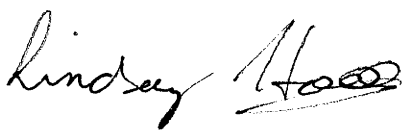
Computer Simulation of Shear Flow in Simple Fluids

Lindsay Malcolm Hood

June, 1989

A thesis submitted for the degree of Doctor of Philosophy of the Australian National University.

Except where stated, the work described in this thesis is my own original work.



Lindsay Hood

ACKNOWLEDGEMENTS

I would like to thank Denis Evans and Gary Morriss for their supervision of the work described here.

I also wish to thank Peter Kusalik for discussions on numerical methods.

Much of the computation described here was performed on the Fujitsu VP100 of the Australian National University Supercomputer Facility.

Abstract

Computer simulation of shear flow in fluids has become an important technique in our understanding of the non-linear behaviour of fluids subjected to an external field. An important method for studying fluids from the atomic or molecular level is Molecular Dynamics. Conventional molecular dynamics simulations of shear flow have been performed at constant shear rate and constant pressure. In this thesis we describe how to perform simulations at constant shear stress or constant pressure, which are the usual **experimental** conditions. The theory used in deriving the constant stress equations is quite general and can be applied for fields other than shear flow, allowing one to simulate at constant thermodynamic force or flux. The constant pressure simulations show clearly the difference between shear thinning and shear dilatancy, a point that has caused confusion in the rheological literature.

Efficient computational methods are important as some simulations can take hundreds of hours of computer time. The arrival of a vector processor at the A.N.U. has necessitated the experimentation with algorithms, with substantial performance gains over a standard code. These increases in performance have allowed us to complete a thorough study of shear flow at constant pressure for the soft sphere system, which complements a large body of data at constant density from other workers. Also a thorough study of shear flow in two dimensions has been completed. The two dimensional system shows a large dependence on system size, and we have been able to show the transition from small system behaviour to large system behaviour.

Table of Contents

List of Figures

List of Tables

1. Introduction

Statistical Mechanics forms a bridge between the macroscopically observable description of a system provided by thermodynamics and hydrodynamics, and the microscopic description of the constituent particles of the system provided by mechanics. This thesis will describe calculations using a method called Molecular Dynamics to examine the rheological behaviour of fluids undergoing shear flow. Molecular dynamics allows us to make predictions based on the properties of model systems, such as the soft sphere and Lennard-Jones potentials. In this chapter we discuss some statistical mechanics formalism necessary in deriving correct algorithms for molecular dynamics simulations. We also give an introduction to simple rheological properties that are of interest from such simulations.

1.1 Classical Statistical Mechanics

Throughout this thesis, and indeed in much statistical mechanics research, classical mechanics is used to describe the motion of the constituent particles. Although numerical quantum statistical mechanics has just begun to be studied, for the properties in which we are interested it is found that the classical approximation is quite adequate. Also the mathematical and computational complexities of quantum statistical mechanics make it a method to resort to only if necessary.

One of the goals of statistical mechanics is to calculate thermodynamic and transport properties of a system based on a knowledge of molecular properties. The macroscopic description of a system consists of a few variables such as mass, temperature, volume, etc. However the microscopic description of a system consists of many more variables, the spatial coordinates and momenta of each particle, of which there are $O(10^{23})$ in a macroscopic sample. The connection between the two descriptions is made by calculating the value of a particular mechanical quantity in each of the large number of states that is consistent with the macroscopic description of the system. It is then postulated that this mechanical property corresponds to some thermodynamic or transport quantity. An ensemble is a collection of systems that are defined by the same macroscopic properties but have different microscopic configurations. Different ensembles have different sets of state-defining macroscopic variables. Thus the microcanonical ensemble has N, the number of particles, V, the volume of the system and E, the total energy of the system fixed. A more commonly used ensemble is the canonical ensemble where N, V and T, the temperature are specified. An ensemble average is the average of a property calculated over all members of the ensemble at a particular time t. A time average is the average of a property calculated in one member of the ensemble followed over a length of time. The ergodic hypothesis is that the time average of an equilibrium or of a steady state system is the same as the ensemble average. If this is the case then the system is called ergodic. The concept of ensembles and their use in statistical mechanics is due to Gibbs [1].

1.2 Phase Space

Consider a classical system of N particles. In three dimensions each particle requires 6 numbers to specify its position, 3 spatial coordinates q_x, q_y, q_z and 3 momentum coordinates p_x, p_y, p_z. These 6N coordinates, along with the equations of motion completely specify the behaviour of the system for all time t. We assume that the particles are all the same chemical species and therefore mass and interaction potential. The 6N dimensional vector Γ, consisting of **q** and **p** for all particles is a point in the phase space for the system. The dynamics of the system evolving in time is described by the trajectory of the phase point through phase space. The trajectory is given by Hamilton's equations, where H is the Hamiltonian of the system

$$\dot{\mathbf{q}}_j = \frac{\partial H}{\partial \mathbf{p}_j} \quad , \quad \dot{\mathbf{p}}_j = -\frac{\partial H}{\partial \mathbf{q}_j} \tag{1.1}$$

We introduce a distribution function which describes the probability of the system being at a particular point in phase space at a particular point in time, $f(\mathbf{p},\mathbf{q},t)$. The distribution function f also describes the distribution of states in the ensemble. The ensemble average of a function $B(\mathbf{p},\mathbf{q})$ is given by

$$\langle B \rangle = \frac{\int \ldots \int B(\mathbf{p},\mathbf{q})\, f(\mathbf{p},\mathbf{q},t)\, d\mathbf{p}d\mathbf{q}}{\int \ldots \int f(\mathbf{p},\mathbf{q},t)\, d\mathbf{p}\, d\mathbf{q}} \tag{1.2}$$

This is more usually written

$$\langle B \rangle = \frac{\int d\Gamma\, B(\Gamma)\, f(\Gamma,t)}{\int d\Gamma\, f(\Gamma,t)} \tag{1.3}$$

Thus it can be seen that statistical mechanics links the exact description of a system provided by classical mechanics to the probabilistic description of the bulk system as provided by thermodynamics and hydrodynamics and measured in experiments.

The time dependence of f is given by the Liouville equation

$$\frac{df}{dt} = -\sum_i \left(\frac{\partial H}{\partial p_i} \frac{\partial f}{\partial q_i} - \frac{\partial H}{\partial q_i} \frac{\partial f}{\partial p_i} \right) \tag{1.4}$$

The Liouville equation is the most fundamental equation of classical statistical mechanics. We now introduce non-equilibrium statistical mechanics where the Liouville equation has to be generalized.

1.3 Non-Equilibrium Statistical Mechanics

Thermodynamics and statistical mechanics were originally formulated for systems at equilibrium. Although equilibrium systems are important, the world around us is clearly full of non-equilibrium systems, and it would be useful to have a theoretical understanding of such. Linear non-equilibrium thermodynamics is valid only in the regime close to equilibrium where deviations from equilibrium are linear in the applied external field such as a temperature or concentration gradient. Although this is useful we would like a description of systems arbitrarily far from equilibrium. In particular we are interested in non-linear transport processes.

The Liouville equation can be written in a more general form

$$\frac{\partial f(\Gamma,t)}{\partial t} = -\frac{\partial}{\partial \Gamma} \cdot \dot{\Gamma} f(\Gamma,t)$$

$$= -iL f(\Gamma,t) \qquad (1.5)$$

The operator L is called the f-Liouvillean because it describes the advance of $f(\Gamma,t)$ from t = 0. Equation (1.5) can be solved for $f(\Gamma,t)$ as a function of $f(\Gamma,0)$

$$f(\Gamma,t) = e^{-iL\,t} f(\Gamma,0) \qquad (1.6)$$

We have considered the advance of the distribution function in time, however we can also consider the time dependence of a phase variable as we follow a phase point along its trajectory,

$$\frac{dB}{dt} = \dot{\Gamma} \cdot \frac{\partial B}{\partial \Gamma} \equiv iLB(\Gamma) \qquad (1.7)$$

The operator L is called the p-Liouvillean, and similarly the formal expression for the description of a phase variable at time t is

$$B(\Gamma(t)) = e^{iLt} B(\Gamma(0)) \tag{1.8}$$

The two Liouville operators are Hermitian adjoints of each other,

$$\int d\Gamma\, B(-iL)\, f = \int d\Gamma\, B\left(\frac{-\partial}{\partial\Gamma} \cdot (\dot{\Gamma} f)\right)$$

$$= \int d\Gamma\, f\, \dot{\Gamma} \cdot \frac{\partial B}{\partial \Gamma}$$

$$= \int d\Gamma\, f\, iL\, B \tag{1.9}$$

If the system is described by a Hamiltonian then $\partial/\partial\Gamma \cdot d\Gamma/dt = 0$, and equation (1.5) reduces to the usual form of the Liouville equation. The expression $\partial/\partial\Gamma \cdot d\Gamma/dt$ is called the phase space compression factor, Λ. The existence of a Hamiltonian is a sufficient but not necessary condition for $\Lambda = 0$. In this case the f- and p-Liouvilleans are the same. However since we will later wish to deal with thermostatted systems in order to achieve steady states, where Λ is not zero, we must use equation (1.5) and distinct f- and p-Liouvilleans.

1.4 Non-Linear Response Theory [2]

The essential difference between an equilibrium and a non-equilibrium system is that in the latter an external field such as an electric field or thermal gradient is acting on the system. When the field is switched on the system will exhibit a transient response and may then relax to a non-equilibrium steady state. If the external field is dissipative then the system will heat up, and a thermostat will be necessary for a steady state to be reached. In this section we present a discussion on the expression for phase averages as they evolve in time under the influence of an external field. The non-linear response theory described here is a generalization of linear response theory. The famous Green-Kubo relations are derived from linear response theory [3]. It is assumed that the field is constant once switched on; the extension to time-dependent fields is exceedingly complex [4]. It is also assumed that the system is thermostatted, although a detailed description of thermostatting is left until chapter 2.

At time t = 0 the distribution function is given by the equilibrium isokinetic distribution

$$f(\Gamma,0) = f_k(\Gamma) = \frac{\delta(K-K_0)\, e^{-\beta\phi}}{\int d\Gamma\, \delta(K-K_0)\, e^{-\beta\phi}} \tag{1.10}$$

where K_0 is the kinetic energy corresponding to the temperature defined by $\beta = 1/k_B T$. The delta function selects only those members of the canonical ensemble that have the desired kinetic energy. The external field is switched on at time t = 0, and changes the dynamics to

$$\dot{q}_i = \frac{p_i}{m} + C_i F_e$$

$$\dot{p}_i = F_i + D_i F_e - \alpha p_i \tag{1.11}$$

in the most general case. F_e is the magnitude of the external field, and $\mathbf{C}_i$ and $\mathbf{D}_i$ describe the coupling of the field to the system. The term $\alpha \mathbf{p}_i$ is the thermostat. Although mechanisms for thermostatting will be discussed later, the theory described here is based on the Gaussian isokinetic thermostat, where α is defined by

$$\alpha = \frac{\sum_i \mathbf{F}_i \cdot \mathbf{p}_i + F_e \sum_i \mathbf{D}_i \cdot \mathbf{p}_i}{\sum_i p_i^2} \qquad (1.12)$$

In these equations (1.11) and (1.12), $\mathbf{p}$ is the peculiar momentum, ie the momentum relative to the local streaming velocity. How the streaming velocity is defined will be discussed in chapter 2, but for the time being it is sufficient to assume that it can be defined.

We assume that the external field is conservative, so that in the absence of a thermostat, Liouville's theorem holds. This assumption is known as the adiabatic incompressibility of phase space, or AIΓ. It is not necessary to derive the theory with this assumption, but it is simpler and has proved sufficient for most systems. We compute the phase space compressibility factor Λ from

$$\Lambda = \frac{\partial}{\partial \Gamma} \cdot \dot{\Gamma} = -3N\alpha + O(1) \qquad (1.13)$$

As the second term is intensive it can be ignored in the thermodynamic limit. At equilibrium $\langle\alpha\rangle$ is zero, and so equation (1.13) reduces correctly to the adiabatic case. This further shows that the thermostat leaves equilibrium correlation functions unchanged. The evolution of $f(\Gamma_0,t)$ in time from $t = 0$ can be computed using a result known as Morriss' Lemma [5].

$$\begin{aligned} f(\Gamma_0,t) &= e^{-iLt} f_k(\Gamma_0) \\ &= \exp\left(-\int_0^t 3N\alpha(\Gamma(-s))\, ds \right) \delta(K-K_0)\, e^{-\beta\phi(\Gamma(-t))} \end{aligned} \qquad (1.14)$$

The potential energy ϕ of the system can be calculated from

$$\phi(\Gamma(-t)) = \phi(\Gamma_0) - \int_0^t \dot{\phi}(\Gamma(-s))\, ds \tag{1.15}$$

$d\phi/dt$ is related simply to the adiabatic heating since the rate of change of kinetic energy is zero (the system is thermostatted).

$$\begin{aligned} \dot{\phi} &= \dot{H} \\ &= \dot{H}^{ad} - \alpha \sum p_i^2/m \\ &= \dot{H}^{ad} - 3N\alpha\beta^{-1} \end{aligned} \tag{1.16}$$

dH^{ad}/dt is the heat flux dissipated in the system due to the external field F_e. The dissipative flux J is defined in general by

$$J \equiv -\dot{H}^{ad}/F_e \tag{1.17}$$

In this case the dissipative flux is defined by

$$3N\alpha + \beta\dot{\phi} = -J F_e \tag{1.18}$$

If we substitute the above results into equation (1.14) and rearrange we obtain the Kawasaki Distribution Function [6]

$$f(\Gamma,t) = \frac{f_k(\Gamma)\exp\left(-\beta\int_0^t J(\Gamma(-s))F_e ds\right)}{\int d\Gamma\, f_k(\Gamma)\exp\left(-\beta\int_0^t J(\Gamma(-s))F_e ds\right)} \tag{1.19}$$

This function describes the phase space distribution of a system that has evolved under the

influence of a dissipative external field F_e and has been thermostatted to guarantee the existence of a steady state. We can evaluate phase variable averages of this distribution from

$$\langle B(t) \rangle = \frac{\left\langle B(\Gamma_0) \exp\left(-\beta \int_0^t f(\Gamma(-s))\, F_e\, ds \right) \right\rangle_K}{\left\langle \exp\left(-\beta \int_0^t J(\Gamma(-s))\, F_e\, ds \right) \right\rangle_K} \qquad (1.20)$$

The subscript K refers to the fact that the averages are computed using the isokinetic equations of motion over the isokinetic ensemble. The Kawasaki distribution function above is the explicitly normalized form. Calculations of properties using the Kawasaki distribution are extremely noisy due to the evaluation of the exponential of the integral, a quantity which rapidly gets very small since it is extensive. From the Kawasaki distribution can be derived a much more efficient method of calculating phase averages, the Transient Time Correlation Function, first derived by Evans and Morriss in 1987 [7]. The TTCF is related to the Kawasaki distribution in the following way.

The average of a phase variable B(t) can be expressed in terms of a correlation function by differentiating equation (1.20). This gives

$$\frac{d\langle B(t) \rangle}{dt} = -\beta F_e \langle B(t)\, J(0) \rangle_K + \beta F_e \langle B(t) \rangle_K \langle J(0) \rangle_K \qquad (1.21)$$

The average dissipative flux at t = 0 is zero (the field is switched on at t = 0, before that the system is at equilibrium), so we get an expression for the average of a phase variable

$$\langle B(t) \rangle_K = \langle B(0) \rangle_K - \beta F_e \int_0^t ds \langle B(s) J(0) \rangle_K \qquad (1.22)$$

This relation is exact for all values of the external field, F_e. This expression is formally

quite similar to that for linear response theory, with one crucial difference. In linear response theory the correlation function is an equilibrium or Green-Kubo one, ie it is evaluated with the field off, whereas for non-linear response theory the correlation function is evaluated with the field on, and "remembers" back to when the field was first switched on. Hence the correlation function includes the transient response as well as the long time steady state response, which usually decays to zero.

It is easy to see that in the linear regime the transient correlation function response reduces to the linear Green-Kubo expression. Thus at small fields the efficiency of the TTCF formalism should be the same as for equilibrium Green-Kubo calculations.

1.5 Rheology

Rheology as discussed in this thesis is the study of fluids under shear. The shearing "experiment" used in calculations described in this thesis is planar Couette flow. In planar Couette flow two plates with the fluid between them move in opposite directions, causing the fluid to shear. Other types of shearing experiments such as circular Couette flow can be considered, but are more difficult to simulate on the computer. The velocity difference between the two walls divided by the perpendicular separation is the strain rate or shear rate, γ. This is a simple non-equilibrium system which permits one to investigate a wide range of phenomena in fluids. Shear flow has technological significance. The pumping of crude oil down a pipe requires careful control otherwise the oil can solidify. As another example the lubricating oil in an engine is undergoing extreme shear flow in the pistons, bearings and other friction surfaces. In internal combustion engines shear rates can be as high as 10^7 Hz, hydrostatic pressures as high as 4 gigapascals, the shear stress up to 300 megapascals and the viscous heat dissipation as high as 10^{15} W/m^3.

The pressure of a fluid is not a simple scalar quantity, but is a second rank tensor, defined by the equation

$$dF = -dP^T.dA \tag{1.23}$$

where A is an area element moving with the streaming velocity of the fluid and F is the force exerted on the area element. This definition of the pressure leads to the following microscopic expression for the pressure

$$PV = \frac{1}{m}\sum \mathbf{p}_i\mathbf{p}_i + \sum \mathbf{q}_i\mathbf{F}_i \tag{1.24}$$

from which it can be seen that the pressure has a kinetic and a configurational component. The hydrostatic pressure is defined in three dimensions by

$$p = \frac{1}{3} \mathrm{tr}\, \mathbf{P} \tag{1.25}$$

At equilibrium the pressure tensor for a fluid is isotropic with $P_{xx} = P_{yy} = P_{zz} = p$. Shear flow causes the off-diagonal elements of the pressure tensor to be non-zero; for steady shear in the bulk fluid in the x-y plane $P_{xy} = P_{yx}$. The shear stress is the negative of P_{xy}. A fluid cannot sustain a shear stress without flowing, a solid however can sustain a suitably small shear stress indefinitely.

A Newtonian fluid is one in which the shear viscosity defined by

$$\eta = - P_{xy} / \dot{\gamma} \tag{1.25}$$

is independent of the shear rate and the diagonal elements of the pressure tensor have their equilibrium values. For a non-Newtonian fluid this is not the case. It is found in practice that all fluids exhibit non-Newtonian behaviour for at least some range of shear rates. It is found that at constant volume the pressure increases with increasing shear rate; for a system at constant pressure, there is a decrease in density with increasing shear rate; this effect is known as shear dilatancy. For some fluids the shear viscosity increases with increasing shear rate, these fluids are described as shear-thickening, the converse behaviour being shear-thinning. It is entirely possible for a fluid to display both types of behaviour depending on the range of shear rates being studied.

