

## Chapter 4

# Molecular Simulation

### 4.1 Introduction

With the development of high performance computing, molecular simulation methods are increasingly more and more relevant in science and technology. In this thesis we use molecular simulation methods to study some properties of shearing simple fluids.

Statistical mechanics is an indispensable tool for studying the properties of materials at the nano/microscale. In theory, if we understand the interactions within and between molecules, we can obtain the material properties of substances from statistical mechanics. Except for some simple and ideal cases, analytic problems in statistical mechanics are either only soluble by approximate methods or else are intractable. Molecular simulation has a valuable role to play in providing essentially exact results for problems in statistical mechanics [All89, Han76]. For a given tractable model, material properties may be obtained from molecular simulation. Molecular simulation can be used as a test of theories. One can also compare the results of molecular simulation to those of real experiments, thus checking if the model is a good one. If the model is good, molecular simulation can also supply much useful information to both theoreticians and experimentalists in which both theory and/or experiment are currently difficult to perform. Simulation has therefore two functions: as a bridge connecting models to theory and as a bridge connecting models with experiment.

Given microscopic details of a system (such as atomic mass, molecular structure, interactions between particles, etc.) one can use molecular simulation to obtain macroscopic properties of the system (such as thermodynamic quantities, transport coefficients, equations of state, etc.). The relation between microscopic quantities and macroscopic quantities is both of academic interest and technologically useful. What is more, molecular simulations can give information on some important quantities which are not easy or even impossible to get from experiments. For instance, it is difficult to carry out experiments under extreme thermodynamic conditions. However, using molecular simulation, one can readily obtain quantitative results of, say, a shock wave experiment, a high-temperature plasma, a nuclear reactor, star evolution, and so on.

One of the most common applications of molecular simulation is to predict the properties of materials, and this is most commonly done by employing both Monte Carlo (MC) and molecular dynamics (MD) computing methods.

The Monte Carlo method is based on probabilities [Fre96, Sad99]. It does not follow the time evolution of configurations and therefore is devoid of dynamics. It involves the computation of probabilities of generating arbitrary molecular configurations. To simulate a fluid system, we allow the system to transit between different states of configurations. These transitions are achieved by the following three steps. Firstly a trial configuration is generated randomly. Then, one evaluates an ‘acceptance criterion’ by calculating the change in energy and other properties in the trial configuration. Finally, the acceptance criterion is compared to a random number to decide whether to accept or reject the trial configuration. To determine accurately the properties of the system in a finite time, it is important to sample those states that make the most significant contributions. This is achieved by generating a Markov chain. A Markov chain is a sequence of trials in which the outcome of successive trials depends only on the immediate predecessor. In a Markov chain, a new state will be accepted with a higher probability if it is more ‘favorable’ than the existing state. This usually means that the new trial state has a lower energy.

In Monte Carlo simulations, a trial configuration is often obtained by displacing, exchanging, removing or adding a molecule. Probabilities are also involved in determining the nature and extent of the attempted move. The transition probability is dependent on what ensemble is chosen, but it always involves evaluating the energy of the new configuration and comparing it with the energy of the existing (old) state. Note that if the attempted configuration is rejected, then the old state is counted again as the new state.

## 4.2 Molecular dynamics

The Monte Carlo method relies on transition probabilities. Molecular dynamics (MD), however, is different to the Monte Carlo method, in that new configurations are obtained by solving the equations of motion of the molecules [All87, Rap95, Sad99]. At first, a system of  $N$  particles is given an initial condition, that is, the coordinates and momenta of each particle. Then the equations of motion are used to calculate step by step the next sequence of coordinates and momenta. In this way, the system evolves dynamically in time and we can calculate the time average of phase variables. Because molecular dynamics follows time evolution, MD simulation can be used to obtain time-dependent properties of the system. Generally, the MD approach is intrinsically more numerically intensive than the Monte Carlo method and the program takes considerably longer time to run.

Consider a system of  $N$  particles. The interaction between these particles is determined by the intermolecular potential  $u(\mathbf{r}_1, \mathbf{r}_2, \dots, \mathbf{r}_N)$ . In a molecular dynamics simulation one needs to solve Newton's equations of motion for classical systems:

$$\mathbf{F}_i = m_i \ddot{\mathbf{r}}_i \quad (4.1)$$

These are second order differential equations. In practice, it is more convenient to express these equations as two first order differential equations, one involving time derivatives of the momenta, and another expressing time derivatives of the particle positions,

$$\dot{\mathbf{r}}_i = \frac{d\mathbf{r}_i}{dt} = \frac{\mathbf{p}_i}{m} \quad (4.2)$$

