximation implies that particle i interacts only with the nearest instance of particle j . The minimum image approximation implies that the calculation of the total force on all N particles due to pairwise interactions involves $N(N-1)/2$ contributions.

The Program

B:

Write a program to carry out a molecular dynamics simulation of a two-dimensional liquid argon system. The system is deterministic so that the motion depends on the initial conditions. To track the motion of each particle you will have to determine the force on each particle, remember that in this model it is sufficient to consider only the interaction of particle i with the nearest instance of every other particle (j). It is also a good idea to use Newton's third law to reduce the number of calculations by a factor of two. The next step in the simulation will be to determine the change in position and velocity of the particle after a time interval Δt . The Verlet algorithm is suitable for this problem:

$$x_{n+1} = x_n + v_n \Delta t + \frac{1}{2} a_n (\Delta t)^2 \quad (3)$$

$$v_{n+1} = v_n + \frac{1}{2} (a_{n+1} + a_n) \Delta t \quad (4)$$

Here a is the acceleration, v the velocity and x the position. As the problem is in two-dimensions there will be an equivalent set of code for the y direction.

Program Structure:

Initial Conditions: Start by using $N = 16$ particles and a time step of 0.01 for the Verlet algorithm. Use a square cell of length 6 and the initial positions and velocities specified below.

$x = [1.09, 3.12, 0.08, 0.54, 2.52, 3.03, 4.25, 0.89, 2.76, 3.14, 0.23, 1.91, 4.77, 5.1, 4.97, 3.9]$
 $y = [0.98, 5.25, 2.38, 4.08, 4.39, 2.94, 3.01, 3.11, 0.31, 1.91, 5.71, 2.46, 0.96, 4.63, 5.88, 0.2]$
 $v_x = [-0.33, 0.12, -0.09, -1.94, 0.75, 1.7, 0.84, -1.04, 1.64, 0.38, -1.58, -1.55, -0.23, -0.31, 1.18, 0.46]$
 $v_y = [0.78, -1.19, -0.1, -0.56, 0.34, -1.08, 0.47, 0.06, 1.36, -1.24, 0.55, -0.16, -0.83, 0.65, 1.48, -0.53]$

To reduce the possibility of round off error, it is useful to choose units so that computed quantities are neither too small or too large. In this simulation it is convenient to choose the Lennard-Jones parameters σ and ϵ to be the units of distance and energy. The unit of mass is the mass of one atom and all other quantities are expressed in terms of σ , ϵ and m (see table) and all variables in the program are in these reduced units.

Quantity	Unit	Argon value
length	σ	$3.4 \times 10^{-10} \text{ m}$
energy	ϵ	$1.65 \times 10^{-21} \text{ J}$
mass	m	$6.69 \times 10^{-26} \text{ kg}$
time	$\sigma(m/\epsilon)^{1/2}$	$2.17 \times 10^{-12} \text{ s}$
velocity	$(\epsilon/m)^{1/2}$	$1.57 \times 10^2 \text{ m/s}$
force	ϵ/σ	$4.85 \times 10^{-12} \text{ N}$
temperature	ϵ/k (Boltzmann constant)	120 K

Force Calculation:

In order to determine the force on each particle it is necessary to calculate the minimum separation between each pair of particles. This is equivalent to finding the nearest instances of the two particles. The following code does this, make sure that you are happy with it before incorporating it into your program.

```
dx=x(i)-x(j)
if dx>0.5*Lx
    dx=dx-Lx;
elseif dx <-0.5*Lx
    dx=dx+L;
end
```

In reduced mass units the mass of each particle is unity and so the acceleration and force are equivalent. Once you have calculated the initial force on each particle, calculate the potential and kinetic energy of the system and determine the number of particles that are on the left-hand side of the box.

System Evolution:

Next determine new positions and velocities of the particles and track the behaviour of the system. You will have to check to see if each particle crosses a boundary. If a particle leaves the box, add or subtract L_x or L_y to the appropriate co-ordinate to bring the particle back into the box on the opposite side. Make sure that the program outputs enough information to monitor the system's evolution effectively.

C:

Check that your program is working properly. The total energy of the system should be approximately conserved and all the particles should remain inside the box. Plot the number of particles on the left-hand side of the box as a function of time. Also compute the time average of this number and plot this as a function of time. What is the mean number of particles on the left of the box as the system approaches equilibrium? Is the behaviour as you expected?

Sensitivity to Initial Conditions:

D:

Consider the following initial conditions corresponding to $N = 11$ particles moving in the same direction with the same velocity. Choose $L_x = L_y = 10$ and $\Delta t = 0.01$. Use the following initial conditions:

$$\begin{aligned}x(i) &= L_x/2 \\y(i) &= (i - 0.5) * L_y/N \\v_x(i) &= 1 \\v_y(i) &= 0\end{aligned}\tag{5}$$

Does the system eventually reach equilibrium? Change the velocity of particle 6 so that $v_x(6) =$

0.99 and $v_y(6) = 0.01$. How does this change affect the behaviour of the system? What does this indicate about sensitivity to initial conditions?

Thermodynamic Quantities:

For a molecular dynamics simulation to be useful it is important to be able to extract information about measurable quantities, such as temperature, from the model. The kinetic definition of temperature follows from the equipartition theorem: each quadratic term in the energy of a classical system in equilibrium at temperature T has a mean value equal to $\frac{1}{2}kT$. The temperature $T(t)$ at a time t is therefore given by the relation:

$$N \frac{d}{2} kT(t) = K(t)$$

or

$$kT(t) = \frac{2}{d} \frac{K(t)}{N} = \frac{1}{dN} \sum_{i=1}^N m_i \vec{v}_i(t) \cdot \vec{v}_i(t)$$

Here K represents the total kinetic energy of the system and d the number of spatial dimensions. The mean temperature can be expressed as the time average of $T(t)$ over many configurations of the particles. For two-dimensions, the mean temperature is therefore;

$$T = \bar{T} = \frac{1}{2N} \sum_{i=1}^N \overline{m_i \vec{v}_i(t) \cdot \vec{v}_i(t)}$$

where the bar signifies a time average. These expressions are only valid if the centre of mass momentum is zero. This is true in this simulation but this constraint removes two degrees of freedom so that the denominator on the right of the last expression should read $2(N-1)$ not $2N$.

E:

Using the initial conditions specified in **B**, modify the program so that the values of the time average of the temperature are displayed. Start calculating this quantity only after the system has come to equilibrium. What is the equilibrium value of temperature? Is it consistent with the kinetic energy of the system? Remember that you have been working in reduced units.

Distribution of speeds:

F:

Modify the program to determine the equilibrium probability $P(v)\Delta v$ that a particle has a speed between v and $v + \Delta v$. To do this estimate a value for the maximum speed that you will encounter. Choose bins of width $dv = v_{max}/n_{bin}$ and take n_{bin} to be 50. Once equilibrium has been reached, build up your probability distribution at each time step. You will probably need at least 500 time steps to accumulate a reasonable distribution. Normalise your probability density by dividing by the number of particles and the number of time steps. Plot the probability density $P(v)$ against v . Is the form consistent with the Maxwell-Boltzmann distribution in two-dimensions:

$$P(v)dv = A e^{-\frac{mv^2}{kT}} v dv$$