1.6 Molecular Dynamics Simulations

Although statistical mechanics theories can be expressed and solved in a very formal or analytic way, many of the advances of the past few years in our understanding of fluids have come from simulations of systems of particles on computers. One of the attractive features of simulations is that in principle they can be made arbitrarily exact, the limitations being due only to numerical rather than physical approximations. The theory outlined earlier has been oriented towards a simulation technique called Molecular Dynamics (MD), first introduced at Los Alomos and Lawrence Livermore Labs in the U.S. in the 1950's [8]. MD has become a very powerful tool for statistical mechanics because of its ability to produce information about transport and other time-related processes, something which Monte Carlo simulations cannot do. For this reason it has become the simulation method of choice for many workers.

In principle MD is very simple; one simply solves the differential equations of motion numerically on a computer and calculates the time average of the desired properties. The equations of motion are integrated for a sufficient length of time until the statistical fluctuations in the calculated mean values of properties have been reduced to a satisfactory level. This may take many hundreds of hours of computer time. Although simple in principle there are several points that need to be expanded upon.

The first of these is the numerical integration method. Today there are many good differential equation solvers. MD was originally performed using the Verlet or leapfrog algorithm. Runge-Kutta methods are self-starting but require 2 or 4 force evaluations per timestep. In MD the force evaluation is the most time-consuming part of the calculation, and so a Runge-Kutta method is used only if necessary. The simulations described in this thesis were all performed using the Gear predictor-corrector algorithm. The Gear method is very accurate but requires a relatively small timestep for stability. It provides very accurate trajectories when compared with the Verlet algorithm. A major advantage of the Gear

algorithm is that it allows one to solve coupled first order differential equations, which is the preferred way of writing the equations of motion. The Verlet algorithm can only be used with a second order differential equation.

The capacity of today's computers limits the number of particles that can be studied to about 100 000 or so but typically much smaller systems of 100 to 1000 particles are used. Such small systems have very large surface to volume ratios; so to observe bulk properties the system is surrounded by periodic images of itself, extending to infinity. This is shown in figure 1.1. Although these periodic boundary conditions impose a periodic structure on the system, for most purposes the length scale is large enough so that there is little effect. However one must still be careful that the boundary conditions or the system size are not influencing the results. Chapter 6 describes some calculations where system size is important.

The force calculation is the most time-consuming part of an MD calculation. If we assume a pairwise-additive potential then we have to evaluate the force between every pair of particles, where the force is defined by

$$\mathbf{F}_{ij} = -\nabla_i \phi_{ij} \quad \text{where} \quad \nabla_i \equiv \frac{\partial}{\partial q_i} \tag{1.27}$$

Clearly for large systems this becomes prohibitively expensive. A simplifying assumption that is usually made is to truncate the potential beyond a certain cutoff distance, r_c. Any pair of atoms further apart than this makes no contribution to the potential energy of the system nor to the force on each other. To eliminate jump discontinuities in the energy it is usual to shift the potential so that it is zero at the cutoff. This however makes the force discontinuous at the cutoff. The force between two particles i and j is calculated between i and the nearest image of j (which may be j itself). This means that interactions are not restricted to occurring just in the primitive cell. This is shown in figure 1.2. It is also possible to consider three-body and higher contributions to the forces, but this is usually not done, as these terms typically contribute little to the structure and dynamics of the

system. For accurate comparisons with real experimental data three body corrections are usually required for a dense system. Methods for improving the computation of the forces are described in chapter 3.

For the liquid noble gases, the simple Lennard-Jones potential gives remarkably accurate thermodynamic and transport data

$$\phi_{LJ} = 4\varepsilon\left((\sigma/r)^{12} - (\sigma/r)^{6}\right) \tag{1.28}$$

The potential is scaled by ε the well-depth, σ where the potential is zero and m the particle mass. The well is located at $2^{1/6}\sigma$. By choosing ε,σ and m appropriately we can **scale** the simulation results to calculate the properties of all of the noble gases from the one simulation. This leads to the important concept in a simulation of reduced quantities; all quantities in the system are scaled or reduced by ε,σ and m. For proper scaling during a simulation all calculations are performed using reduced quantities with $\varepsilon = \sigma = m = 1$. Reduced quantities are usually denoted by a *, but because reduced quantities are always used the superscript is often dropped. Similar scaling also applies to other potentials used in molecular dynamics, including the soft sphere potential

$$\phi_{SS} = 4\varepsilon\,(\sigma/r)^{12} \tag{1.29}$$

which is purely repulsive. For qualitative understanding of phenomena the choice of potential makes little difference; the qualitative behaviour of a system is determined to a large extent by geometric considerations.

1.7 Non-Equilibrium Simulations

Non-equilibrium molecular dynamics (NEMD) simulations require more careful design than do equilibrium ones. The aim is to construct an homogeneous algorithm. This is sometimes difficult because the physical experiment we are modeling may be very inhomogeneous. For example, consider thermal conductivity. The physical experiment consists of a hot wall and a cold wall, and a corresponding thermal gradient. Were we to model such a system exactly on the computer the small system sizes that can be used with today's computers would lead to very large edge effects and other such problems because the ratio of volume to surface area is so small. A much more efficient approach is to impose a *synthetic* external field which couples with the equations of motion to drive the system in the desired manner. Such an approach transforms a boundary condition driven problem to one with an external field and allows us to use the response theory developed earlier. It may still be necessary however to modify the periodic boundary conditions to accommodate the external field. This is the case with simulations of planar Couette flow which are discussed in the next chapter. An excellent review on the design of NEMD algorithms is given by Evans and Morriss [9].

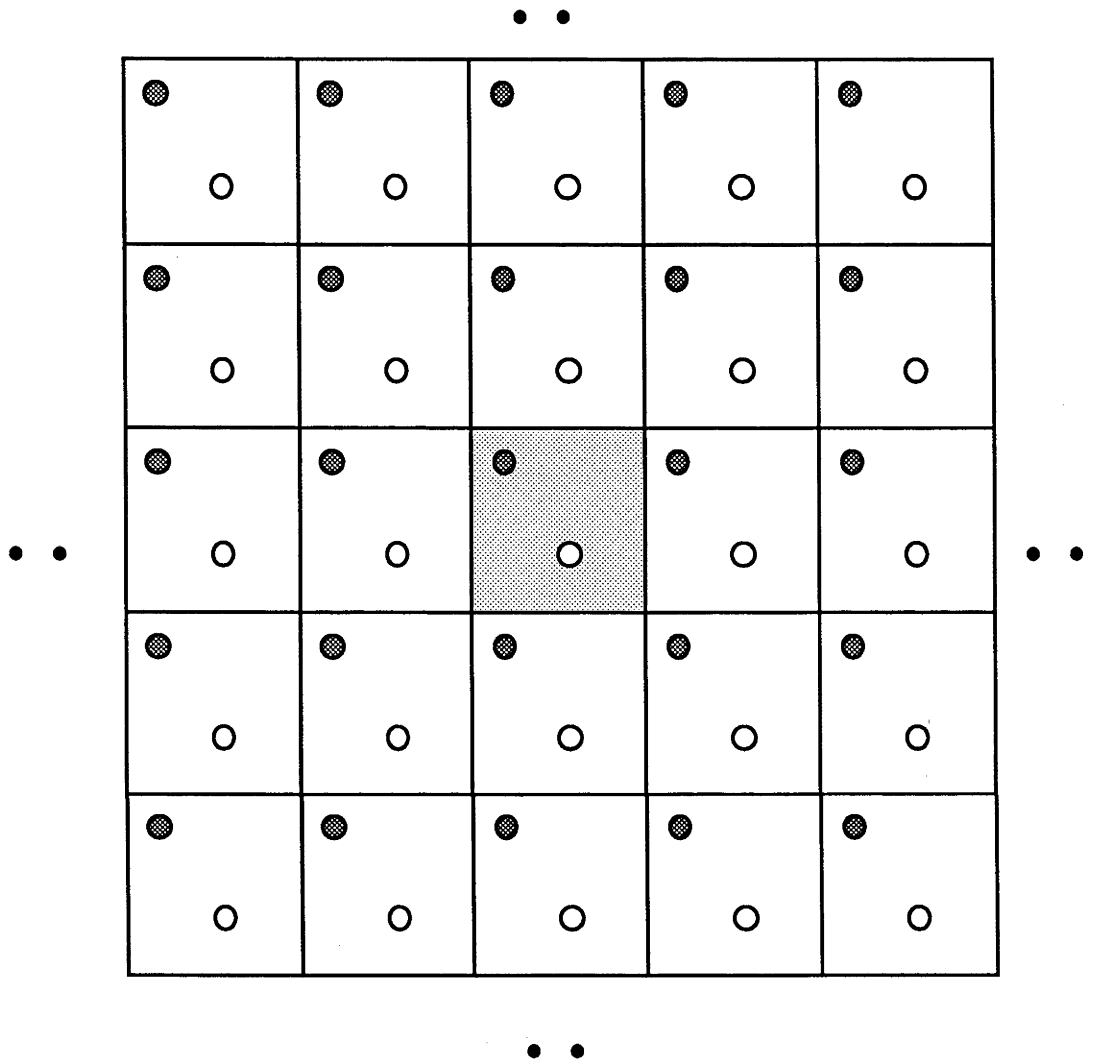


Figure 1.1 Periodic boundary conditions for molecular dynamics simulations. The primitive cell (shown stippled) is replicated to infinity. Only two images in each direction are shown.

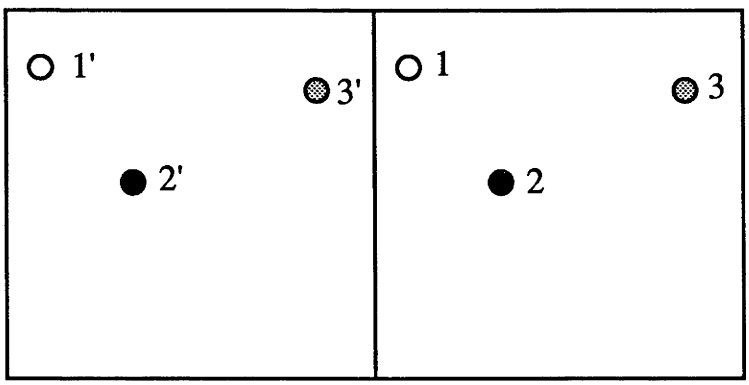


Figure 1.2 The primitive cell is shown with an image cell to its left. The force between particles 1 and 2 is calculated using the positions of particles 1 and 2, but using 1 and 3' for the force between particle 1 and 3.

2. NEMD of Planar Couette Flow

This chapter describes in detail the application of NEMD to the particular case of planar Couette or simple shear flow. A detailed description of thermostatting mechanisms is also given.

2.1 Equations of Motion for Planar Couette Flow

Several algorithms have been proposed for simulating shear flow, but it has been shown that the Sllod equations of motion [10] combined with Lees-Edwards boundary conditions [11] give an exact description of planar Couette flow, arbitrarily far from equilibrium. The Lees-Edwards boundary conditions are also called, quite descriptively, sliding brick boundary conditions. In these boundary conditions the periodic images are translated horizontally from one layer to the one below it. As time progresses, the images are translated further. When the images have been translated the length of the primitive cell the images line up orthogonally again. Because the primitive cell and its images do not change shape there is no volume change to account for in the simulation. These boundary conditions are shown in figure 2.1.

The Sllod equations of motion are

$$\dot{\mathbf{q}}_i = \frac{\mathbf{p}_i}{m} + \gamma \, \mathbf{i} \, q_{yi}$$

$$\dot{\mathbf{p}}_i = \mathbf{F}_i - \gamma \, \mathbf{i} \, p_{yi} \qquad (2.1)$$

γ is the strain rate or shear rate, $\partial u_x/\partial y$. We assume that the shear rate is constant once switched on. Differentiating the dq/dt equation gives

$$\ddot{\mathbf{q}}_i = \dot{\mathbf{p}}_i/m + \gamma \, \mathbf{i} \, \dot{q}_{yi} + \dot{\gamma} \, \mathbf{i} \, q_{yi}$$

$$= \frac{\mathbf{F}_i}{m} - \frac{\gamma \, \mathbf{i} \, p_{yi}}{m} + \gamma \, \mathbf{i} \, \dot{q}_{yi} + \dot{\gamma} \, \mathbf{i} \, q_{yi}$$

$$= \frac{\mathbf{F}_i}{m} + \dot{\gamma} \, \mathbf{i} \, q_{yi} \qquad (2.2)$$

Thus except at t = 0 when the shear rate is switched on, the Sllod equations of motion are simply Newton's. Once the shear motion has impulsively started at t = 0, the Lees-Edwards boundary conditions continue the shear flow motion. The advantage of the Sllod equations is that they transform a boundary driven system to one with an external field, which makes the system more amenable to theoretical analysis. The equations of motion as currently written will not let a steady state be reached. The system will continue to heat up due to the work performed on it by the external field. To achieve a steady state we need to use a feedback mechanism to extract the heat.

The Reynolds number for shear flow is defined by $R_e = \rho\gamma L^2/\eta$. For low Reynolds number the streaming velocity at a point (x,y) is $\mathbf{i}\,\gamma y$. Thus from Sllod we see that the $\mathbf{p}_i$ are peculiar momenta measured relative to the local streaming velocity.

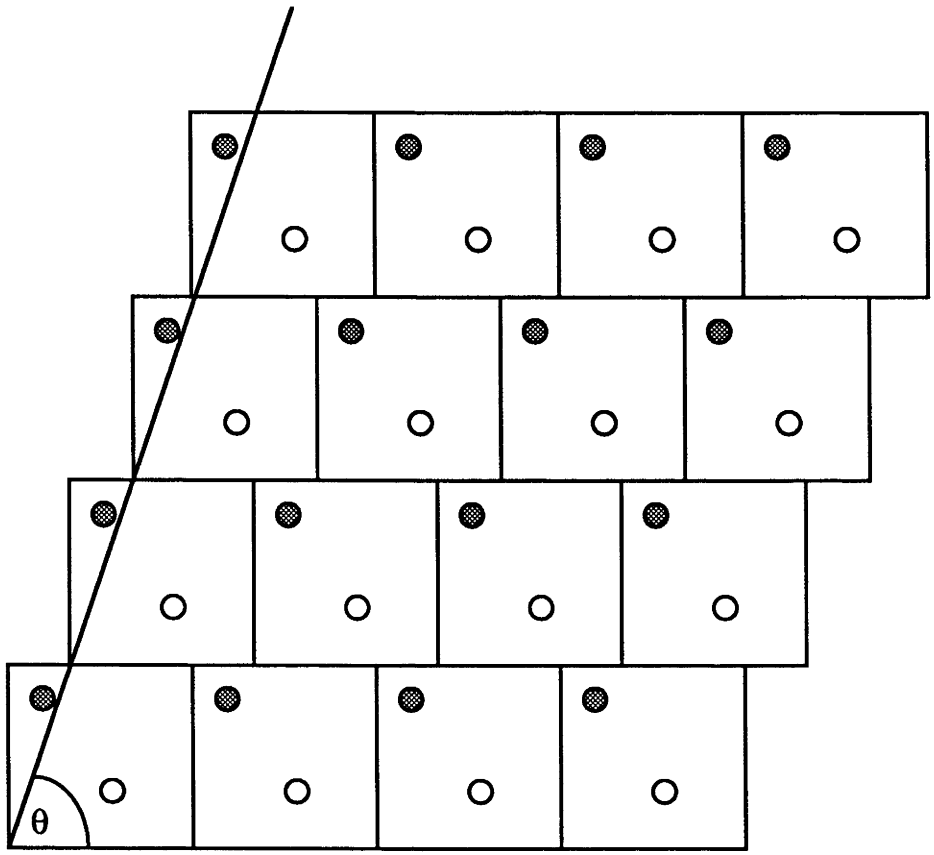


Figure 2.1 The Lees-Edwards boundary conditions used for planar Couette flow simulations. For the purposes of the simulation when $\theta = 45^{\circ}$ the cells line up orthogonally and θ can be reset to 0.

2.2 Feedback Mechanisms

The use of a feedback mechanism allows one to constrain some property of the system. In this case we wish to constrain the kinetic energy which we postulate is related to the ideal gas temperature of the system. Ignoring the precise details for the moment of how the temperature is defined, we describe two mechanisms which have been used to control the temperature of NEMD simulations, namely Gauss's Principle of Least Constraint and the Nosé-Hoover feedback mechanism. A third thermostatting mechanism due to Berendsen [12] is not discussed because of its complexities, the most serious of which is that it is not time reversible.

2.3 Gauss's Principle of Least Constraint

Over 150 years ago Gauss proposed a system of mechanics that was more general than Newton's. This mechanics is based on Gauss's principle of least constraint, a principle he called the most fundamental dynamical principle. The constraint can be written as

$$g(\mathbf{q},\dot{\mathbf{q}},t) = 0 \tag{2.3}$$

If g does not depend explicitly on dq/dt then the constraint is termed holonomic, otherwise non-holonomic. The application of this principle in NEMD simulations is to fix thermodynamic or hydrodynamic quantities so that a non-equilibrium steady state is achieved. To show how Gauss's principle works we shall derive the equations of motion for isokinetic shear flow.