$$\dot{\mathbf{p}}_i = \frac{d\mathbf{p}_i}{dt} = \mathbf{F}_i \quad (4.3)$$

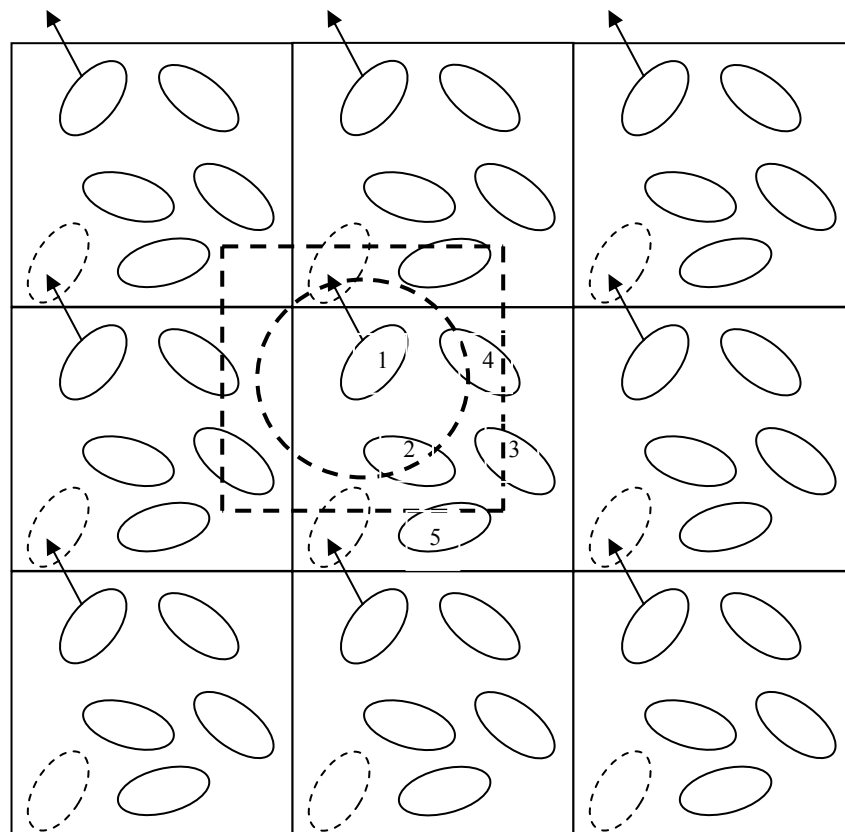
$\mathbf{r}_i$  is the position of particle  $i$ ,  $\mathbf{p}_i$  is its momentum and  $\mathbf{F}_i$  is the total force acting on the particle ( $\mathbf{F}_i = -\nabla_i u$ ; If the potential is pairwise additive, then,  $\mathbf{F}_i = \sum_{j \neq i}^N \mathbf{F}_{ij} = -\sum_{j \neq i}^N \frac{du_{ij}}{d\mathbf{r}_{ij}}$ ).

In general, the larger the size of the simulation system, the more accurate the simulation results will be. However because computer memory is limited and a large number of particles greatly increases computing time, the number may be from ten to several thousands, though we note that with teraflop computing now available it is possible to perform MD simulations of several hundred million atoms. In molecular dynamics, the force calculation occupies the majority of the simulation time. In every loop, the force computation time is proportional to the square of the particle number. Some special techniques may reduce this dependence to some degree ( eg. Cell neighbour lists can reduce this dependence from  $N^2$  to  $N$  [All89] ). Clearly, if we increase the number of particles, the time of execution of the program will increase more rapidly.

The  $N$  particles (atoms, molecules or ions) may be confined by a container, which prevents them from drifting apart. But these arrangements are not good for the simulation of bulk liquids, due to surface effects. When one does such a simulation, there will always be a large fraction of molecules which lie on the surface of the cube containing these particles. Molecules on the surface will have different behavior from molecules in the bulk.

To overcome the obstacle of surface effects, periodic boundary conditions are commonly applied. The original simulation cubic box, with the states of all the molecules within it, is replicated throughout the whole of space. The simulation box and the replicated boxes together form an infinite lattice. During the simulation, when a molecule in the original box moves, its periodic images in each of the surrounding boxes move in exactly the same way.

If a molecule leaves the central box, one of its images will enter the box through the opposite face. There are no walls at the boundary of the original box, and there are no surface molecules. Figure 4.1 shows the periodic boundary condition in two-dimensions. The number of molecules in the central box (and hence in the entire system) is conserved, and so are the “virtual” molecules in all the other periodic boxes. It is neither necessary nor possible to store the coordinates and the momenta of all the images in a simulation. In fact, there are an infinite number of such images. Instead we just need to record those of the molecules in the central box. All the coordinates and momenta of other images can be obtained from the translation of the central box. Periodic boundary conditions are often applied to simulate bulk properties, because they minimize the surface effects.



**Figure. 4.1 Diagram demonstrating periodic boundary condition, minimum image convention and cutoff.**

Now we consider how to evaluate properties of systems subject to periodic boundary conditions. The main part of the molecular dynamics programs comprises the calculation of the potential energy of a system in a particular configuration, and the forces acting on each molecule. Consider how to calculate the force on molecule 1 in Figure 4.1, and the portion of the potential energy involving molecule 1. Here we assume that the potential is pairwise additive, while the effect of three-body or higher level many body interactions can be ignored. First, we include the interactions between particle 1 and all the other particles in the simulation box. In addition, we should include the interactions between particle 1 and all the images in those replica boxes. So there are an infinite number of interactions all together and it is impossible to calculate them all in practice. In order to deal with this problem, we often use an approximation called the “minimum image convention”. If the force is short-ranged, we can only consider those particles close to particle 1, while other interactions with far distant particles can be ignored. Taking particle 1 as a center, we can construct a box which has the same size and shape of the original box (see Figure 4.1, the dashed cube). We only consider the interactions between particle 1 and all the other particles whose centers lie in this ‘minimum image convention’ box. There are only  $N-1$  summations involving particle 1 now.

Under the minimum image convention, there are  $\frac{1}{2}N(N-1)$  terms involved in calculating the energy and force. For simulating a few hundred to several thousand particles, this is still computationally intensive. We can significantly reduce this task further by another approximation, that is, potential truncation. From the same consideration as ‘minimum image convention’, we know that for a short-range potential, the main contribution to energy and force comes from nearby particles. We can therefore truncate the potential at some distance,  $r_c$ . We approximate the potential as zero at distances larger than  $r_c$ . This truncation distance is also called cutoff distance. Of course, we should select a big enough cutoff distance to ensure it will not change the properties of the material significantly. We should also note that our cutoff distance must be not larger than half the box length to keep it consistent with the “minimum image convention”.

### 4.3 Intermolecular Potential

The properties of a fluid are a manifestation of the interaction between molecules. This interaction is quantitatively described by the system's intermolecular potential. The calculation of the potential and subsequent force calculations is the most time consuming and crucial step in molecular dynamics.

To evaluate the interactions between particles, we need to have a suitable potential function. In general, the system potential energy ( $E_p$ ) can be expressed as the following formula [Sad99]:

$$E_p = \sum_i u_1(r_i) + \sum_i \sum_{j>i} u_2(r_i, r_j) + \sum_i \sum_{j>i} \sum_{k>j>i} u_3(r_i, r_j, r_k) + \dots \quad (4.4)$$

where the first term denotes the effect of external fields and other terms denote the interactions between particles. Among them,  $u_2$  is the potential between pairs of particles,  $u_3$  is the potential between triplets, and  $u_4$ ,  $u_5$ , etc, are due to quadruplets and quintuplets, respectively, etc..

The two-body interaction, or pairwise potential part, makes most contribution to particle interaction and system energy. So usually we ignore higher order interactions. However in some cases, three-body interactions may be important. The higher the order of the many-body interaction, the more time is needed in the simulation. If the order of many-body interactions is  $m$ , then, in one loop for calculating the potential, there are  $N^m$  terms to calculate. It is apparent that increasing the order of many-body interactions will rapidly increase the time necessary for simulation, which is the main reason why only two-body interactions are used. A common compromise is to use an effective two-body interaction (such as the Lennard-Jones potential), that, while being a pairwise additive potential, includes the spherically averaged effects of all orders of many-body interactions.

In our work we consider only effective two-body potentials. Many pairwise potentials have been developed and applied to atoms. Pair potentials for atoms can be used in the molecular simulation of polyatomic molecules and even macromolecules because molecules are composed of atoms. Hence, the atomic pairwise potential is a foundation for evaluating molecular properties.

The Lennard-Jones potential is probably the most widely used pairwise potential for molecular simulation [Han76]. The most common form of the Lennard-Jones potential is

$$u(r) = 4\varepsilon \left[ \left( \frac{\sigma}{r} \right)^{12} - \left( \frac{\sigma}{r} \right)^6 \right] \quad (4.5)$$

where  $r$  is the interatomic separation,  $\sigma$  is the collision diameter, which is the separation of the particle at which  $u(r)=0$ , and  $\varepsilon$  is the depth of the potential well at the minimum in  $u(r)$ . The Lennard-Jones potential provides an adequate description of the interaction between pairs of inert gas atoms and also of some molecules.