The constraint function can be differentiated with respect to time to give an equation of the form

$$\mathbf{n}(\mathbf{q},\dot{\mathbf{q}},t) \cdot \ddot{\mathbf{q}} = s(\mathbf{q},\dot{\mathbf{q}},t) \tag{2.4}$$

This equation is the equation of a hyperplane called the constraint plane in 3N-dimensional acceleration space. To satisfy the constraint we must project the unconstrained accelerations back onto the constraint plane. Gauss's principle states that the actual trajectories followed deviate minimally in a least squares sense from the unconstrained trajectories. The projection which minimizes the magnitude of the constraint force is the normal projection of the unconstrained acceleration. The Gaussian equations of motion then are

$$\ddot{\mathbf{q}} = \frac{\mathbf{F}}{m} - \lambda \mathbf{n} \tag{2.5}$$

λ is a Lagrange multiplier that must be determined at every timestep. If we substitute

equation 2.5 into 2.4 and rearrange we obtain

$$\lambda = (\mathbf{n}.\mathbf{F}/m - s)/n^2 \tag{2.6}$$

We wish the ideal gas or kinetic temperature to be a constant of the motion. In three dimensions the temperature is defined by

$$\sum_i \frac{p_i^2}{2m} - \frac{3}{2}Nk_BT = 0 \tag{2.7}$$

To be strictly correct the factor 3N should be 3N-4, as four degrees of freedom are lost, 3 because momentum is zero in x, y and z directions and one more because the kinetic energy is constant. In practice it is usual to ignore the correction term because 3N is much larger. However for very small systems the correction factor becomes important. $\mathbf{p}_i$ is the peculiar momentum, measured relative to the local streaming velocity. This is the constraint equation. Differentiating with respect to time gives

$$\sum_i \mathbf{p}_i \cdot \dot{\mathbf{p}}_i = 0 \tag{2.8}$$

It is easy to apply Gauss's principle to the Sllod equations of motion. Remembering that the Sllod $\mathbf{p}_i$ are peculiar, the constant temperature equations of motion are

$$\dot{\mathbf{q}}_i = \frac{\mathbf{p}_i}{m} + \gamma \mathbf{i}\, q_{yi}$$

$$\dot{\mathbf{p}}_i = \mathbf{F}_i - \gamma \mathbf{i}\, p_{yi} - \alpha\, \mathbf{p}_i \tag{2.9}$$

To solve for α we simply substitute the equations of motion into the differentiated form of the constraint equation, equation (2.8). This gives

$$\sum_i \mathbf{p}_i \cdot \dot{\mathbf{p}}_i = 0$$

$$\Rightarrow \sum_i \mathbf{p}_i \cdot (\mathbf{F}_i - \gamma i\, p_{yi} - \alpha p_i^2) = 0$$

$$\Rightarrow \sum_i \mathbf{p}_i \cdot \mathbf{F}_i - \gamma \sum_i p_{xi} \cdot p_{yi} - \alpha \sum_i p_i^2 = 0$$

$$\Rightarrow \alpha = \frac{\sum_i \mathbf{p}_i \cdot \mathbf{F}_i - \gamma \sum_i p_{xi} \cdot p_{yi}}{\sum_i p_i^2} \qquad (2.10)$$

The application of Gauss's principle to constrain other properties follows similar lines.

The above constraint serves to keep the kinetic temperature constant. At equilibrium no net work is done by the thermostat, but away from equilibrium the thermostat will extract the viscous heat generated by the external field. It should be strongly stressed that at no stage has the actual **value** of the peculiar kinetic energy been constrained. We are merely ensuring that its time derivative should be zero. The Gaussian constraint is thus a differential one. The Gaussian isokinetic equations of motion were first proposed independently by Hoover and by Evans [13]. The equations were initially not derived from Gauss's principle, but were instead derived empirically. It was not until some time later that it was realized that the equations of motion could be derived from Gauss's principle and hence put the method on a firm theoretical foundation.

2.4 Nosé-Hoover Feedback Mechanism

In contrast to the differential constraint of the last section, the Nosé-Hoover feedback mechanism is an example of an integral constraint. The Berendsen thermostat [12], not discussed further, is an example of a proportional constraint. The Nosé-Hoover mechanism is a very general one that was first proposed by Nosé [14] to thermostat equilibrium systems. The original formulation by Nosé involved a cumbersome external reservoir and a non-linear time transformation. Hoover made significant simplifications to the application of the method [15], which has since become known as the Nosé-Hoover method. Almost all applications of the method use the simplified Hoover formulation.

In the Nosé-Hoover method the constrained property is not constrained rigidly like the Gaussian method, but instead fluctuates about the target value. The constraint works by relating the feedback variable to the difference between the current value of the property and its target value. For use as a thermostat, the dq/dt and dp/dt equations remain the same, but the value of α is instead obtained from integrating the equation

$$\dot{\alpha} = \frac{1}{\tau^2}\left(\frac{K}{K_0} - 1\right) \tag{2.11}$$

Typically the Gear predictor-corrector algorithm is used to integrate this equation. K_0 is the desired or target kinetic energy and K is the instantaneous kinetic energy. Where the target value is zero (impossible for temperature but not for other properties) an alternative form can be used

$$\dot{\alpha} = \frac{1}{Q}(K - K_0) \tag{2.12}$$

This form of the constraint equation is used in the work described here. The constraint can work on intensive or extensive quantities, although working with intensive variables throughout is simpler.

Although the Nosé-Hoover mechanism looks appealing due to its simplicity, it has one glaring weakness. The $1/\tau^2$ or $1/Q$ factor alters the rate at which excursions from the target value are damped. If $1/Q$ is too large, ie Q is too small, the equations of motion will be stiff and the system will exhibit ringing. On the other hand, if Q is too large the system will damp excursions from the target value too slowly; in the limit of infinite Q the system will not respond at all. In the case of thermostats, infinite Q turns the system into an adiabatic one. The problem is that currently there is no prescription for choosing a precise value of Q; one has to use trial and error to find an appropriate value. Provided the extremes of Q being too large or too small are avoided, the average values obtained for thermodynamic quantities do not depend on Q. An example of the variation of results with Q will be shown in chapter 4.

The principal advantages of the Nosé-Hoover method are that it results in simple expressions for the constraint multiplier and that it is self-correcting. Numerical inaccuracies result in a slight drift of properties constrained using Gaussian dynamics, and therefore require periodic rescaling to the correct value is required. However the final choice as to which method is used often depends on efficiency. For thermostatting systems with the standard kinetic definition of temperature the Gaussian constraint is substantially more efficient than the Nosé-Hoover method [15].

2.5 Temperature Definition

In the previous sections we stated that the kinetic temperature depended on the peculiar momenta. We have neglected one important point however; what is the streaming velocity of particle i at time t ? The answer to such a question is not as simple as it might at first seem. Take the example of planar Couette flow. An infinite linear velocity profile is imposed on the system, but there is no requirement that the streaming velocity be $u_x = \gamma y$ for all y. At the origin $u_x = 0$ and at the top of the primitive cell $u_x = \gamma L$, but there is no requirement that the velocity profile be linear within these two points. It is known from hydrodynamics that the linearity of the streaming velocity is controlled by the Reynolds number, R_e, defined earlier. For small shear rates, and therefore R_e, the profile is linear, but for large shear rates it is likely that the profile develops some secondary structure, such as an S-shaped kink.

The earliest use of thermostats by Evans and Hoover did assume that the expected streaming velocity at a point y was $u_x(y) = \gamma y$. Such thermostats are referred to as profile biased thermostats (PBT). In 1984 and 1985 various PBT simulations for large (896 particle) 2 dimensional systems were performed at very high shear rates [16], where R_e was greater than 10^3. The assumption of a stable linear velocity profile under such conditions is extremely dubious. What was observed for large shear rates was a so-called string phase where particles lined up in layers parallel to the flow direction, and formed "strings" in the perpendicular direction. Such systems were highly non-ergodic, with properties very much dependent on the preparation history of the system. Also observed was the co-existence of a string and an amorphous phase. Such co-existence is unlikely because the two phases must be described by different state points, in particular the shear viscosities must be different. The assumption of a stable linear velocity profile was later shown to stabilize such co-existence [17].

Evans and Morriss introduced a Profile Unbiased Temperature (PUT) thermostat [17] that does not make any assumptions about the expected shape of the velocity profile; in fact it is not even required that a stable velocity profile exist. The local streaming velocity about a given particle is simply defined to be the average velocity of all particles in a small region about that central particle. For a suitably small region the difference in y-coordinate of the surrounding particles will not cause them to have significantly different imposed shear rates. Thus this average gives a good description of the local streaming velocity. For small shear rates it was found that the PUT thermostat gave identical results to the standard Gaussian, profile biased thermostat. For larger shear rates, typically $\gamma \sim\geq 2$, the two thermostats differed markedly in their results. The PUT thermostatted system exhibited large shear thickening at high field, whereas the string phase was markedly shear thinning, with a shear viscosity proportional to the number of strings that had formed in the system. Snapshots of the system showed the PUT system to be more homogeneous and isotropic than the string phase. The merit of having an unbiased definition of the temperature had been shown, but there were substantial gaps in the understanding of the implementation of the thermostat that needed investigation.

2.6 Implementations of the PUT Thermostat

The PUT thermostat as described earlier was implemented simply by doing a periodic rescaling of the velocities. This proved to be unsatisfactory for two reasons. Firstly, there was no formal theory behind the rescaling, in particular the dissipation of the system could not be checked. Secondly, it did not hold the temperature well at large fields; although a steady state was achieved, the final temperature was not the one wanted. Thus a more efficient implementation of the PUT concept was required.

The standard case we shall use for the development of the PUT thermostat is N atoms in 2 dimensions. For the PBT Gaussian thermostat the temperature is defined to be

$$T^* = \frac{1}{N-3}\frac{1}{2m}\sum_i p_i^2 \tag{2.13}$$

The factor N-3 comes in from three degrees of freedom having been removed, two because total momentum in x and y is zero and one because the total kinetic energy is fixed. In practice the factor 1/(N-3) is usually simplified to 1/N. This leads to the constraint equation

$$\sum_i \mathbf{p}_i \cdot \dot{\mathbf{p}}_i = 0 \tag{2.14}$$

From this we get the expression for the multiplier obtained earlier

$$\alpha = \frac{\sum_i \mathbf{p}_i \cdot \mathbf{F}_i - \gamma \sum_i p_{xi}\, p_{yi}}{\sum_i p_i^2} \tag{2.15}$$

We can define the temperature of a small volume of space, a cell, to be

$$T_c^* = \frac{1}{n_c - 1}\frac{1}{2m}\sum_{i \in c}(\mathbf{p}_i - \frac{1}{n_c}\mathbf{p}_c)^2 \qquad (2.16)$$

where $\mathbf{p}_c = \Sigma\, \mathbf{p}_i$ for all particles in the cell and n_c is the number of particles in the cell. If n_c is 0 or 1 then that cell has no temperature according to the definition here. Here it is important to include the correction factor in the counting of the degrees of freedom because n_c is small. The factor is n_c - 1 because there is no requirement for the momentum of the cell to be zero. Providing the size of the cell is sufficiently small this is a good definition of the kinetic temperature relative to the local streaming velocity. The problem now is to define the temperature of the whole system based on these individual cell temperatures. The general temperature definition is

$$T_{cell}^* = \frac{\sum_c w_c T_c^*}{\sum_c w_c} \qquad (2.17)$$

where w_c is a weight function. It was found that the following weight function counted the degrees of freedom correctly

$$w_c = \begin{cases} 2(n_c - 1)\,m & n_c \geq 2 \\ 0 & n_c = 0,1 \end{cases} \qquad (2.18)$$

Thus the temperature definition becomes

$$T^* = \frac{\sum_c \sum_{i \in c} (\mathbf{p}_i - \frac{1}{n_c}\mathbf{p}_c)^2}{\sum_c 2(n_c - 1)\,m} \qquad (2.19)$$

The constraint equation arising from this is

$$\sum_c \sum_{i \in c} (\mathbf{p}_i - \frac{1}{n_c}\mathbf{p}_c) \cdot (\dot{\mathbf{p}}_i - \frac{1}{n_c}\dot{\mathbf{p}}_c) = 0 \qquad (2.20)$$

where it has been assumed that dn_c/dt is sufficiently small and can be ignored. We can evaluate the rate of change of the cell momenta,

$$\dot{\mathbf{p}}_c = \sum_{i \in c} \dot{\mathbf{p}}_i$$

$$= \sum_{i \in c} (\mathbf{F}_i - \gamma \mathbf{i} p_{yi})$$

$$= \mathbf{F}_c - \gamma \mathbf{i} p_{yc} \qquad (2.21)$$

The equation for dp/dt becomes

$$\dot{\mathbf{p}}_i = \mathbf{F}_i - \gamma \mathbf{i} p_{yi} - \alpha(\mathbf{p}_i - \frac{1}{n_c} \mathbf{p}_c) \qquad (2.22)$$

Substituting (2.18) and (2.19) into (2.17) and rearranging gives the expression for the cell-based PUT thermostat Gaussian isokinetic multiplier

$$\alpha = \frac{\sum_c \left(\sum_{i \in c} (\mathbf{p}_i \cdot \mathbf{F}_i - \gamma p_{xi} p_{yi}) - \frac{1}{n_c} (\mathbf{p}_c \cdot \mathbf{F}_c - \gamma p_{xc} p_{yc}) \right)}{\sum_c \sum_{i \in c} (\mathbf{p}_i - \frac{1}{n_c} \mathbf{p}_c)^2} \qquad (2.23)$$

The above definitions and expression were derived by Morriss in 1986 [18].

Although the above scheme was better than the velocity rescaling scheme used earlier, it was still not ideal. In particular, the number of particles in a given cell is not constant in time, despite the assumption made in deriving the expression for α. Obviously by going to larger cells the movement of particles from one cell to another will be reduced, and the jump discontinuities will be relatively smaller. However, if the cell is too large, then the above expressions do not give an accurate description of the *local* streaming velocity. Also, the difference in streaming velocity from one layer to the next will be substantial causing the

"temperature" of the particle to change significantly and lead to problems in integrating the equations of motion.

One of the aims of this thesis was to investigate thermostatting mechanisms more fully. We performed some experiments on two system sizes, N = 896 and N = 3584 particles, to see the effects of cell size on the the temperature constancy and the stability of the equations of motion. It was found that the optimal size of for the cells was one in which on average there were a little over two particles per cell, ie the cell was about as small as possible for the definitions to make sense. Although larger cells made for smoother distributions for the number of particles within the cells, the difference in streaming velocity from one layer to the next was too large for the integrator to handle without having to reduce the timestep. Coincidentally, this cell size arises from the use of a cell algorithm (to be described in chapter 3) with a cutoff of 1.5σ, and was the cell size used in the early work by Evans and Morriss [17] although their thermostat used a simple velocity rescaling.

2.7 Cell Velocity Interpolation

The jump discontinuities in streaming velocity between layers could be avoided by using an interpolation from one layer to the next. The problem with this though is the top and bottom layers. Due to the Lees-Edwards boundary conditions there is no requirement for the cells on the bottom of the primitive cell to line up in registry with the cells in the top layer of the periodic image below. It is thus hard to define which cells to interpolate with which across an image boundary. This problem could be avoided by supposing that the cells in a particular layer all have the same streaming velocity, ie the flow is laminar. But this is an assumption on the expected streaming velocity profile, something that we are trying to avoid. Also, to be strictly accurate with our interpolation, we would need to interpolate in both x and y directions in two dimensions, and in the z direction for three dimensions. Because of these difficulties it was decided that the current cell-based PUT thermostat whether Gaussian or velocity rescaling could not be simply and efficiently improved, so it was decided that a different approach was needed.

2.8 The Entropy Connection

One of the problems with the above thermostats is the difference between the thermodynamic definition of temperature and the kinetic one. Recent work by Evans [19] shows that the kinetic temperature is the thermodynamic temperature at equilibrium, but not away from equilibrium. Because temperature is actually defined in terms of the entropy, the whole question of thermostatting non-equilibrium systems may well have to wait until we have a thorough understanding of the entropy of non-equilibrium systems.

3. Numerical Methods

For most molecular dynamics programs the evaluation of the interparticle forces is by far the most expensive part of the calculation. Most efforts at improving program performance should be directed at improving the efficiency of calculating the interparticle forces. In this chapter we describe some algorithms designed to do this. Also we discuss an important type of computer called a vector processor and its impact on scientific computation. It is likely that within the next few years all scientific and technical computation will be done using either vector or parallel processing techniques [20].

3.1 Vector Processors

In the past 10 years a very different type of computer called a vector processor has dominated large-scale scientific and technical computation. The vector processor differs substantially from a standard or scalar processor by processing vectors of data in an arithmetic pipeline with a single instruction. The pipeline at any one stage has several operands in various stages of completion. There is a startup time from when the instruction is issued to when the first result is delivered, after that a result is delivered per machine cycle. Although there is no requirement for this, vector processors have tended to be at the very high end of the market; only in the past few years has vector processing technology been implemented in VAX class machines. Currently the fastest available computers are multiprocessor vector processors with peak computational speeds of 1-2 Gflops, with a sustained rate 2 to 10 times less than this.

The efficient use of a vector processor requires substantially more careful coding than does a scalar processor. This is because of the enormous speed difference between the vector unit and the scalar unit of the computer, and the conditions under which vector processing of operands can occur. To achieve high computational speeds on a vector processor the following aims should be met :

i) Long vectors of operands should be processed to minimize the effect of the startup time.

ii) The selection of data to be processed and the processing of that data should be as distinct as far as is possible. This allows the arithmetic pipes to be kept busy processing data rather than idling while waiting for data to arrive.

iii) The computation of one element of the vector does not depend on the computation of an earlier element which may still be in the pipeline; ie there is no recursive reference or recurrence relation. This is because the vector is processed as a single entity, one could *imagine* that the elements are processed in parallel. In fact a vector processor is logically

equivalent to a single instruction multiple data (SIMD) parallel processor.

Simple MD programs vectorize extremely well and it is not hard to achieve 99% vectorization for such programs [21]. However the simple code is not always the fastest. For large systems the simple approach is extremely inefficient, even though its vectorization is extremely high. These systems require substantially more complicated code and are hence far more difficult to vectorize. In the following sections a number of algorithms to improve the performance of MD codes are described, together with comments on memory requirements, performance and vectorization characteristics.

3.2 The Brute Force Approach

Brute force methods vectorize extremely well, show spectacular vector to scalar speedups, but are hopeless for all but the smallest systems !! The code is a simple nested pair of loops over all pairs of particles :

```
DO I = 1, N
  DO J = 1, N
    CALCULATE R_IJ
    CALCULATE FORCE F_IJ AND SUM IN TO FORCE ARRAYS
    ...
  ENDDO
ENDDO
```

Simple use of Newtons Law of action and reaction allows one to calculate over the triangle of interactions rather than the whole square. This results in a speed increase of about 2. Both these methods are $O(N^2)$ and so the method is inefficient for large systems with small cutoffs. The method is inefficient becaus large numbers of pairs of particles do not interact with each other, and we have to evaluate the distance between them before this can be determined. The aim of the methods to be described subsequently is to efficiently determine the near neighbours for each particle. This is a problem inherent not just to MD but to a wide variety of computational disciplines.

3.3 Cell Codes

A very simple but very efficient method (for scalar processors) for finding the near neighbours is to grid the primitive cell into cells such that the near neighbours are found in the central or home cell and the immediately surrounding ones. The cell size is chosen so as to be bigger than the cutoff distance, r_c. This is shown in figure 3.1.

Two arrays are required for accounting purposes. The first of these is a counter of the number of particles in each cell labeled by indices ix and iy (for two dimensions). The second array is a list of the particles in each cell. This second array has three dimensions, 2 for the cell indices and one for the particle index within that cell. The array bound for this dimension has to be at least as large as the maximum number of particles found in any cell, something which has to be determined by trial and error. Because not all cells will have the same number of particles there is some wastage of memory with this method, especially for low density systems where $<\Delta n^2>$ is large.