The Weeks-Chandler-Anderson (WCA) potential is another pairwise potential which is frequently used in molecular simulations [Wee71]. It has the form:

$$u(r) = 4\varepsilon \left[ \left( \frac{\sigma}{r} \right)^{12} - \left( \frac{\sigma}{r} \right)^6 \right] + \varepsilon, \quad r < 2^{1/6} \sigma$$

$$= 0, \quad r > 2^{1/6} \sigma \quad (4.6)$$

The WCA potential is a modified Lennard-Jones potential. We truncate the Lennard-Jones potential at a value of  $r = 2^{1/6} \sigma$ , where the potential has the minimum value  $-\varepsilon$ . Then we shift the potential up a value  $\varepsilon$ , so at the cutoff distance the potential is zero. For distances beyond the cutoff the potential is also zero. The WCA potential has its own advantage.

Compared with the Lennard-Jones potential, it is extremely short ranged so it can save considerable simulation time than the long ranged Lennard-Jones potential. Compared with the hard-sphere potential, it is continuous and softer so it is more realistic than ideal. Furthermore, as the transport properties of fluids are dominated by repulsive interactions, the essential physics is still retained by the WCA potential.

The Barker-Fisher-Watts (BFW) potential [Bar71] is a highly accurate interatomic potential that is specific to the noble gases, such as krypton and xenon. The BFW potential has the following form

$$u(r) = \varepsilon \left[ \sum_{i=0}^5 A_i (x-1)^i \exp[\alpha(1-x)] - \sum_{j=0}^2 \frac{C_{2j+6}}{\delta + x^{2j+6}} \right] \quad (4.7)$$

Here,  $x = r/r_m$  where  $r_m$  is the interatomic separation at which the potential has the minimum value. The other parameters are constants for a particular atom obtained by fitting the potential to experimental data for molecular beam scattering, second virial coefficients, and long-range interaction coefficients. We note that the BFW two-body potential is a genuine two-body potential, not an ‘effective’ potential. As such it needs to be complemented with at least a three-body potential (such as Axilrod-Teller potential) for accuracy. However, in this work the properties of interest are dominated by two-body interactions [Mar00], and so we ignore inclusion of three-body interactions.

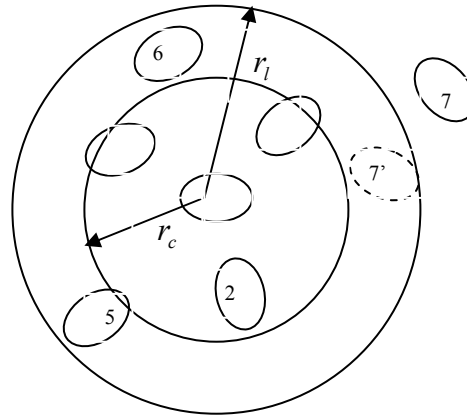
#### 4.4 Neighbour list

From the above description, we know in principle how to perform a molecular dynamics simulation. One can take  $N$  particles within a cube as a studied system and use periodic boundary conditions to reduce surface effects. When one evaluates the interaction between one particle  $i$  with all other particles  $j$ , one can use the minimum image convention and consider only those  $N-1$  particles around this molecule in a space just equal to the original

cube. So the minimum image convention can reduce the number of particles need to be considered from infinite in the whole space to just those in one cube. By using a cutoff radius, one can further reduce this number from one minimum image convention box to a sphere of radius  $r_c$ , where the volume of the sphere is smaller than that of the cube.

In every time step of a molecular dynamics simulation, we need to calculate the potential energy and forces. When we calculate these quantities associated with one particular molecule  $i$ , we first calculate the separations or distances between this molecule and all other molecules in a minimum image convention box, so as to decide whether their separations are greater or smaller than the cutoff radius. However, during certain time steps, the particles close to particle  $i$ , ( the “neighbours” of particle  $i$  ) are not changed. So during these time steps, we do not need to calculate the separations from  $i$  to every particle in the whole minimum image convention box. We only need to consider those close to  $i$ , or its neighbours whose separations from  $i$  are a little larger than the cutoff radius. After a certain number of time steps, we can update the neighbours. In this way, one can save time and increase the speed of the program.

Here we describe the neighbour list method we used [All89]. In Figure 4.2, we enclose particle 1 within a sphere whose radius  $r_l$  is larger than the potential cutoff  $r_c$ . Beyond  $r_c$  there is thus a ‘skin’ or layer whose thickness is  $r_l - r_c > 0$ . We suppose that during a certain number of time steps, only the particles in this larger sphere may have separations within the potential cutoff. Other particles, like particle 7, cannot penetrate the ‘skin’ and enter the region of the sphere with radius  $r_c$  during this time. So at the beginning of a simulation, we can make a list of the neighbours of each molecule which include all particles in the sphere of radius  $r_l$ . For the next few time steps, the program only considers those particles appearing in the neighbour list for calculating the separations and selects those whose distances with particle  $i$  are smaller or equal to the potential cutoff, and then calculate potential energy and forces. This reduces the calculation of the minimum image separations. From time to time, the neighbour list is updated, and the cycle is repeated.



**Figure 4.2 Diagram showing the construction of neighbour list.**

For the update of the neighbour list, one can fix the interval at about 10 to 20 time steps at the beginning of the simulation. However a better way is allowing the program to update the neighbour list automatically. When a neighbour list is constructed, we begin to accumulate the displacements of each molecule. After some time steps, when the largest displacement exceeds the thickness of the ‘skin’,  $r_i - r_c$ , then the particles outside the neighbour sphere can go through the ‘skin’ and enter the cutoff sphere. At this time, the neighbour list should be reconstructed again. Note that not only can the particles outside the neighbour sphere go in to the cutoff sphere; the particle  $i$  itself can also move outward. So the update criterion is that the largest displacement exceeds half of the thickness of the ‘skin’.

The choice of the thickness of the cover layer depends on the system studied. That is to say, the magnitude of  $r_i$  should be appropriate. It should not be too large or too small. If it is too large, then at every time step, there are still too many particles that need to be considered. If it is too small, then the updates of the neighbour list are too frequent. In both of these two situations the neighbour list is not effective.

## 4.5 Green-Kubo relations

The Green-Kubo relations were the first method used to calculate the Navier-Stokes transport coefficients from molecular dynamics simulations [Eva84]. Through the Green-Kubo relations, one can calculate transport coefficients by observing fluctuations in equilibrium states. We can perform molecular dynamics simulations of an equilibrium system, and solve the equations of motion stepwise to follow the time evolution of the system. In the course of the simulation, thermodynamic fluxes such as the heat flux, the pressure tensor, etc., will fluctuate about their mean values. By calculating the average of appropriate time correlation functions, we can obtain all the Navier-Stokes transport coefficients.

The Green-Kubo relations are as follows for the self-diffusion coefficient, thermal conductivity, shear and bulk viscosities, respectively:

$$D = \frac{1}{3} \int_0^{\infty} dt \langle \mathbf{v}_i(t) \cdot \mathbf{v}_i(0) \rangle \quad (4.8)$$

$$\lambda = \frac{V}{3kT^2} \int_0^{\infty} dt \langle \mathbf{J}_Q(t) \cdot \mathbf{J}_Q(0) \rangle \quad (4.9)$$

$$\eta = \frac{V}{kT} \int_0^{\infty} dt \langle P_{xy}(t) P_{xy}(0) \rangle \quad (4.10)$$

$$\eta_v = \frac{1}{VkT} \int_0^{\infty} dt \langle (p(t)V(t) - \langle pV \rangle)(p(0)V(0) - \langle pV \rangle) \rangle \quad (4.11)$$

Here  $\langle \rangle$  denotes the equilibrium ensemble average.  $k$  is the Boltzmann constant.  $\mathbf{v}_i(t)$  is the velocity of particle  $i$  at time  $t$  and  $\mathbf{v}_i(0)$  is the velocity of particle  $i$  at time zero.  $V$  is the volume of system.  $\mathbf{J}_Q$  is the heat flux vector.  $P_{xy}$  is the off-diagonal element of the pressure tensor and  $\rho$  is the density.

The biggest drawback of the Green-Kubo method is that it takes a very long time. The Green-Kubo relations involve computing time correlation functions. This must then be integrated from zero to infinity. In practice, we can not integrate to infinity because of the numerical nature of these calculations. If the time correlation function decays with time rapidly, we can truncate the integration at a reasonable time. However, the Green-Kubo time correlation functions decay with time slowly. This means that a significant part of the transport coefficients is contributed by the long time tail of the correlation function, thus requiring sufficiently long times to reach convergence. The other disadvantage of the Green-Kubo relations is that they can not be used for systems that are far from equilibrium.

#### 4.6 Non-equilibrium molecular dynamics

In the above sections, we only discussed the molecular dynamics of systems at thermodynamic equilibrium. Now we introduce nonequilibrium molecular dynamics (NEMD) [Eva84, All89]. By ‘nonequilibrium’ we mean a fluid away from equilibrium and, in particular, those systems that have attained a nonequilibrium steady state. One motivation for developing nonequilibrium molecular dynamics is to improve the efficiency of calculating transport coefficients, because the Green-Kubo method is time consuming and subject to large statistical errors. Another motivation is to develop a method which can directly study systems far away from equilibrium, that is, outside the linear response regime.