The extension to shearing boundary conditions is surprisingly simple, but rather subtle. There is no difference for cells in the bulk of the fluid, but the cells along the top and bottom layer require special consideration. A particle in a cell in the top layer can find its neighbours in 4 cells from the bottom layer of the next periodic image. The physical picture is shown in figure 3.2 and the code for finding the neighbouring particles in two dimensions under shearing boundary conditions is shown in figure 3.3.

One of the better known algorithms in computational science is Hockney's link cell algorithm. The algorithm is very efficient in its use of memory at the expense of some more complicated code. The algorithm is not described here, instead the reader is referred to Hockney's original work [22]. Code for a three dimensional non-shearing system is shown in figure 3.4.

3.4 Performance of Cell Codes

Although Hockney's link cell code is very tight in its use of memory, it is found that with today's computers the extra memory required by the standard cell code is of little concern. For dense systems the density fluctuations are quite small, thus there is only a small amount of wasted space due to the array bounds described earlier. We will be using this property of dense systems in another algorithm to be described. A far more serious drawback for the two algorithms is that they are essentially scalar [22,23] and thus do not make efficient use of the vector hardware. The performance of the two algorithms is similar on a scalar machine, although the standard cell code is somewhat faster.

An efficient vectorized cell code would greatly improve computational efficiency in many fields [23]. Such a scheme called the Monotonic Lagrangian Grid has been proposed by Boris [23,24]. In the MLG algorithm the nearest n neighbours in each direction on a grid are scanned. There is no way of precisely defining over what distance neighbours are searched for. This causes problems for MD where we need to know the cutoff distance for the potential. It is not clear that the method will become useful in MD calculations, although work continues in the area.

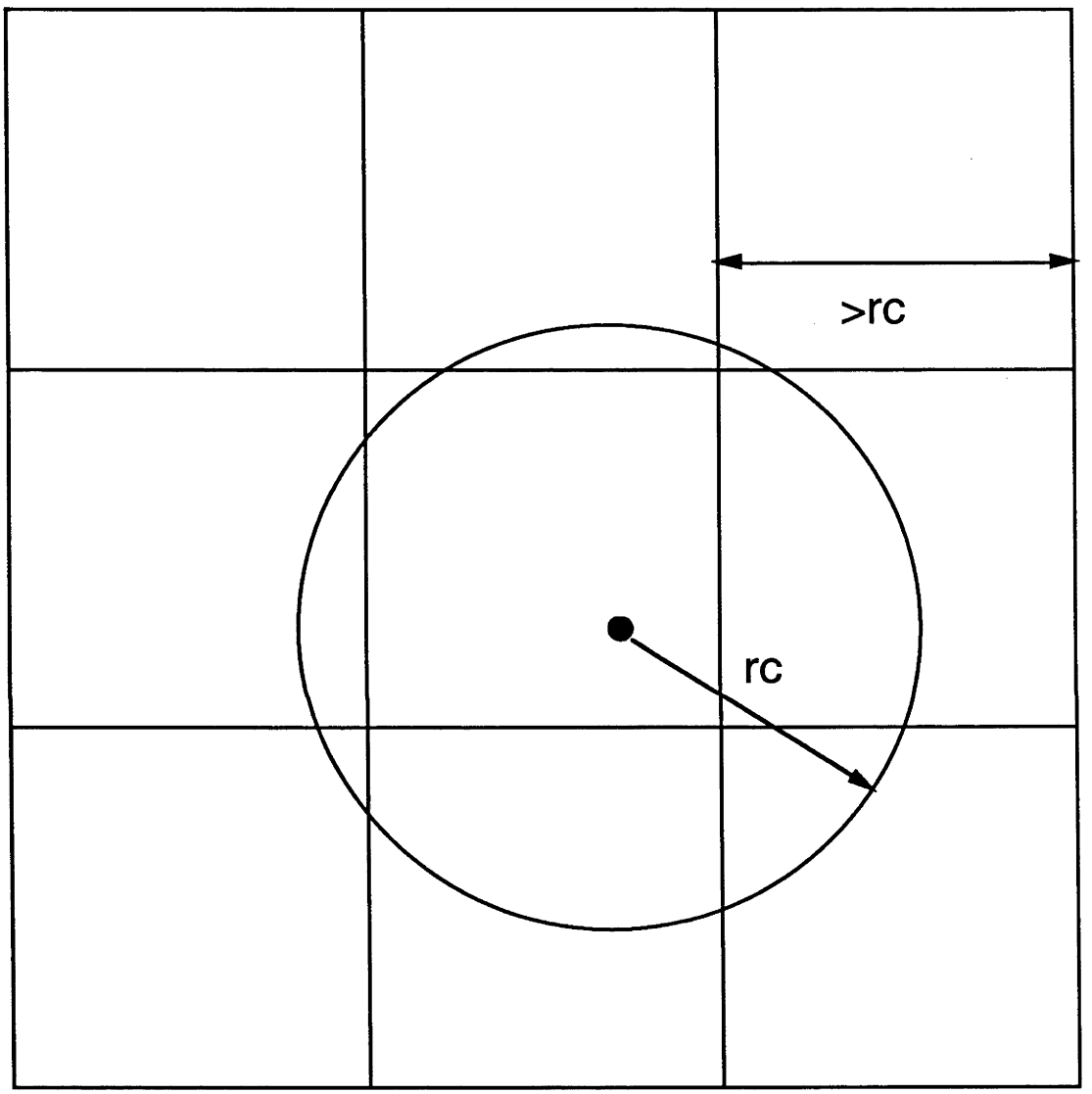


Figure 3.1 Shows the relation between cell size and the cutoff circle. In general the neighbours for each particle will be found in the central cell plus the eight surrounding ones. By using Newton's Law the only neighbouring cells that have to be searched are the cell immediately to the left plus the three across the top.

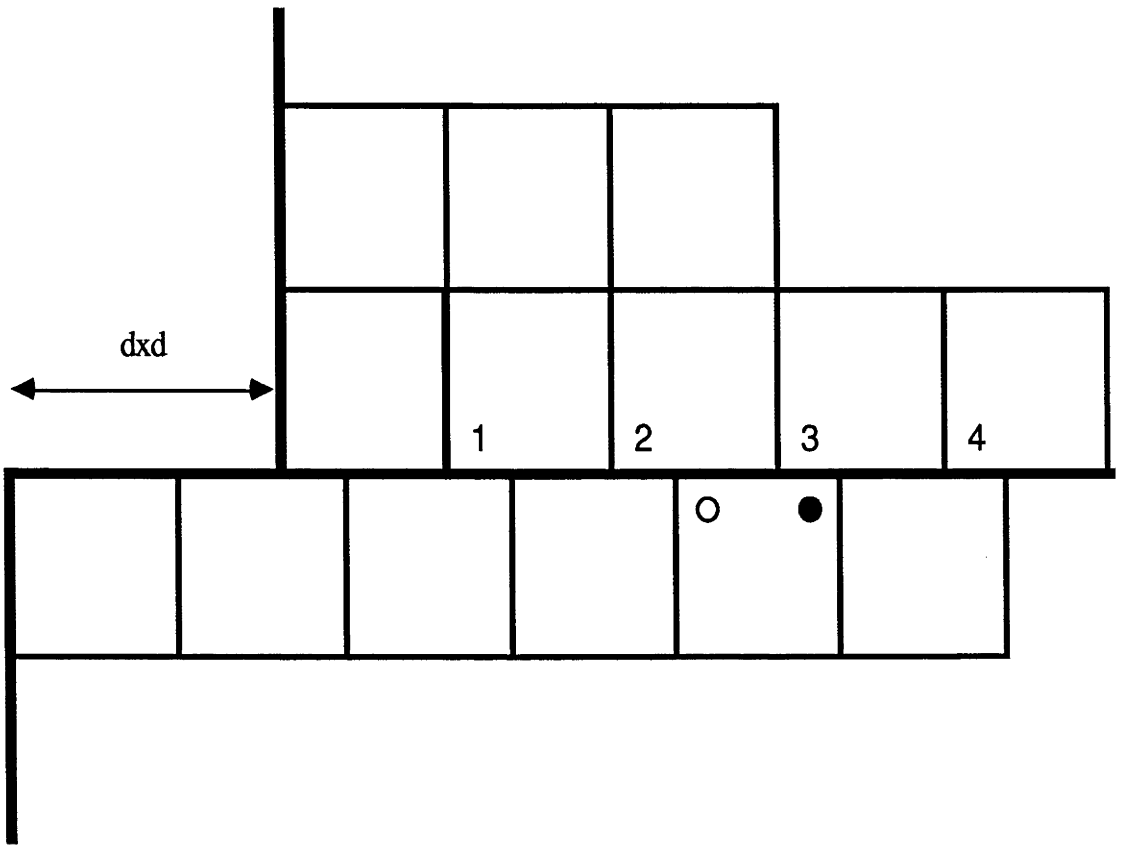


Figure 3.2 Cells for shearing boundary conditions. The hollow circle will interact with particles in cells 1, 2 and 3, while the filled circle will interact with particles in cells 2, 3 and 4. In general we have to search 4 cells. dxd is the fraction that the image cells have been displaced, $\Delta x/Lx$.

```
      NCM1 = 2*NCELL - 1
      DO 55 IX = 1, NCELL
        DO 50 IY = 1, NCELL
C ------- NO PARTICLES IN THIS CELL SO SKIP THE LOOP
          IF (JN(IX,IY) .EQ. 0) GOTO 50
C ------- HOME CELL
          DO 25 NI = 1, JN(IX,IY)-1
            I = ICELL(NI,IX,IY)
            DO 20 NJ = NI+1, JN(IX,IY)
              J = ICELL(NJ,IX,IY)

C ***           EVALUATE INTERACTION BETWEEN PARTICLES I AND J

   20         CONTINUE
   25       CONTINUE
C ------- LOOK OVER CELLS ABOVE
          DO 45 ICY = 0, 1
            IIY = MOD(IY + ICY + NCM1, NCELL) + 1
            IF (ICY .EQ. 0) THEN
C ----------- ONLY LOOK AT ONE CELL TO THE LEFT WHEN ON SAME LAYER
              ILOWER =   1
            ELSE
              ILOWER = - 1
            ENDIF
            IF (IY + ICY .GT. NCELL) THEN
C ----------- TOP LAYER SO LOOK ONE MORE CELL TO THE LEFT
              ILOWER = - 2
            ENDIF
C --------- LOOK OVER CELLS TO THE LEFT
            DO 40 ICX = ILOWER, 1
              IIX = MOD(IX + ICX + NCM1, NCELL) + 1
C ----------- ARE THERE ANY PARTICLES IN CELL (IIX,IIY)
              IF (JN(IIX,IIY) .EQ. 0) GOTO 35
              DO 35 IP = 1, JN(IX,IY)
                I = ICELL(IP,IX,IY)
                DO 30 JP = 1, JN(IIX,IIY)
                  J = ICELL(JP,IIX,IIY)

C ***             EVALUATE INTERACTION BETWEEN PARTICLES I AND J

   30           CONTINUE
   35         CONTINUE
   40       CONTINUE
   45     CONTINUE
   50   CONTINUE
   55 CONTINUE
```

Figure 3.3 Standard two-dimensional cell code for shearing boundary conditions. The actual evaluation of the forces and pressure tensor is not shown.

```
      SUBROUTINE FORCE
      PARAMETER (NP=16384,NC=25,N0=8,NK=4,NM=1,
     & NVECT=1000,NDOUB=2000)
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)
      INTEGER TITLE(20)
      DIMENSION FIJX(NVECT),FIJY(NVECT),FIJZ(NVECT),
     & RXX(NVECT),RYY(NVECT),RZZ(NVECT),JADDR(NVECT),LTOP(NDOUB),
     & LINK(NP),NIX(14),NIY(14),NIZ(14),JJADR(NVECT),LAST(NP)
      DATA NIX/ 0,-1,-1,-1, 0, 0,-1, 1,-1, 0, 1,-1, 0, 1/
      DATA NIY/ 0, 0,-1, 1, 1, 0, 0, 0,-1,-1,-1, 1, 1, 1/
      DATA NIZ/ 0, 0, 0, 0, 0, 1, 1, 1, 1, 1, 1, 1, 1, 1/
C
      NLX = CUBE/RCUT
      NLX1= NLX - 1
      FNLX=FLOAT(NLX)
      NCELLS = NLX**3
      DO 1000 L = 1,NCELLS
 1000 LTOP(L)  = 0
C
C LTOP WILL CONTAIN THE NAME OF THE FIRST PARTICLE FOR EACH LINK CELL
      DO 2000 I = 1,NP
      IX = INT(FNLX*X(I)*CUBEI)
      IY = INT(FNLX*Y(I)*CUBEI)
      IZ = INT(FNLX*Z(I)*CUBEI)
      ICELL       = 1 + IX + NLX*(IY + NLX*IZ)
      J           = LTOP(ICELL)
      LTOP(ICELL) = I
      LINK(I)     = J
 2000 CONTINUE
C
C        FIJ = FORCE ON I DUE TO J.
C        PRIMARY LOOP OVER ALL CELLS
C
C
      DO 5001 IZ = 0,NLX1
      DO 5001 IY = 0,NLX1
      DO 5001 IX = 0,NLX1
      IC = 1 + IX + NLX*(IY + NLX*IZ)
C        IC DENOTES THE CURRENT HOME CELL
      I = LTOP(IC)
C
C        BYPASS IF HOME CELL IS EMPTY
      IF(I.EQ.0) GOTO 5001
      M = 0
   99 M = M+1
      JADDR(M) = I
C CONTIGUOUS ADDRESSING ARRAY
      RXX(M)    = X(I)
      RYY(M)    = Y(I)
      RZZ(M)    = Z(I)
      I         = LINK(I)
      IF(I.GT.0) GOTO 99
      MSTART    = M
```

Figure 3.4 Hockney's cell code for a three dimensional system. The Forces routines is shown in its entirety.

```
C THERE ARE MSTART HOME-CELL ATOMS.  POSITIONS STORED IN FIRST MSTART
C ELEMENTS OF RXX,RYY,RZZ
C
C SECONDARY LOOP OVER NEIGHBOURING CELLS
      DO 4001 KC = 2,14
      SX       = 0.0
      SY       = 0.0
      SZ       = 0.0
      JX       = IX + NIX(KC)
      JY       = IY + NIY(KC)
      JZ       = IZ + NIZ(KC)
C
C PERIODIC BOUNDARY CONDITIONS
      IF((IX.EQ.NLX1).AND.(JX.GT.IX)) THEN
      JX       = 0
      SX       = CUBE
      ELSE
         IF((IX.EQ.0).AND.(JX.LT.IX)) THEN
       JX       = NLX1
       SX       =-CUBE
       ENDIF
      ENDIF
      IF((IY.EQ.NLX1).AND.(JY.GT.IY)) THEN
      JY       = 0
      SY       = CUBE
      ELSE
         IF((IY.EQ.0).AND.(JY.LT.IY)) THEN
       JY       = NLX1
       SY       =-CUBE
       ENDIF
      ENDIF
      IF((IZ.EQ.NLX1).AND.(JZ.GT.IZ)) THEN
      JZ       = 0
      SZ       = CUBE
      ELSE
         IF((IZ.EQ.0).AND.(JZ.LT.IZ)) THEN
       JZ       = NLX1
       SZ       =-CUBE
       ENDIF
      ENDIF
C
C INDEX OF NEIGHBOURING CELL
      JC = 1 + JX + NLX*(JY + NLX*JZ)
      J  = LTOP(JC)
C
C BYPASS IF CELL IS EMPTY
      IF(J.EQ.0) GOTO 4001
  199 M = M + 1
      JADDR(M) = J
      RXX(M)   = X(J) + SX
      RYY(M)   = Y(J) + SY
      RZZ(M)   = Z(J) + SZ
      J        = LINK(J)
      IF(J.GT.0) GOTO 199
 4001 CONTINUE
C WE HAVE NOW FINISHED FINDING ALL NEIGHBOURING PARTICLES OF BOX IC.
```

Figure 3.4 continued

```
C RXX,RYY,RZZ CONTAINS ADDRESSES FROM MSTART+1 MAX
      MAX = M
      IF(MAX.GT.NVECT) THEN
       WRITE(6,*) ' MAX>NVECT',MAX,NVECT
       STOP
      ENDIF
      MSTRT1 = MSTART
C IF THERE ARE NO PARTICLES IN ANY OF THE NEIGHBOURING CELLS
C THEN I<J IN DOUBLE PARTICLE LOOP MEANS MSTRT1 CAN BE DECREASED
C BY ONE
      IF(MAX.EQ.MSTART) MSTRT1 = MSTART - 1
      IF(MSTRT1.LE.0) GOTO 5001
C
C NOW DO THE PARTICLE-PARTICLE INTERACTIONS (AT LAST!)
C THE IM INDEX ONLY SCANS THE HOME CELL
      MM  = 0
      DO 6001 IM = 1,MSTRT1
      I   = JADDR(IM)
      RXI = RXX(IM)
      RYI = RYY(IM)
      RZI = RZZ(IM)
C
C DO THE INNER VECTORIZED LOOP
C THE JM INDEX SCANS ALL OTHER PARTICLES WITH INDEXES
C GREATER THAN IM. THESE PARTICLES CAN BE IN EITHER THE
C HOME CELL OR IN NEIGHBOURING CELLS.
      DO 6002 JM = IM + 1,MAX
       J  = JADDR(JM)
C
       RX = RXI - RXX(JM)
       RY = RYI - RYY(JM)
       RSQ = RX*RX + RY*RY + RZ*RZ
C
       IF (RSQ .GT. RMAX) GO TO 6002
C
       MM        = MM + 1
       JJADR(MM) = J
       RSI  = 1./RSQ
       R6   = RSI**3
       U    = U + 4.*R6*(R6 - 1.) - SHIFT
C STORE THE FORCE-ON-I-DUE-TO-J ARRAYS FOR VECTOR SUMMATION
       FIJX(MM)= RSI*RX*R6*(2.*R6 - 1.)
       FIJY(MM)= RSI*RY*R6*(2.*R6 - 1.)
       FIJZ(MM)= RSI*RZ*R6*(2.*R6 - 1.)
 6002 CONTINUE
      LAST(I)  = MM
 6001 CONTINUE
C
      ILAST  = I
      MFIRST = 1
      MMAX   = MM
      IF(MMAX.LE.0) GOTO 6000
      DO 5998 IM = 1,MSTRT1
      I     = JADDR(IM)
      MLAST = LAST(I)
      IF(MFIRST.GT.MMAX) GOTO 5998
```

Figure 3.4 continued

```
*VOCL LOOP,NOVREC(FX,FY,FZ)
      DO 5999 MM = MFIRST,MLAST
      J       = JJADR(MM)
       FX(J) = FX(J) - FIJX(MM)
       FY(J) = FY(J) - FIJY(MM)
       FZ(J) = FZ(J) - FIJZ(MM)
       FX(I) = FX(I) + FIJX(MM)
       FY(I) = FY(I) + FIJY(MM)
       FZ(I) = FZ(I) + FIJZ(MM)
 5999 CONTINUE
       MFIRST= MLAST + 1
 5998 CONTINUE
 6000 CONTINUE
C
C FORCE ACCUMULATION COMPLETE
C
 5001 CONTINUE
C
C END THE HOME CELL SCANNING LOOP
C
      RETURN
      END
```

Figure 3.4 continued

3.5 Neighbour List Algorithms

An alternative approach to efficiently finding the near neighbours is to keep a list of near neighbours for each particle which is updated periodically. Thus the cost of finding the near neighbours is amortized over several timesteps. The near neighbours for each particle are looked for in a volume defined by a radius slightly larger than the cutoff distance. The list then contains particles that may possibly interact with the central one, only those within the cutoff sphere actually interact. The neighbour list can be formed either using a cell code or nested loops to determine the possibly interacting near neighbours. The choice depends on system size and cutoff distance. Having formed the neighbour list the interparticle interactions are calculated using that neighbour list until it is necessary to reform the list. Deciding when to reform the list is not as obvious as may at first be thought.