Green-Kubo relations describe the response of systems to natural fluctuations within equilibrium systems. Because such a system is at thermodynamic equilibrium, on average, system properties will not fluctuate far from their mean values. Hence the signal-to-noise ratio is very small. This is undesirable for the evaluation of transport coefficients. This problem is more serious at long times, which contributes a significant portion to transport coefficients. And because of the finite size of simulation systems, there is a time limit beyond which we can not get a trustworthy time correlation functions [All89].

On the contrary, in nonequilibrium molecular dynamics a much larger fluctuation, or perturbation, is induced artificially. Thus, the signal-to-noise ratio for the system response is improved remarkably. We can measure the steady state response to such an external perturbation with high accuracy and overcome the long time tail problem.

In the early days of molecular simulation, inhomogeneous nonequilibrium molecular dynamics methods were used to calculate transport coefficients. These methods closely imitate the conditions in real experiments [Hoo75]. To maintain the system in a nonequilibrium steady state, these techniques require boundary conditions (for example sliding walls, heat reservoirs, etc.). On microscopic length scales, this leads to very strong inhomogeneities in the fluid density because the size of the simulated system is comparable with the range of the interatomic interactions. An alternative to boundary driven flow is to apply a spatially varying external field. But this also induces strong inhomogeneities. To overcome this problem, synthetic algorithms, which use artificial forces to maintain the system in a homogeneous nonequilibrium steady state, were developed.

In synthetic NEMD methods, we can use an external force field to perturb the system away from equilibrium. The force fields maintain thermodynamic fluxes or gradients in the system and prevent relaxation to the equilibrium state. To prevent this relaxation these fields need to continuously perform work on the system. This work will heat up the system. To keep a steady state we thus need to remove this heat from the system. So in steady-state NEMD we often use thermostat algorithms to remove the heat produced by synthetic fields and maintain the system at a fixed temperature. The usually used thermostat methods are the Nose-Hoover thermostat [Hoo82, Nos84a, Nos84b] and the Gaussian thermostat [Eva85]. In this thesis we have used the Gaussian thermostat.

#### 4.7 SLLOD algorithm and Lees-Edwards periodic boundary conditions

We use the thermostatted SLLOD algorithm [Eva84, Eva90] for simulating a simple one-component atomic fluid undergoing planar Couette flow. The equations of motion are:

$$\dot{\mathbf{r}}_i = \frac{\mathbf{p}_i}{m} + \mathbf{i}\dot{\gamma}_i \quad (4.12)$$

$$\dot{\mathbf{p}}_i = \mathbf{F}_i - \mathbf{i}\dot{\gamma}p_{yi} - \alpha\mathbf{p}_i \quad (4.13)$$

where  $\mathbf{p}_i$  is the ‘peculiar’ (or thermal) momentum with respect to the streaming velocity  $\mathbf{u}$  of particle  $i$ .  $\mathbf{i}$  is the unit vector in the  $x$ -direction. The thermostating term,  $\alpha\mathbf{p}_i$ , enables the system to reach a steady state by removing the dissipative heat produced irreversibly by the driving shearing force. Here we employ the Gaussian isokinetic thermostat in which the thermostat multiplier  $\alpha$  is calculated as:

$$\alpha = \frac{\sum_i (\mathbf{F}_i \cdot \mathbf{p}_i - \dot{\gamma} p_{xi} p_{yi})}{\sum_i p_i^2} \quad (4.14)$$