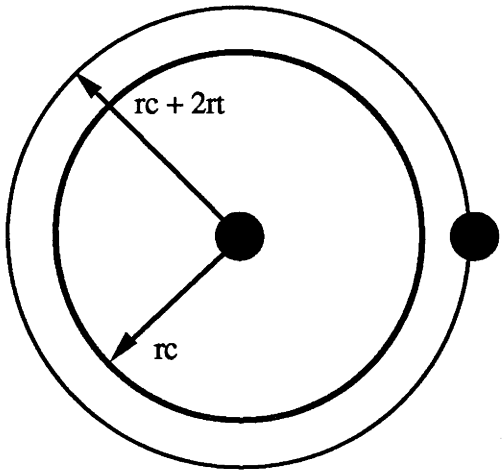
The simplest scheme is to update the list every x timesteps, where x may be around 10. The problem then is how much beyond the cutoff distance has to be searched to find the possibly interacting neighbours so as to include **all** interactions that will occur over the next x timesteps. This is simply a matter of trial and error. If too large a distance is chosen then the program will not be as fast as it should, too small a distance and interactions will be missed.

A for more reliable scheme for deciding when to update the list is to determine how far a particle has moved since the list was formed. If the near neighbours are found in a volume of radius $r_c + 2r_t$, when a particle has diffused a distance r_t the list must be reformed as this is the worst case; if 2 particles have diffused a distance r_t towards each other then they are at the limit of interacting this timestep without being included in each others neighbour list. This situation is shown in figure 3.5.

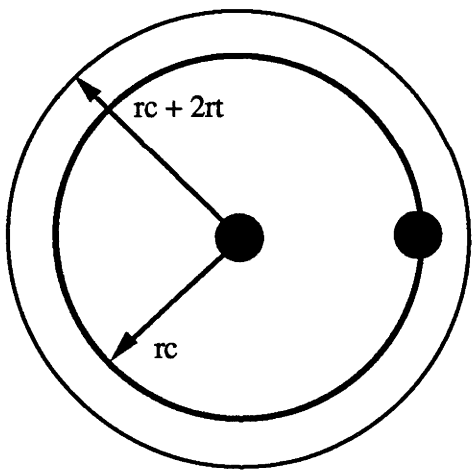
The above algorithm guarantees that no interactions are missed at equilibrium. However in a system under shear there is an added complication; particles can approach each other not just through diffusion but also through the shear motion, the streaming motion as it was called in chapter 2. At zero temperature there is no diffusion and hence at equilibrium the distances between particles do not change, however under shear particles with slightly different y-coordinates will have slightly different velocities and will *eventually* come into interaction range with each other. To compare the distance moved through the shear motion relative to other particles is an $O(N^2)$ problem and we wish to avoid it, especially as this is just a test to determine whether the list needs updating. Fortunately we can make an assumption based on worst case behaviour that simplifies matters. It can be shown that the worst case, ie the most rapid approach of the two neighbour list spheres, occurs when two particles $r_c + 2r_t$ apart form an angle of 45^o with the x-axis. This leads to a new criterion for the formation of the neighbour list,

$$r_d + \gamma t/4\ (r_c + 2r_t) > r_t \qquad (3.1)$$

where t is the time since the list was formed and r_d is the maximum distance a particle has diffused. It can be seen that this reduces correctly to the equilibrium case.



The two particles are initially just outside the extended cutoff volume, they are not included as neighbours.



Each particle has now moved rt towards the other and the two particles are now at the limit of actually interacting.

Figure 3.5 Worst case situation for updating the neighbour list.

3.6 Additional Tricks

To completely vectorize the force summation it is necessary to ignore Newton's Law and actually calculate the i-j and the j-i interactions. This has to be done to avoid a recursive reference which inhibits vectorization (point iii in section 3.1). In systems with small cutoffs the number of neighbours per particle is quite small relative to the number of particles in the system. Vectorization would be improved if we vectorized over the number of particles rather than the number of neighbours. Because density fluctuations are small in dense systems we can invert the loops and vectorize over the number of particles with little overhead. Thus criteria i and iii from section 3.1 are satisfied. However the code has now become substantially more complicated, contrary to what is desired. Modern compilers though manage to vectorize such code reasonably well, and we find that this code is the best performing of the algorithms described so far. It is possible to improve the speed further by introducing a second, outer neighbour list from which the inner list is formed. This method is very expensive on memory and a fairly simple extension to a single neighbour list program and so it is not discussed further.

One nice feature of the present code is that its performance depends on r_t, the extra distance added to the cutoff for the neighbour search. Thus we can tune the program for optimal performance by varying r_t. The optimal value will depend on system size, cutoff distance and the actual computer. For the Fujitsu VP100, a large (~>1000 particles) soft-disk system with a 1.5σ cutoff using cell code to form the neighbour list, the optimal value of r_t is $r_t = 0.1$, although the minimum is fairly broad. The performance of the programs is shown in table 3.1. The code for the forces routine for the cell-based neighbour list is shown in figure 3.6.

Algorithm	Performance at equilibrium (1000 timesteps/hour)
Cell code	40
Neighbour list	250
Neighbour list, inverted loops	400

Table 3.1 Performance of three algorithms for 896 soft disks with a cutoff of 1.5σ.

```
      SUBROUTINE FORCE

      PARAMETER (NP=896,NC=30,N0=7,NM=2,NTCF=3,NAB=37)
      IMPLICIT REAL*8 (A-H,O-Z)

      REAL MEAN

      COMMON/COORD/ X0(NP),Y0(NP),X1(NP),Y1(NP),X2(NP),Y2(NP),
     &              X3(NP),Y3(NP)
      COMMON/MOMENT/PX0(NP),PY0(NP),PX1(NP),PY1(NP),PX2(NP),PY2(NP),
     &              PX3(NP),PY3(NP)
      COMMON/FORCE/ REG(0:NM,NTCF),FX(NP),FY(NP),A,
     &              PXT(NP),PYT(NP),DT(NP)
      COMMON/AV/    AVER(NM,10),MEAN(0:NM,NTCF),FCORR(0:NM,NTCF)
      COMMON/COUNT/ K1(NM),KTCF(0:NM)
      COMMON/FLUX/  U,PT(2,2)
      COMMON/TIME/  DELTA,ODELT,TR,DR,SHEAR,GAMMA
      COMMON/TEMP/  ALPH,A1,A2,A3,A4,EK,ACOR
      COMMON/PARM/  CUBE,CUBE2,CUBINV,RCUT,RMAX,GT,DXD,SHIFT
      COMMON/INTAR/ NTYPE,ICOUNT,KB,ISEED,NGAUS,KPROP,NON,NTIME,LCHECK
      COMMON/CEL/   DCELL,NCELL,JN(NC,NC),ICELL(N0,NC,NC),INMAX,
     &              INP(NP),NABR(NP,NAB),IGMAX,JNGMAX,JNMAX
      COMMON/DIF/   XD(NP),YD(NP),RTOL,DELTAT,
     &              TOL,TOL2,RMAT,RMAT2,DMAX,DMAX2,ICOL(NP)

C --- EVALUATE DISTANCE DIFFUSED SINCE LIST WAS FORMED

      DELTAT = DELTAT + DELTA
      DO 10 I = 1, NP
        YD(I) = YD(I) + PY0(I)*DELTA
        XD(I) = XD(I) + (YD(I)*SHEAR + PX0(I))*DELTA
        RI    = XD(I)**2 + YD(I)**2
        IF (RI .GT. DMAX) DMAX = RI
   10 CONTINUE

C --- DMAX INCLUDES DIFFUSION AND SHEAR STREAMING COMPONENTS

      DMAX = SQRT(DMAX) + 0.25*SHEAR*(RCUT + 2.0*TOL)*DELTAT

      IF (DMAX .LT. TOL) THEN
C ----- GO STRAIGHT TO FORCES CALCULATION
        GOTO 500
      ENDIF

C --- FORM INTERACTION LISTS
C --- NABR(I,IN) IS THE J-NAME OF THE IN-TH NABOR OF PARTICLE I
C --- INMAX IS THE MAX # OF NABORS THAT ANY PARTICLE HAS
```

Figure 3.6 Forces routine using a cell based neighbour list.

```
C --- ZEROING ARRAYS

      DO 14 IY = 1,NCELL
        DO 13 IX = 1,NCELL
          JN(IX,IY) = 0
          DO 7 J = 1,N0
            ICELL(J,IX,IY) = 0
    7     CONTINUE
   13   CONTINUE
   14 CONTINUE

C --- DO PERIODIC BOUNDARY CONDITIONS AND SORT INTO CELLS

      DO 1 I = 1, NP
        INP(I) = 0
        RX = X0(I) - CUBE2
        RY = Y0(I) - CUBE2
        CIS = CUBE*NINT(RY*CUBINV)
        X1(I) = X1(I) - CIS*GT*DELTA
        RY = RY - CIS
        RX = RX - DXD*CIS
        RX    = RX - CUBE*NINT(RX*CUBINV)
        X0(I) = RX + CUBE2
        Y0(I) = RY + CUBE2
        IX = X0(I)/DCELL + 1
        IY = Y0(I)/DCELL + 1
        JN(IX,IY) = JN(IX,IY) + 1
        ICELL(JN(IX,IY),IX,IY) = I
    1 CONTINUE

      JNMAX = 0
      JNTOT = 0
      DO 205 IY = 1,NCELL
        DO 203 IX = 1,NCELL
          IF (JN(IX,IY) .GT. JNMAX) JNMAX = JN(IX,IY)
          JNTOT = JNTOT + JN(IX,IY)
  203   CONTINUE
  205 CONTINUE

C --- CHECK THAT ALL PARTICLES WERE ACCOUNTED FOR AND ARRAY
C --- BOUNDS NOT EXCEEDED

      IF (JNTOT .NE. NP) THEN
        WRITE(6,'('' FOUND '',I6,'' PARTICLES'')') JNTOT
      ENDIF
      IF (JNMAX .GT. N0) THEN
        WRITE(6,*) '  N0 EXCEEDED : ',JNMAX
        STOP
      ENDIF

C --- ZERO THE NEIGHBOUR LIST

      DO 2 INAB = 1, NAB
        DO 2 I  = 1, NP
    2     NABR(I,INAB) = 0
```

Figure 3.6 continued.

```
C --- EXTRA CODE FOR CELL CALCULATIONS

      NCM1 = 2*NCELL - 1
      IDXD = NINT(DXD*NCELL - 0.499999)
      DO 20 IX = 1, NCELL
      DO 20 IY = 1, NCELL
      IF (JN(IX,IY) .EQ. 0) GOTO 20

C --- HOME CELL

      DO 25 NI = 1, JN(IX,IY)-1
       I = ICELL(NI,IX,IY)
       DO 25 NJ = NI+1, JN(IX,IY)
        J = ICELL(NJ,IX,IY)
C ----- WE NEED TO INCLUDE BOTH I-J AND J-I INTERACTIONS
        INP(J) = INP(J) + 1
        NABR(J,INP(J)) = I
        INP(I) = INP(I) + 1
        NABR(I,INP(I)) = J
   25 CONTINUE

      DO 30 ICY = 0, 1
      IIY = MOD(IY + ICY + NCM1, NCELL) + 1

      IF (ICY .EQ. 0) THEN
        ILOWER =   1
      ELSE
        ILOWER = - 1
      ENDIF

      IF (IY + ICY .GT. NCELL) THEN
        IJK    =   1
        ILOWER = - 2
       ELSE
        IJK    =   0
      ENDIF

      DO 30 ICX = ILOWER, 1
      IIX = MOD(IX + ICX - IJK*IDXD + NCM1, NCELL) + 1

C --- ARE THERE ANY PARTICLES IN CELL (IIX,IIY)

      IF (JN(IIX,IIY) .EQ. 0) GOTO 30

C --- LOOK UP PARTICLE INDICES

       DO 40 IP = 1, JN(IX,IY)
         I = ICELL(IP,IX,IY)
         DO 40 JP = 1, JN(IIX,IIY)
           J = ICELL(JP,IIX,IIY)
C -------- INCLUDE BOTH I-J AND J-I INTERACTIONS
           INP(J) = INP(J) + 1
           NABR(J,INP(J)) = I
           INP(I) = INP(I) + 1
           NABR(I,INP(I)) = J
   40   CONTINUE
   30  CONTINUE
   20 CONTINUE
```

Figure 3.6 continued.

```
      DELTAT = 0.0
      DMAX = 0.0
      INMAX = 0

C --- CALCULATE THE MAXIMUM NUMBER OF NEIGHBOURS PER PARTICLE
C --- AND ZERO DISPLACEMENTS

      DO 100 I = 1, NP
        IF (INP(I) .GT. INMAX) INMAX = INP(I)
        XD(I) = 0.0
        YD(I) = 0.0
  100 CONTINUE

C --- CHECK THAT ARRAY BOUNDS NOT EXCEEDED

      IF (INMAX .GT. NAB) THEN
        WRITE(6,*) '  NAB EXCEEDED : ',INMAX
        STOP
      ENDIF

C --- FORCE CALCULATION STARTS HERE

  500 CONTINUE

      DO 110 I = 1, NP
        FX(I) = 0.0
        FY(I) = 0.0
        PXT(I) = 0.0
        PYT(I) = 0.0
  110 CONTINUE
      PT(1,1) = 0.0
      PT(1,2) = 0.0
      PT(2,1) = 0.0
      PT(2,2) = 0.0
      U  = 0.0

C --- LOOPS ARE INVERTED, VECTORIZING OVER THE NUMBER OF PARTICLES

      DO 150 INAB = 1,INMAX
        DO 152 I = 1,NP
          J = NABR(I,INAB)
C ------- J IS ZERO IF PARTICLE I DOES NOT HAVE 'INAB' NEIGHBOURS
          IF (J .NE. 0) THEN
            RX = X0(I) - X0(J)
            RY = Y0(I) - Y0(J)
            CIS = CUBE*NINT(RY*CUBINV)
            RY  = RY - CIS
            RX  = RX - DXD*CIS
            RX  = RX - CUBE*NINT(RX*CUBINV)
            RSQ = RX*RX + RY*RY
            IF (RSQ .LE. RMAX) THEN
              RSI = 1.0/RSQ
              R12 = RSI**6
              FAC = 1.0
              UB = R12 - SHIFT
```

Figure 3.6 continued.

```
              FJIX = RSI*R12*RX
              FJIY = RSI*R12*RY
              U    = U + UB*0.5
              FX(I) = FX(I) + FJIX
              FY(I) = FY(I) + FJIY
C ------------ COMPUTE FLUX TENSORS..
              PT(1,1) = PT(1,1) + RX*FJIX
              PT(1,2) = PT(1,2) + RX*FJIY
              PT(2,1) = PT(2,1) + RY*FJIX
              PT(2,2) = PT(2,2) + RY*FJIY
            ENDIF
          ENDIF
  152   CONTINUE
  150 CONTINUE

C --- PRESSURE TENSOR HAS FACTOR OF 6 IN IT BECAUSE
C --- WE INCLUDED I-J AND J-I INTERACTIONS

      PT(1,1) = 6.0*PT(1,1)
      PT(1,2) = 6.0*PT(1,2)
      PT(2,1) = 6.0*PT(2,1)
      PT(2,2) = 6.0*PT(2,2)

C --- FORCES HAVE FACTOR 12 BECAUSE ONLY SUMMED OVER I INDEX

      DO 153 I = 1,NP
        FX(I) = 12.0*FX(I)
        FY(I) = 12.0*FY(I)
  153 CONTINUE

C --- CODE FOR GAUSSIAN ISOTHERMAL CONSTRAINT

      IF (NGAUS .EQ. 1) THEN
        PXPY = 0.0
        ANUM = 0.0
        ADEN = 0.0
        DO 113 I = 1,NP
          ANUM = ANUM + PX0(I)*FX(I)  + PY0(I)*FY(I)
          ADEN = ADEN + PX0(I)*PX0(I) + PY0(I)*PY0(I)
          PXPY = PXPY + PX0(I)*PY0(I)
  113   CONTINUE
        ALPH = (ANUM - SHEAR*PXPY)/ADEN
      ENDIF

      RETURN
      END
```

Figure 3.6 continued.

4. Constant Stress Ensemble Simulations

In chapter 2 we described planar Couette flow performed at a constant strain rate. However, in many experimental situations, it is not the strain rate that is constant but rather the shear stress. Two examples of this are the slip and slide creeping flow motion of glaciers and the fracturing of a crystal in the determination of its yield stress. To enhance the similarity with physical experiments it would be useful to perform simulations where the shear stress is the driving variable rather than the shear rate. The constant strain rate equations of motion combined with the standard isokinetic thermostat force the system to always flow.

Simulation in the constant stress ensemble is an example of a simulation where the thermodynamic flux is the independent variable and the thermodynamic force fluctuates in response. By analogy with electric circuit theory, the constant flux ensemble is called the Norton ensemble and the constant force ensemble the Thevenin ensemble [25]. Two methods for performing simulations at constant stress have recently been formulated. The first of these was by Brown and Clark [26] who used a constraint derived from Gauss's principle, while the second was a Nosé-Hoover mechanism by Evans and Ely [27], and later extended by Hood, Evans and Morriss [28].

4.1 Brown and Clark's Constant Stress Ensemble [26]

Like conventional shear flow simulations, the equations of motion are :

$$\dot{q}_i = \frac{p_i}{m} + \gamma i q_{yi}$$

$$\dot{p}_i = F_i - \gamma i p_{yi} - \alpha p_i \tag{4.1}$$

where α is a thermostatting multiplier, which for the Gaussian isokinetic case is

$$\alpha = \frac{\sum F_i \cdot p_i - \gamma \sum p_{xi} p_{yi}}{\sum p_i^2} \tag{4.2}$$

The difference in this case is that γ is not a constant value, but instead has its own equation of motion. The off-diagonal element of the pressure tensor, P_{xy}, is defined by the equation

$$P_{xy} V = \sum_i p_{xi} p_{yi} + q_{xi} F_{yi} \tag{4.3}$$

If we differentiate this with respect to time we get the constraint equation :

$$V \dot{P}_{xy} = \sum_i \frac{p_{xi} \dot{p}_{yi} + \dot{p}_{xi} p_{yi}}{m} + \sum_i \dot{q}_{xi} F_{yi} + q_{xi} \dot{F}_{yi} \tag{4.4}$$

We do not require that dP_{xy}/dt be zero, this allows for a frequency dependent stress to be imposed on the system. Substituting for **p** and **q** and rearranging leads to the Gaussian constraint for constant imposed shear stress :

$$\gamma = \frac{\sum(2p_{xi}F_{yi} + F_{xi}p_{yi} - 2\alpha p_{xi}p_{yi})/m + \sum p_{xi}\dot{F}_{yi} - \dot{P}_{xy}V}{\frac{\sum(p_{yi}^2/m - q_{yi}F_{yi}) - 2(\sum p_{xi}p_{yi})^2}{m\sum p_i^2}} \tag{4.5}$$

In the above expression a substitution for α has to be made. Also the term involving dF_y/dt involves the second derivative of the potential so a further substitution has to be made. We assume that the potential is pairwise-additive (the extension to three body terms would be extremely complicated). After some algebra the full expression for the Gaussian isokinetic-constant stress multiplier is obtained.

$$\gamma = \frac{L_1 - L_2 - \dot{P}_{xy}V}{L_3 + L_4} \tag{4.6}$$

Where the sums L_1 to L_4 are defined by

$$L_1 = \frac{1}{m}\sum_i \left(2\,p_{xi}\,F_{yi} + F_{xi}\,p_{yi} - 2\,(\mathbf{p}_i \cdot \mathbf{F}_i)\frac{\sum_j p_{xj}\,p_{yj}}{\sum_k p_k^2} \right) \tag{4.7}$$

$$L_2 = \frac{1}{m}\sum_i \sum_j \frac{q_{xij}}{|q_{ij}|}\left(p_{yij}\,\phi'_{ij} + (\mathbf{q}_{ij} \cdot \mathbf{p}_{ij})\frac{q_{yij}}{q_{ij}^2}\left(\frac{\phi''_{ij}}{|q_{ij}|} - \phi'_{ij}\right)\right) \tag{4.8}$$

$$L_3 = \frac{1}{m}\sum_i p_{yi}^2 - \frac{2}{m}\frac{\left(\sum_i p_{xi}\,p_{yi}\right)^2}{\sum_i p_i^2} \tag{4.9}$$

$$L_4 = \sum_i \sum_j \frac{q_{xij}^2\,q_{yij}^2}{q_{ij}^2}\left(f''_{ij} - \phi'_{ij} \cdot \frac{\left(1 - q_{ij}^2/q_{xij}^2\right)}{|q_{ij}|}\right) \tag{4.10}$$

The expression was derived by Clark and Brown in 1986, but is very complicated to analyze theoretically. They used it in the analysis of the yield stress for the soft sphere system at crystalline state points. We show here a simpler method that uses a Nosé-Hoover feedback mechanism.

4.2 Nosé-Hoover Stress Ensemble

The Nosé-Hoover stress ensemble has the same equations for d**q**/dt and d**p**/dt as previously. We also use a Gaussian thermostat with the same expression as before. The difference is in the expression for $\dot{\gamma}$, which we derive from Nosé-Hoover principles [14,15]

$$\dot{\gamma} = \frac{1}{Q_\gamma}(P_{xy}(t) - S_{xy}(t)) = \frac{1}{\tau_\gamma^2}\left(\frac{P_{xy}(t)}{S_{xy}(t)} - 1\right) \tag{4.11}$$

where $-S_{xy}$ is the desired shear stress. Q_γ is the Nosé-Hoover damping constant which determines the rate at which the actual value of P_{xy} returns to the set value of S_{xy}. In the steady state, where P_{xy} is independent of t, $\langle d\gamma/dt \rangle = 0 = \langle P_{xy} \rangle - S_{xy}$. Obviously the second expression for $d\gamma/dt$ cannot be used at equilibrium where $S_{xy} = 0$.

Evans and Ely [27] derived a Green-Kubo relationship for the reciprocal of the zero frequency shear viscosity in terms of fluctuations in the value of γ in the $S_{xy} = 0$ equilibrium ensemble.

$$\eta^{-1} = \beta V \int_0^\infty dt \langle \gamma(0)\gamma(t) \rangle_{S_{xy}=0} \tag{4.12}$$

and

$$-\beta V \int_0^\infty dt \langle P_{xy}(t) P_{xy}(0) \rangle_{S_{xy}=0} = 0 \tag{4.13}$$

Equation (4.12) is the fundamental result of their work. However, this result only pertains to the zero-frequency case. In the next section we generalize these results to the case $\omega \neq 0$.

4.3 Frequency Dependent Stress Ensemble Time Correlation Functions

In chapter 1 a formal outline was given to the derivation of some of the results of non-linear response theory. The results described in this section are based on linear response theory, but with a time-dependent external field. We derive the results in considerable detail to show how the theory is applied.

From the equations of motion (4.11), the adiabatic ($\alpha = 0$) time derivative of $H_0 = \Sigma p_i^2/2m + \Phi$ is

$$\dot{H}_0^{ad} = - P_{xy} V \gamma \tag{4.14}$$

The Nosé-Hoover constant stress and Gaussian isothermal dynamics satisfy the Liouville equation in which phase space behaves as a 6N + 1 dimensional fluid. The equilibrium distribution function, f_0, is a function of **p**,**q** and γ ; $f_0 = f_0(\Gamma,\gamma)$. The Liouville equation for this system is

$$\frac{df_0}{dt} = - f_0 \left(\frac{\partial}{\partial \Gamma} \cdot \dot{\Gamma} + \frac{\partial}{\partial \gamma} \dot{\gamma}\right) \tag{4.15}$$

If we consider the adiabatic time derivative of $H_0 + {}^1/_2 Q_\gamma \gamma^2$, we find that

$$\begin{aligned} \frac{d}{dt}\left(H_0 + \frac{1}{2} Q_\gamma \gamma^2\right)\Big|^{ad} &= \dot{H}_0^{ad} + Q_\gamma \gamma \dot{\gamma} \\ &= - P_{xy} V \gamma + (P_{xy} - S_{xy}) V \gamma \\ &= -S_{xy} V \gamma \end{aligned} \tag{4.16}$$

Thus at equilibrium, where $S_{xy} = 0$, $H_0 + {}^1/_2 Q_\gamma \gamma^2$ is a constant of the adiabatic equations of motion. The equilibrium analogue of the canonical distribution for the Nosé-Hoover

zero stress ensemble is therefore

$$f_0 = \frac{\exp(-\beta(H_0 + \frac{1}{2}Q_\gamma \gamma^2))}{\int d\Gamma \int d\gamma \exp(-\beta(H_0 + \frac{1}{2}Q_\gamma \gamma^2))} \tag{4.17}$$

We now wish to calculate the linear response of a system, characterized initially by the distribution, f_0, to an externally imposed shear stress, $-S_{xy}(t)$.

The linear response of an arbitrary variable as a function of phase space, $B(\Gamma)$ is

$$\langle B(t) \rangle = \langle B(0) \rangle - \int_0^t ds \int d\Gamma\, B(\Gamma) \exp(-iL_0(t-s))\, i\Delta L(s)\, f_0(\Gamma) \tag{4.18}$$

where iL_0 is the equilibrium thermostatted f-Liouvillean and $\Delta L = L(s) - L_0$. We need to calculate $i\Delta L(s)\, f_0$, using the equations of motion and the distribution function obtained previously. Since the adiabatic incompressibility of phase space, $A\Gamma$ condition is satisfied,

$$\begin{aligned} i\Delta L(s)\, f_0 &= -\beta \left(\dot{H}_0 + \dot{\gamma}\gamma Q_\gamma + f_0 \frac{\partial}{\partial \Gamma} \cdot \dot{\Gamma} \right) \\ &= -\beta\, \dot{H}_0^{ad} + (\dot{\gamma}^{ad} \gamma Q_\gamma) \\ &= \beta V S_{xy}\, \gamma(\Gamma)\, f_0 \end{aligned} \tag{4.19}$$

Combining these results, we find that the linear response for an arbitrary phase variable, $B(\Gamma)$, in the Nosé-Hoover stress ensemble is

$$\langle B(t) \rangle = \langle B(0) \rangle - \beta V \int_0^t ds \langle B(t-s)\, \gamma \rangle_{S_{xy}=0} S_{xy}(s) \tag{4.20}$$

Although the stress $-S_{xy}(t)$ is the driving force, there will be fluctuations in both $P_{xy}(t)$ and

$\gamma(t)$. The Nosé-Hoover method does not rigidly fix the instantaneous value of P_{xy} to be S_{xy}. From equation (4.20) the time-dependent values of these two quantities are

$$< \gamma (t) > = -\beta V \int_0^t d\tau < \gamma(t-\tau)\, \gamma (0) >_{S_{xy}=0} S_{xy} (\tau) \tag{4.21}$$

and

$$< P_{xy}(t) > = -\beta V \int_0^t d\tau < P_{xy} (t-\tau)\, \gamma (0) >_{S_{xy}=0} S_{xy} (\tau) \tag{4.22}$$

We can rewrite these equations so that they involve a susceptibility, χ, which is defined by $\chi(t) = -\beta < B(t)J(0) >$, where **J** is the dissipative flux. This is defined by the equation,

$$\mathbf{J}\, \mathbf{F}_e(t) = - \dot{H}_0^{ad}(t) \tag{4.23}$$

In this case **J** is $V\gamma$, and $\mathbf{F_e}(t)$ is the external driving field, $-S_{xy}(t)$. We can Fourier-Laplace transform equations (4.21) and (4.22). The Fourier-Laplace transform is defined by

$$\tilde{\chi}(\omega) = \int_0^\infty dt\ \exp(-i\omega t)\, \chi(t) \tag{4.24}$$

Applying the transformation we obtain,

$$\tilde{\gamma} = \tilde{\chi}_{\gamma\gamma} S_{xy}(\omega) \tag{4.25}$$

$$\tilde{P}_{xy} = \tilde{\chi}_{P\gamma} S_{xy}(\omega) \tag{4.26}$$

The definitions for these frequency dependent susceptibilities are

$$\tilde{\chi}_{\gamma\gamma} = -\beta V \langle \gamma(0)\, \tilde{\gamma}(s) \rangle_{S_{xy}=0} \tag{4.27}$$

$$\tilde{\chi}_{P\gamma} = -\beta V \langle P_{xy}(0)\, \tilde{\gamma}(s) \rangle_{S_{xy}=0} \tag{4.28}$$

$$\tilde{\chi}_{PP} = -\beta V \langle P_{xy}(0)\, \tilde{P}_{xy}(s) \rangle_{S_{xy}=0} \tag{4.29}$$

where $s = i\omega$. The frequency dependent shear viscosity is defined as

$$\tilde{\eta} = -\frac{\tilde{P}_{xy}}{\tilde{\gamma}} \tag{4.30}$$

Thus the frequency dependent viscosity in the Norton ensemble is the ratio of two susceptibilities,

$$\tilde{\eta} = -\frac{\tilde{\chi}_{p\gamma}}{\tilde{\chi}_{\gamma\gamma}} \tag{4.31}$$

Equation (4.31) is awkward because it involves cross time correlation functions, which are difficult to compute with a high precision. However, we can derive expressions for the viscosity which involve just one of these susceptibilities, $\chi_{p\gamma}$, χ_{pp} or $\chi_{\gamma\gamma}$. We can do this by using the equations of motion for $d\gamma/dt$. This gives

$$\tilde{\chi}_{p\gamma} = 1 + \frac{sQ_{\gamma}}{V}\tilde{\chi}_{\gamma\gamma} \tag{4.32}$$

Substituting this into equation (4.31), we can calculate the viscosity from strain rate fluctuations in the Norton ensemble,

$$\tilde{\eta}(s) = -\frac{(\frac{sQ_\gamma}{V}\tilde{\chi}_{\gamma\gamma} + 1)}{\tilde{\chi}_{\gamma\gamma}} \tag{4.33}$$

We can also derive expressions for the viscosity in terms of equilibrium stress fluctuations. With a method similar to that used in deriving (4.32) we find that

$$\tilde{\chi}_{P\gamma} = -\frac{V}{sQ_\gamma}\tilde{\chi}_{PP} \tag{4.34}$$

Substituting this into equation (4.31) we get a relationship for the shear viscosity which does not depend on cross susceptibilities,

$$\tilde{\eta}(s) = -\frac{V}{sQ_\gamma}\frac{\tilde{\chi}_{PP}}{\tilde{\chi}_{\gamma\gamma}} \tag{4.35}$$

We can eliminate the strain rate susceptibility by combining equations (4.33) and (4.35),

$$\tilde{\chi}_{\gamma\gamma} = -\frac{V}{sQ_\gamma}\left(\frac{V}{sQ_\gamma}\tilde{\chi}_{PP} + 1\right) \tag{4.36}$$

We can now derive expressions for $\tilde{\eta}(s)$ in terms of any one susceptibility,

$$\tilde{\eta}(s) = -\frac{(\frac{sQ_\gamma}{V}\tilde{\chi}_{\gamma\gamma} + 1)}{\tilde{\chi}_{\gamma\gamma}} \tag{4.33}$$

$$= -\frac{sQ_\gamma}{V}\left(\frac{\tilde{\chi}_{P\gamma}}{\tilde{\chi}_{P\gamma} - 1}\right) \tag{4.37}$$

$$= -\frac{\tilde{\chi}_{PP}}{1 + \frac{V}{sQ_\gamma}\tilde{\chi}_{PP}} \tag{4.38}$$

These equations show how we can obtain viscosities by measuring the stress autocorrelation function, the strain-rate autocorrelation function or the stress-strain time correlation function in the Norton ensemble. Because of the difficulty in measuring cross time correlation functions to a high degree of precision, equation (4.37) is the least preferred of these equivalent forms, and so will not be discussed further. We can rearrange the above equations to obtain the susceptibilities as a function of the frequency dependent shear viscosity. We find that

$$\tilde{\chi}_{PP} = -\frac{\tilde{\eta}(s)}{1 - \frac{V}{sQ_\gamma}\tilde{\eta}(s)} \tag{4.39}$$

and

$$\tilde{\chi}_{\gamma\gamma} = -\frac{1}{\tilde{\eta}(s) + \frac{sQ_\gamma}{V}} \tag{4.40}$$

Some important limits to note from equation (4.39) are :

$$\lim_{V\to\infty} \tilde{\chi}_{PP} = -\tilde{\eta}(s) \tag{4.41}$$

$$\lim_{s\to 0} \tilde{\chi}_{PP} = \frac{sQ_\gamma}{V} = 0 \tag{4.42}$$

Equations (4.41) and (4.42) show that in the limit of infinite Q_γ, the Norton ensemble is the same as the Thevenin ensemble, except at $\omega = 0$.

4.4 Simulation Results

To test the equations obtained we performed two sets of simulations of the Lennard-Jones fluid. We used 108 particles, with the potential truncated at $r_c^* = 2.5$. The reduced temperature and density were $T^* = 0.722$ and $\rho^* = 0.8442$, the standard triple point state. All quantities are reduced with respect to the Lennard-Jones parameters, σ and ε and the particle mass, m. A Gaussian rather than Nosé-Hoover thermostat was used because of its greater computational efficiency [29]. A fifth-order Gear predictor-corrector scheme with an integration timestep, Δt^* of 0.005, was used in all simulations.

To check the correctness of the program the first set of simulations were non-equilibrium runs comparing the average values of P_{xy} and γ for an imposed shear stress, $-S_{xy}$ of 2.1 at three different values of Q_γ, namely $Q_\gamma = 50, 100, 200$. The results are shown in table 4.1. The runs were 20 000 time steps in each case. As can be seen, the results are in very good agreement with each other, and with the earlier results of Evans and Ely [27], and consistent with the Thevenin ensemble result.

Equilibrium Norton ensemble simulations were performed to obtain the strain-rate and the stress autocorrelation functions. Time correlation functions are difficult to compute, and require very long runs for good statistics. The runs were 200 000 timesteps for each of the three values of Q_γ used previously. The time correlation functions obtained are plotted as discrete points in figures 4.1 and 4.2. Error bars are not shown, but are typically less than 5%, for $t^* \leq 1.0$.

Given an expression for the stress autocorrelation function in the Thevenin ensemble we can calculate the time correlation functions in the Norton ensemble using equations (4.39) and (4.40). We used an expression developed by Evans [30] in 1981 which summarizes all NEMD and linear visco-elasticity data for the Lennard-Jones fluid at the triple point. He found that to a good approximation the integral of the stress autocorrelation

function as a function of the upper limit is

$$\eta(t) = \frac{G_\infty}{\exp(-t/\tau) + \dfrac{G_\infty t^{1.5}}{A_{\tilde{\eta}}}} \tag{4.43}$$

where G_∞ is the infinite frequency shear modulus, with the value $G_\infty^* = 24.89$. The other constants have the values

$$A_{\tilde{\eta}}^* = \frac{0.65}{\sqrt{2\pi}} \quad \text{and} \quad \tau^* = 0.12$$

This equation is consistent with the enhanced long time tail behaviour [30], decaying as $t^{-3/2}$ at long times. Because of this slow algebraic decay, we require a large upper integration limit in order to compute accurate viscosities.

Equation (4.43) was Fourier transformed numerically, and used in equations (4.39) and (4.40) which were used to calculate the frequency-dependent stress and strain autocorrelation functions in the Norton ensemble for the three values of Q_γ used. The absolute values of the real parts of the resulting expressions are plotted in figures 4.3 and 4.4. Figure 4.3 shows the limits quite clearly; as Q_γ approaches infinity, the Norton ensemble curve approaches that of the Thevenin ensemble, except at the origin, where the function goes to zero. This is in agreement with the prediction of equations (4.41) and (4.42). Figure 4.4 shows that regardless of Q_γ, the long time or zero-frequency limit of the strain rate auto-correlation function is independent of Q_γ, and is in fact the reciprocal of the viscosity. These equations are therefore seen to be consistent with the zero-frequency results of Evans and Ely. The real parts of these expressions were Fourier-Laplace transformed to produce the time-domain autocorrelation functions. These functions are plotted as continuous lines in figures 4.1 and 4.2. As can be seen, the agreement between the simulation data points and the calculated curves is good.