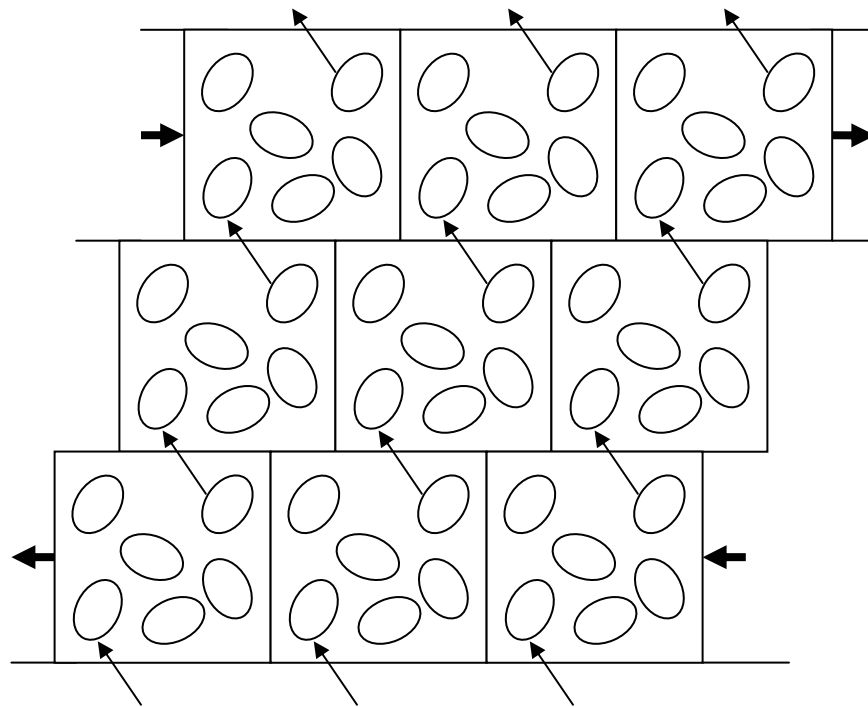
The above SLLOD equations have the beneficial property of transforming a boundary driven flow into one that is driven by a smooth mechanical force. When implemented with suitable periodic boundary conditions, such as Lees Edwards periodic boundary conditions, they generate homogeneous isothermal flow, making extraction of the shear, bulk or elongational viscosities a relatively simple task.

We should notice that in the strongly nonlinear regime or for molecules of high molecular weight, the thermostat in Equation 4.13 is not valid. When the strain rate is very strong, the assumption of a stable streaming velocity profile is incorrect. For atomic fluids one can use profile unbiased thermostats (PUT) to overcome this problem [Tra95]. In molecular fluids, it is suggested to apply a configurational temperature thermostat, because this thermostat does not make any assumption about the streaming velocity profile and it accounts for all degrees of freedom [But98, Del01].

We now introduce the Lees Edwards periodic boundary conditions [Lee72, All87, Eva90].

Figure 4.3 illustrates an infinite periodic system subjected to shear. It has the appearance of bricks sliding across one another. The central box is the simulation box. The boxes in the middle layer are stationary. Boxes in the lower layer move in the negative  $x$  direction at a velocity of  $-\dot{\gamma}L$ , where  $L$  is the box length. The boxes in the upper layer move in the positive  $x$  direction at a velocity of  $+\dot{\gamma}L$ .

In time  $t$ , the upper layer has a shift of  $+\dot{\gamma}Lt$  in the  $x$  direction relative to the middle layer. The lower layer also has the same shift in the negative  $x$  direction.



**Figure 4.3 Representation of a shearing cell under the Lees-Edwards periodic boundary condition.**

We note here that in Figure 4.3 the central box is the actual simulation box, whereas all others are periodic images of it. Thus, at time  $t$ , if a particle crosses the lower boundary at  $y = -L/2$ , it will re-emerge through the  $y = +L/2$  boundary with a shift of  $+\dot{\gamma}L t$  in the  $x$  direction, and a streaming velocity  $\dot{\gamma}L$  will be added to it. Note that the peculiar velocities do not change. Similarly, when a particle crosses the upper boundary at  $y = +L/2$ , it is

transformed through  $y = -L/2$  with a shift of  $\dot{\gamma}L t$  in the negative  $x$  direction, and loses a streaming velocity of  $\dot{\gamma}L$ .

We should note that when the upper layer has a displacement of  $+\dot{\gamma}L t$  larger than a box length, for example,  $nL < \dot{\gamma}L t < (n + 1)L$ ,  $n$  is an integer, then the particle reappearing at  $+L/2$  will get a shift which is not  $+\dot{\gamma}L t$ , but which is this value minus a whole number multiple of box lengths  $nL$ . For the lower layer, the principle is the same.

The other aspects of periodic boundary conditions (eg, particles moving out of the left/right boundaries) are the same as in equilibrium. However, we note that when we use the “minimum image convention”, we need to account for the shift in the  $x$  direction when particles interact with image particles.

#### 4.8 Transient time correlation function and mapping

Mode-coupling theory predicts asymptotic behaviour in the limit of zero field. To check the stress-strain relation, we need to calculate those quantities in very weak fields. We know that direct NEMD cannot be used in the weak field limit because in this situation the signal to noise ratio is very low. The transient time correlation function (TTCF) method, on the other hand, can be used in the weak field limit [Eva90]. TTCF is a nonlinear response theory. It generalizes the Green-Kubo relations to the full nonlinear regime in a simple way.

In the TTCF formalism, an arbitrary phase variable  $B(t)$  can be expressed in terms of a nonequilibrium transient time correlation function [Eva90]

$$\langle B(t) \rangle = \langle B(0) \rangle - \beta F_e \int_0^t ds \langle B(s) J(0) \rangle, \quad (4.15)$$

where  $\beta = 1/kT$ , and  $J(0)$  is the dissipative flux at  $t = 0$ , which is itself related to the adiabatic rate of change of the internal energy due to the field. Here  $T$  is the temperature of

the system. The angular brackets indicate time averages. This formula assumes that the applied external field  $F_e$  is time independent. It is important to notice that the above equation describes the correlation of a nonequilibrium phase variable  $B(t)$  at some time  $t$ , with a dissipative flux value in equilibrium at the starting time.

For planar Couette flow  $J$  is the negative of the shear stress  $P_{xy}$  times the system volume  $V$  [Eva88]

$$\begin{aligned} (\dot{H}_0)^{ad} &= -JF_e = -\dot{\gamma} \sum_i \left[ \frac{p_{xi} p_{yi}}{m} + y_i F_{xi} \right] \\ &= -\dot{\gamma} P_{xy} V. \end{aligned} \quad (4.16)$$

where  $H_0$  is the system internal energy.  $F_{xi}$  is the  $x$  direction component of force exerted on particle  $i$  by other particles.

If a canonical ensemble (or isothermal ensemble) of systems is suddenly subject to a constant shearing deformation at  $t = 0$ , then, the formula (4.15) becomes:

$$\langle B(t) \rangle = \langle B(0) \rangle - \beta \dot{\gamma} V \int_0^t ds \langle B(s) P_{xy}(0) \rangle \quad (4.17)$$

To calculate the transient time correlation function, it is necessary to first generate a field free equilibrium molecular dynamics trajectory. At fixed time step  $N_e$  intervals along this trajectory, we take the phase state (coordinates and momenta) as starting points of new ‘daughter’ NEMD trajectories. In this way we generate many starting points. From each starting point we can add a constant shear field to the system and run a nonequilibrium trajectory of  $N_n$  time steps. We can run many nonequilibrium trajectories and take averages over these trajectories. The equilibrium time steps  $N_e$  should be long enough to ensure that the starting points are not correlated. Likewise, the nonequilibrium time steps

$N_n$  should be long enough too, to ensure that the system relaxes to a nonequilibrium steady state.

Further-more, we can use ‘phase space mapping’ to improve the efficiency of the calculation of transient time correlation functions. The use of phase space mapping has two advantages: (a) it can generate a group of starting phases from one phase state; (b) it can reduce statistical uncertainty.

From one initial starting phase  $\Gamma = (x, y, z, p_x, p_y, p_z)$ , we can obtain four starting phases by using the mappings  $M^I$ ,  $M^T$ ,  $M^Y$  and  $M^K$  [Eva90];

$$\Gamma^I = M^I[\Gamma] = (x, y, z, p_x, p_y, p_z) \quad (4.18)$$

$$\Gamma^T = M^T[\Gamma] = (x, y, z, -p_x, -p_y, -p_z) \quad (4.19)$$

$$\Gamma^Y = M^Y[\Gamma] = (x, -y, z, p_x, -p_y, p_z) \quad (4.20)$$

$$\Gamma^K = M^K[\Gamma] = (x, -y, z, -p_x, p_y, -p_z) \quad (4.21)$$

For planar Couette flow these four phases give four different starting states, and lead to four different nonequilibrium trajectories from the single equilibrium phase point  $\Gamma$ . Each of these phase states appears with the same probability because the Hamiltonian  $H_0$  is not changed under these mappings.

Phase space mapping has another function. Consider Eqn. (4.17). The correlation kernel function is  $\langle B(t)P_{xy}(0) \rangle$ . At long time, the phase variable  $B$  will not correlate to the shear stress value at the starting time, and  $\langle B(t)P_{xy}(0) \rangle$  becomes  $\langle B(t) \rangle \langle P_{xy}(0) \rangle$ . If we choose equilibrium starting states  $\Gamma$  in such a way that  $\langle P_{xy}(0) \rangle \equiv 0$ , we could minimize the statistical errors in calculating the transient correlation integral. This procedure minimizes statistical error because numerical integration of Eqn. (4.17) results in error accumulation at long times. We can accomplish this by using the phase space mapping in

pairs. In the above four mappings,  $\Gamma^I$ ,  $\Gamma^T$  have the same shear stress  $P_{xy}$ , and  $\Gamma^Y$ ,  $\Gamma^K$  have the shear stress  $-P_{xy}$ . Their difference is a negative symbol, but the magnitude is the same. If we choose the mappings in pairs such that one has positive shear stress and the other has negative shear stress at time zero, we can guarantee that  $\langle P_{xy}(0) \rangle \equiv 0$ . In our work, we choose  $\Gamma^I$  and  $\Gamma^Y$  mappings.