From the work of Evans and Ely, we have an expression for the shear viscosity in the Norton ensemble, equation (4.12). We can calculate the equilibrium viscosities from the three runs using this equation. Because the strain-rate autocorrelation function exhibits substantial noise for $t^* > 2$, and is close to zero beyond that point, all time integrations in equation (4.12) were truncated. The truncation times, together with the viscosities obtained, are shown in Table 4.2. Although the viscosities calculated in this way are not very accurate, they do agree within statistical uncertainties with the known viscosity for this system [31], which is also shown in Table 4.2.

Non Equilibrium MD results in the Norton Ensemble.

Q_γ*	$\langle P_{xy}^* \rangle$	$\langle \gamma^* \rangle$	Ensemble
50	-2.100 ± 0.002	1.03 ± 0.01	N
100	-2.099 ± 0.005	1.03 ± 0.01	N
200	-2.101 ± 0.006	1.03 ± 0.01	N
-	-2.09 ± 0.02	1.00	T

Table 4.1 Average value of P_{xy} and γ for an imposed shear stress, $-S_{xy} = 2.1$ Run lengths were 20 000 time steps. This shows the equivalence of Norton, N, and Thevenin, T, ensembles in the nonlinear regime.

Equlibrium MD viscosities computed in the Norton Ensemble.

Q_γ *	t^*_{cut}	$\eta^*(\gamma^* = 0)$	Ensemble
50	1.5	3.3 ± 0.2	N
100	2.0	3.4 ± 0.3	N
200	2.0	3.1 ± 0.2	N
-	∞	3.15 ± 0.05	T

Table 4.2 Values of shear viscosity obtained for three values of Q_γ. t^*_{cut} is the upper limit of the integration in equation (4.12). Each run was 200 000 time steps long.

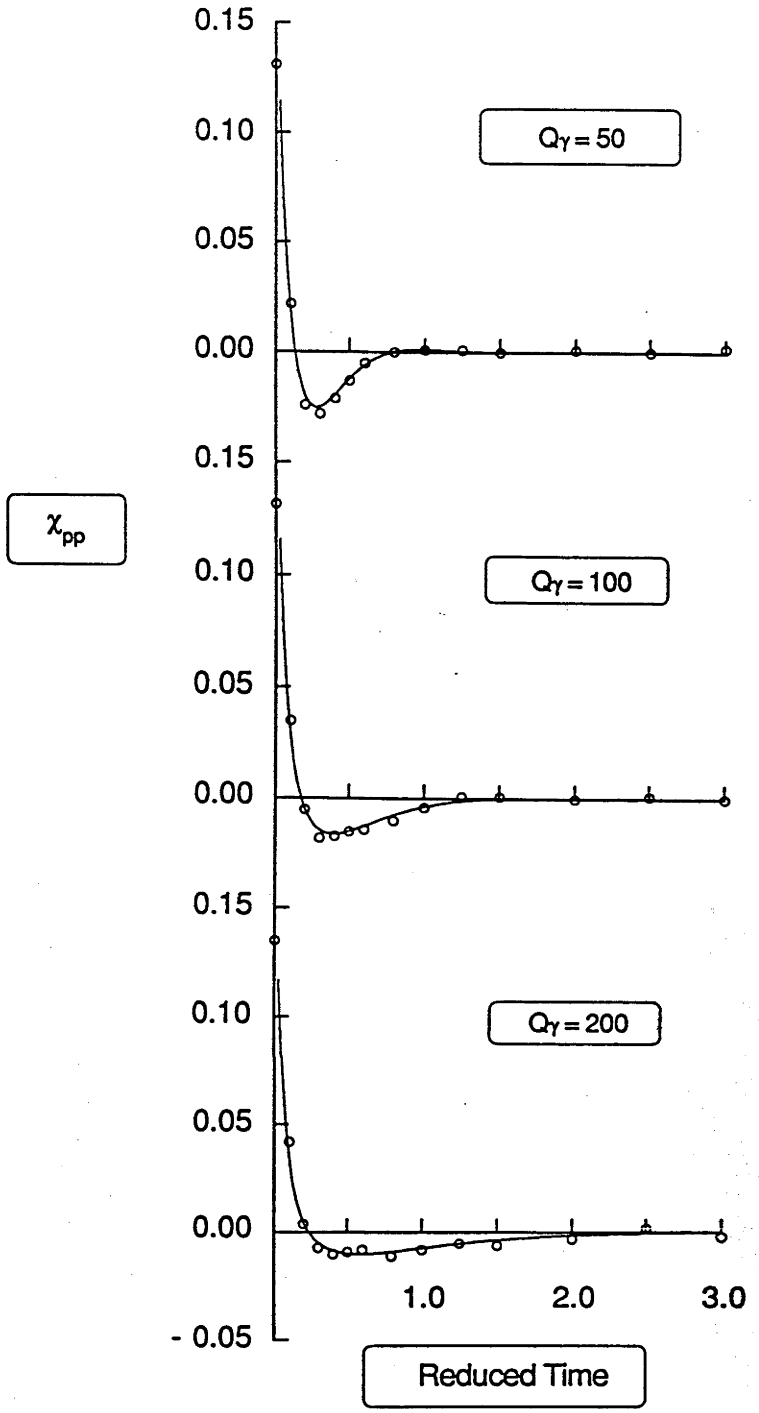


Figure 4.1 shows the stress autocorrelation function for the three values of Q_γ. The circles represent actual data points from simulation. The solid lines are calculated from the Fourier-Laplace transform of equation (4.39). The $t^* = 0$ values of the autocorrelation functions are independent of Q_γ. The the well depth is approximately proportional to $1/Q_\gamma$ and the first x-axis crossing time is roughly proportional to $Q_\gamma^{1/2}$.

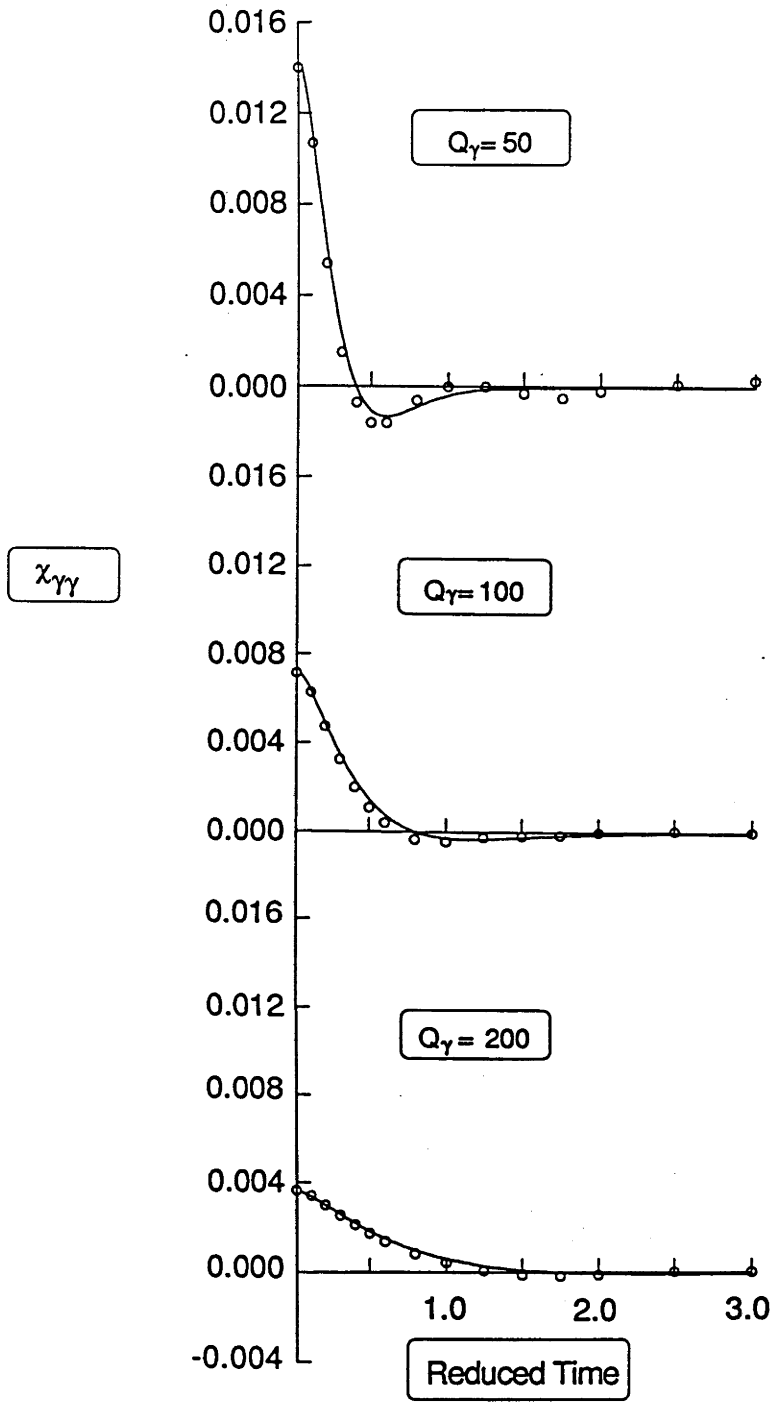


Figure 4.2 shows the strain-rate autocorrelation function for the three values of Q_γ. Again the circles are actual data points from simulation, while the solid lines are calculated from the Fourier-Laplace transform of equation (4.40). The time zero values of the correlation functions are in agreement with equation (4.44). The first axis crossing time is approximately proportional to Q_γ.

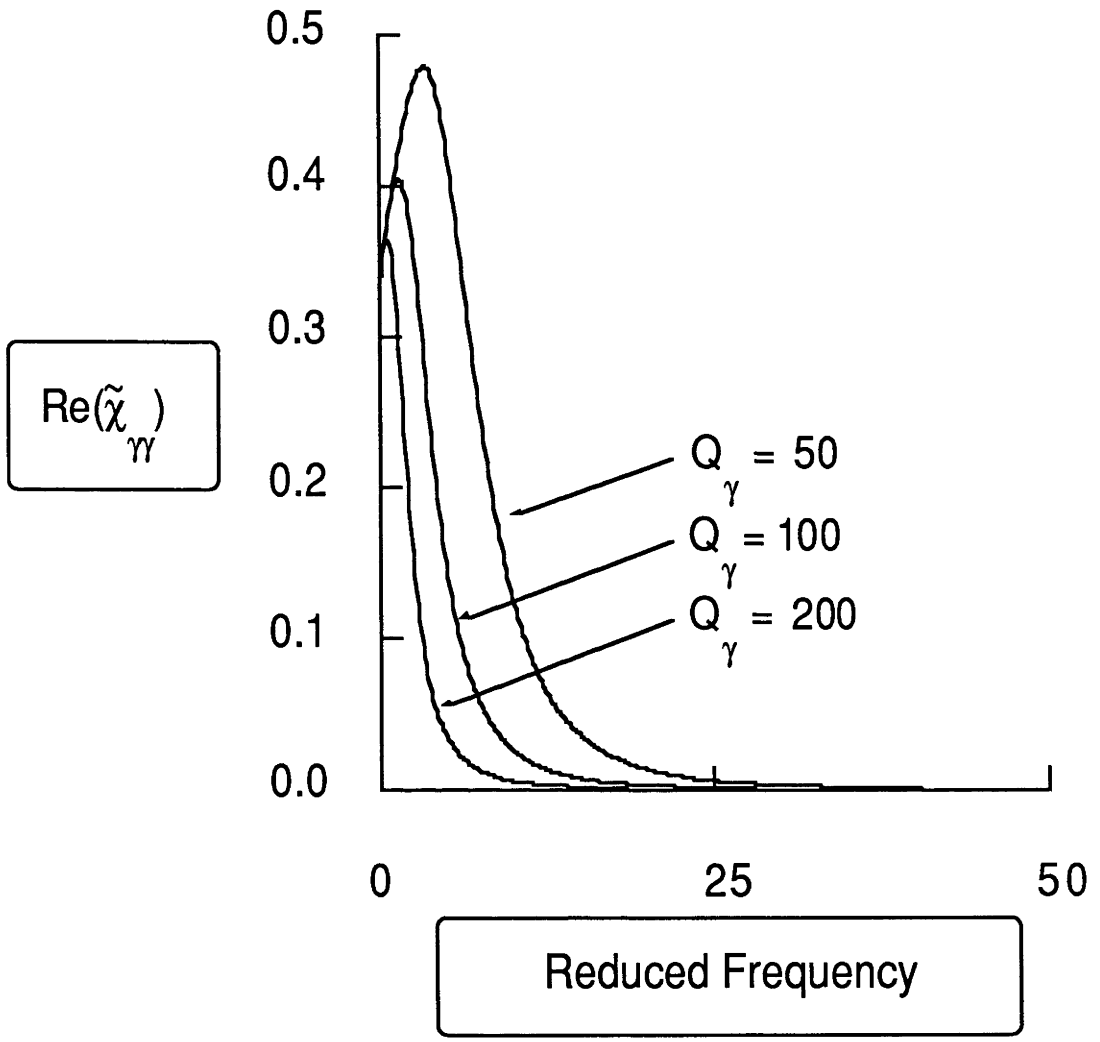


Figure 4.3 shows the Fourier transform of the strain-rate autocorrelation function in the Norton ensemble. At zero frequency, the correlation function is independent of Q_γ and is equal to η^{-1}. All the data displayed in this figure was generated from the interpolation formula of reference 30 together with equation (4.40).

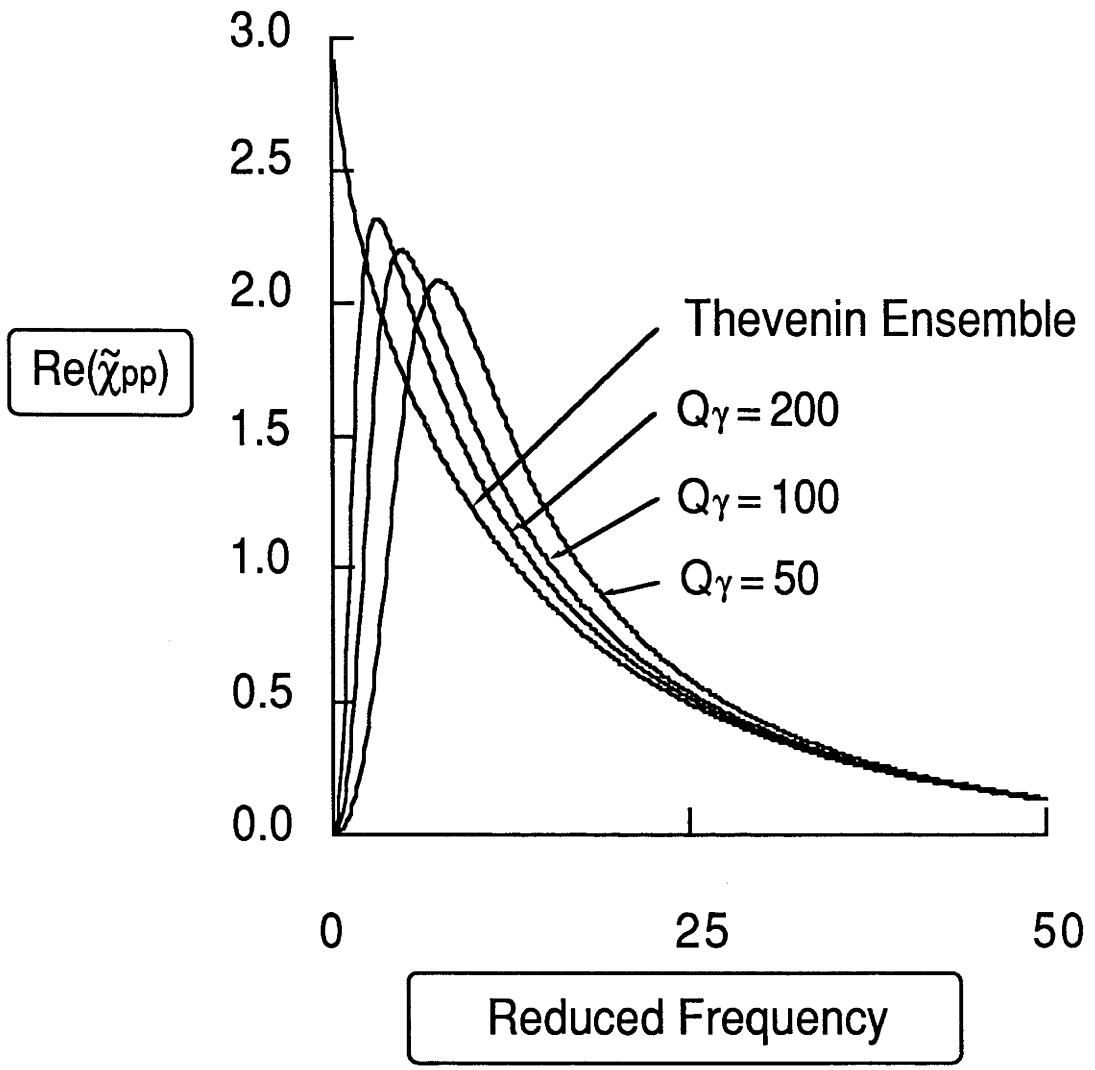


Figure 4.4 gives Norton ensemble stress autocorrelation function in the frequency domain. We also display the same function as calculated in the Thevenin ensemble (ie. the frequency dependent shear viscosity). As Q_γ increases, the Norton ensemble curve follows the Thevenin curve to lower frequencies before decaying to zero. In the limit of infinite Q_γ, the Norton curve would be identical to the Thevenin case except at the origin, where the Norton curve would be zero. All the data displayed in this figure were generated from the interpolation formula of reference 30 together with equation (4.39).

4.5 Discussion

The time correlation functions in the Norton ensemble show some interesting properties. The stress autocorrelation function shows constant rise-height, ie $\langle P_{xy}^2 \rangle$, a well-depth inversely proportional to Q_γ, and an initial axis-crossing time proportional to $Q_\gamma^{1/2}$. We have no simple theoretical justification for these dependencies. The strain-rate autocorrelation function shows that $\langle \gamma^2 \rangle$ is inversely proportional to Q_γ, and that the axis-crossing time is proportional to Q_γ, while the well depth appears to have no simple dependence on Q_γ. Evans and Ely showed that

$$\langle \gamma^2 \rangle_{S_{xy}=0} = \frac{kT}{Q_\gamma} \tag{4.44}$$

Using this relationship, we can calculate $\langle \gamma^2 \rangle_{S_{xy}=0}$ from the current data. The results are shown in Table 4.3.

We have derived theoretical relationships between the various equilibrium correlation functions (equations 4.32, 4.34 and 4.36). These relationships imply that any one of the correlation functions, $\chi_{pp}(t)$, $\chi_{\gamma\gamma}(t)$ or $\chi_{p\gamma}(t)$, is sufficient to obtain the frequency dependent shear viscosity.

In practice the cross correlation function seems to be so noisy that it is not recommended as a means of computing the viscosity from simulation data. As expected, NEMD seems to be more efficient for calculating viscosities than analysis of any of the relevant equilibrium correlation functions. As a computational technique, our results show that for liquids at least, analysis of either the equilibrium stress or strain rate fluctuations is relatively inefficient. The Norton ensemble generalization of the Green-Kubo method is even less efficient than the familiar Thevenin ensemble Green-Kubo method [30].

Previously, Norton ensemble results for shear viscosity were only available at zero frequency. The only non-zero frequency results were for the colour diffusion process [31]. The present theory is quite general and could be extended to any Navier-Stokes transport coefficient.

Values of $<\gamma^2>_{S_{xy}=0}$

Q_γ	$<\gamma^2>_{calc}$	$<\gamma^2>_{obs}$
50	0.0144	0.0140 ± 0.0004
100	0.0072	0.0071 ± 0.0001
200	0.0036	0.0036 ± 0.0001

Table 4.3 Average values of γ^2 at $S_{xy} = 0$. The calculated values come from equation (4.44).

5. Constant Pressure Ensemble Simulations

The previous chapters have discussed shearing simulations performed at constant density. Physical experiments however are often at constant pressure, and it would be useful to simulate systems under these conditions. In this chapter we discuss two methods for constant pressure simulations together with some results from studies of 2-d and 3-d systems. The section on soft sphere simulations is based on a paper by Hood, Evans and Hanley which has been submitted to the Journal of Statistical Physics.

5.1 Gaussian Isobaric Simulations

The isobaric-isokinetic equations of motion for a system under shear can be written as

$$\dot{q}_i = \frac{p_i}{m} + \dot{\varepsilon} q_i + \gamma i q_{yi}$$

$$\dot{p}_i = F_i - \dot{\varepsilon} p_i - \gamma i p_{yi} - \alpha p_i \qquad (5.1)$$

where α is a thermostatting multiplier which, for the Gaussian isokinetic equations of motion, is given by

$$\alpha = \frac{\sum F_i \cdot p_i - \gamma \sum p_{xi} p_{yi}}{\sum p_i^2} \qquad (5.2)$$

There is no coupling between the pressure constraint and the temperature constraint.

The dilation rate, $d\varepsilon/dt$ acts as a volume control to constrain the pressure and can be evaluated as follows. In three dimensions the hydrostatic pressure is

$$3pV = \sum p_i \cdot p_i/m + \sum q_i \cdot F_i$$

$$= \sum p_i \cdot p_i/m - \frac{1}{2} \sum_{i \neq j} q_{ij} \cdot F_{ij} \qquad (5.3)$$

The time derivative of this is

$$3p\dot{V} + 3\dot{p}V = \sum 2 p_i \cdot \dot{p}_i/m - \frac{1}{2} \sum_{i \neq j} \left(\dot{q}_{ij} \cdot F_{ij} + q_{ij} \cdot \left(\frac{\partial F_{ij}}{\partial q_{ij}} \cdot \dot{q}_{ij} \right) \right) \qquad (5.4)$$

The second term on the left hand side is zero at constant pressure, and the first term on the right hand side is zero at constant temperature. If we have a spherically symmetric potential ϕ, then

$$3p\dot{V} = \frac{-1}{2}\sum_{i\neq j}\left(\dot{\mathbf{q}}_{ij}.\mathbf{F}_{ij} + \mathbf{q}_{ij}.\left(\frac{\partial^2\phi_{ij}}{\partial q_i^2}.\dot{\mathbf{q}}_{ij}\right)\right)$$

$$= \frac{-1}{2}\sum_{i\neq j}\left(\dot{\mathbf{q}}_{ij}.\mathbf{F}_{ij} + \mathbf{q}_{ij}.\left(\frac{\phi'_{ij}}{q_{ij}}\mathbf{I} - \frac{\mathbf{q}_{ij}\mathbf{q}_{ij}}{q_{ij}^2}\left(\frac{\phi'_{ij}}{q_{ij}} - \phi''_{ij}\right)\right).\dot{\mathbf{q}}_{ij}\right) \tag{5.5}$$

which reduces to

$$3p\dot{V} = \frac{-1}{2}\sum_{i\neq j}\left(\dot{\mathbf{q}}_{ij}\cdot\mathbf{F}_{ij} + \phi''_{ij}(\mathbf{q}_{ij}\cdot\dot{\mathbf{q}}_{ij})\right) \tag{5.6}$$

If we now substitute in the equations of motion, we get an expression for the Gaussian isokinetic-isobaric multiplier. Using the fact that,

$$\dot{V} = 3\dot{\varepsilon}V \tag{5.7}$$

and defining

$$\Phi_{ij} \equiv \frac{\phi'_{ij}}{q_{ij}} + \phi''_{ij} \tag{5.8}$$

we get the expression

$$\dot{\varepsilon} = -\frac{\sum_{i\neq j} \Phi_{ij}\left(\mathbf{q}_{ij}\cdot\mathbf{p}_{ij}/m + \gamma q_{ijx}q_{ijy}\right)}{\sum_{i\neq j} \Phi_{ij}q_{ij}^2 + 9pV} \tag{5.9}$$

At zero shear rate the above expression reduces correctly to that given in reference 9. The Gaussian constraint is a differential one, which means that the time derivative of the pressure is zero, rather than the pressure itself being constrained to a particular value. Thus to start the simulation at the desired pressure a Newton-Raphson method that alters the volume until the required pressure is obtained is used. It can be seen from the pressure definition that it would also be possible to rescale momenta, but rescaling the volume is more intuitive and does not cause problems with thermostatting. Because of numerical inaccuracies the pressure and temperature drift slowly with time. To maintain a property at its desired value it is necessary to rescale occasionally, typically every 25 to 100 timesteps, before the drift has become appreciable in size. The pressure is rescaled using the Newton-Raphson technique while the temperature is rescaled using simple velocity rescaling. The Newton-Raphson scheme, although simple and efficient, is not particularly robust. It is quite possible for the procedure to diverge or get stuck in an infinite loop if the tolerance set on the pressure is too tight.

5.2 Nosé-Hoover Isobaric Simulations

The difficulties presented by the Newton-Raphson scheme can be avoided by the use of a Nosé-Hoover type barostat. The equation for the feedback variable for constant pressure simulations, the dilation rate, is

$$\dot{\varepsilon} = \frac{1}{Q_p}(p - p_{target}) \tag{5.10}$$

Q_p is the damping constant that affects the fluctuations inherent in the Nosé-Hoover method. As was shown in chapter 4 the precise value of this constant is not important provided it is within a certain range. It was found in practice that Gaussian and Nosé-Hoover methods for constraining the pressure gave the same results. The simulations described in this chapter used the Nosé-Hoover barostat with $Q_p = 10.0$, and a Gaussian PBT thermostat.

5.3 2-d Simulation Results

In chapter 2 it was shown that at constant pressure the density of a system would decrease with increasing shear. The isokinetic-isobaric ensemble allows us to test this prediction under very controlled circumstances. The system studied was 896 soft disks, at a temperature $T^* = 1.0$ and at a pressure $p^* = 10.9$, the equilibrium pressure for the system at the usual density $\rho^* = 0.9238$. Figure 5.1 shows a graph of density against shear rate and shows clearly this prediction.

The rheology literature has long confused shear thinning and shear dilatancy for some time. Shear thinning is the reduction in viscosity with shear rate while shear dilatancy is the decrease in density with shear rate. In figure 5.2 we graph the viscosities for two sets of simulations; one at a constant density $\rho^* = 0.9238$ and the other at a constant pressure $p^* = 10.9$. The isobaric viscosity falls away much quicker than the isochoric one indicating that at constant pressure a significant component of the shear thinning is due to the shear dilatancy. This is the first time that such a distinction has been clearly observable.

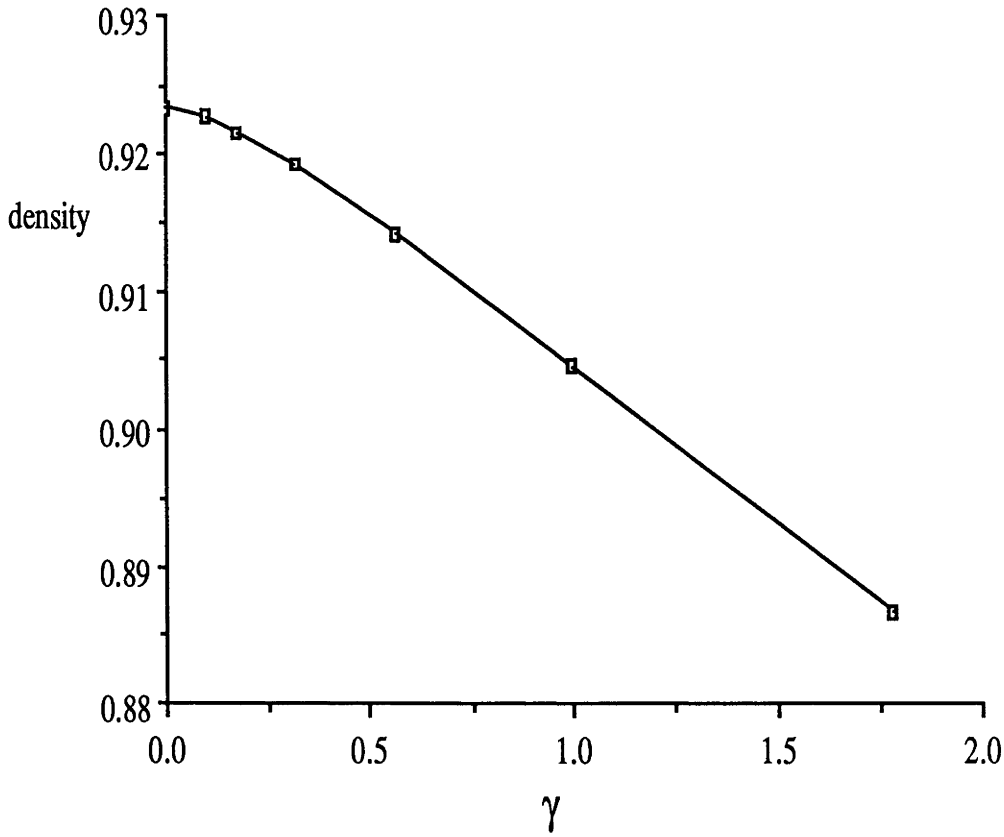


Figure 5.1 Density as a function of shear rate for the soft disk system at constant pressure.

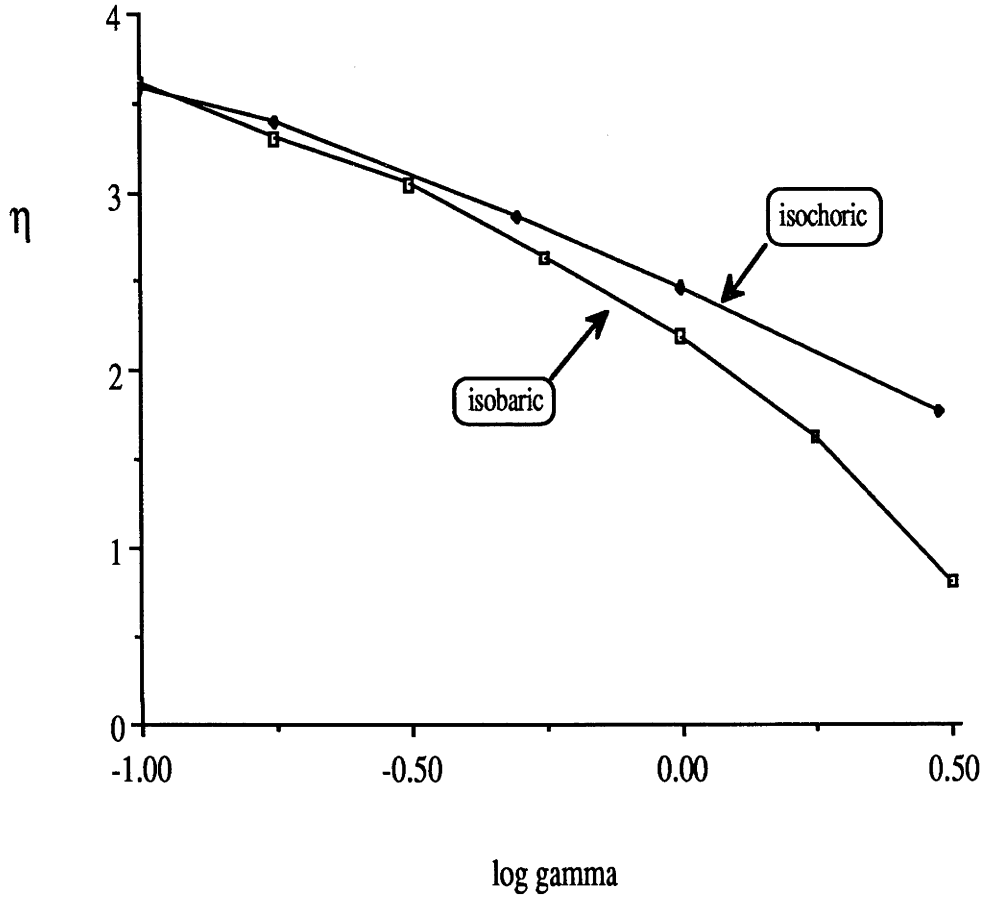


Figure 5.2 Comparison of isobaric and isochoric viscosities for the soft disk system.

5.4 Soft Sphere Simulations

A large body of data has been collected for the rheological properties of the soft sphere system over the past decade. In collaboration with Hanley at the National Institute of Standards and Technology we have produced an extensive set of rheological data for the soft sphere system at constant density and at constant pressure. The following section describe the properties studied and the results. Many authors have used NEMD to show that a simple fluid is not simple when subjected to shear and has rheological properties more usually associated with complex fluids such as polymers [33,34,35,36,37]. The principle results of this section are descriptions of the strain rate dependence of the viscosities and pressures in the isochoric-isokinetic ensemble, and of the viscosities and densities in the isobaric-isokinetic ensemble. The isokinetic-isobaric results are new [38,39].

5.4.1 Definitions

The definitions used are as for earlier work of Hanley and Hess [40,41]. The shear viscosity is evaluated from the constitutive relation

$$P_{xy} = - \eta_+ (\gamma)\gamma \tag{5.12}$$

Viscosity coefficients are defined to account for normal pressure differences

$$\eta_- (\gamma)\gamma = - [P_{xx} - P_{yy}] \tag{5.13}$$

and

$$\eta_0(\gamma)\gamma = - [P_{zz} - (1/2)(P_{xx} + P_{yy})] \tag{5.14}$$

A state point is defined via the scaling, density and temperature variables as $x = \rho\sigma^3(\varepsilon/kT)^{1/4}$, where σ and ε are the usual potential parameters and ρ is the density. The potential parameters and the mass, m, and the temperature are set to unity, hence $x=\rho$. The pressure, p, and shear viscosity, η_+, are evaluated as a function of density and shear rate, γ, in the isokinetic-isochoric ensemble; and the density and shear viscosity are evaluated as a function of pressure and shear-rate in the isokinetic-isobaric ensemble.

5.4.2 Simulation Details

The system studied consisted of 256 soft spheres interacting with an inverse twelve potential truncated at $r_c = 1.5\sigma$. Runs were carried out with a reduced timestep $\Delta t^* = 0.004$ for the smaller shear rates, $\gamma \leq 1.0$, and $\Delta t^* = 0.002$ otherwise. The runs varied in length from 2 million timesteps for the larger shear rates to 25 million timesteps for the smallest shear rate studied, $\gamma = 0.0625$. Calculations were carried out for reduced densities in the range $0.8485 < \rho^* < 1.0607$, where the freezing density of the 256 particle system is 1.15, and for the equivalent reduced pressures, $8.143 < p^* < 17.0$. Data were taken for the system at equilibrium ($\gamma = 0$), and in the range $0.64 < \gamma < 1.55$. The shear-rate range was extended at the density of 0.9899 since this particular state point corresponded to that selected in previous work [40,41].

5.4.3 Pressure-Density Results

Figure 5.3 plots the pressure as a function of shear-rate at constant density. One observes the $\gamma^{3/2}$ dependence as reported in earlier work [36,41]. Figure 5.4 shows the corresponding curves in the isobaric ensemble: again the $\gamma^{3/2}$ dependence is clear as expected. We estimate the data to be precise to 0.01 percent. The data was taken over the same range of state points as reported previously [41,42,43]. The results were fitted to simple polynomials consistent with the smoothing equations in reference 44;

$$p_{eq} = \rho T*(1.0 + 3.31769 *\rho - 0.685244*\rho^2 + 10.131153*\rho^3) \quad (5.15)$$

$$p_\gamma = \rho T*(0.092405*\rho + 0.074186 *\rho^2 + 0.432951*\rho^3) \quad (5.16)$$

$$p = p_{eq} + p_\gamma *\gamma^{3/2} \quad (5.17)$$

Expressions for the density as a function of pressure are given in section 5.4.7. Note that equations (5.15) and (5.16), and the equations in the section 5.4.7, are merely fitting functions; they extrapolate correctly to zero density but are not exact for the moderately dense gas. A detailed discussion on the second and third virial coefficients of the soft sphere gas is given by Rainwater [45].

5.4.4 Shear Viscosity Results

Figures 5.5 and 5.6 show the variation of the shear viscosity with shear-rate at constant density and pressure, respectively. We have assumed a $\gamma^{1/2}$ dependence as in most previous studies, although the constant density plot, in particular, is not entirely consistent with this assumption over all γ. In view of the errors in the data, assessed as 0.5 percent, and a consistency check from the expression

$$(\partial\eta_+/\partial\gamma)_{p,T} = (\partial\eta_+/\partial\rho)_{\gamma,T} (\partial\rho/\partial\gamma)_{p,T} + (\partial\eta_+/\partial\gamma)_{\rho,T} \quad (5.18)$$

the $\gamma^{1/2}$ dependence is justified. Equation (5.18) is satisfied to within 12%, or better. The isochoric data were compared to previous results of Ashhurst and Hoover [46] Hess and Hanley [41], and other authors [37]. The Newtonian viscosity at zero-γ agreed well, but the $\gamma^{1/2}$ dependence of our data was slightly weaker than that observed previously. Using data from the smoothing equations in Reference 44 to ensure a well behaved function at the

lower densities, the zero-γ viscosity is represented by the expression,

$$\eta(\gamma=0) = 0.171 + 0.025604*[\ \exp(46.579*\rho) - 1.0\ ] \qquad (5.19)$$

Equation (5.19) is proposed only to give a fit of the data and does not give the correct moderately dense behavior. The second virial for the viscosity has been evaluated by Rainwater [47]. The shear-rate dependence is represented by simple equations given in section 5.4.7.

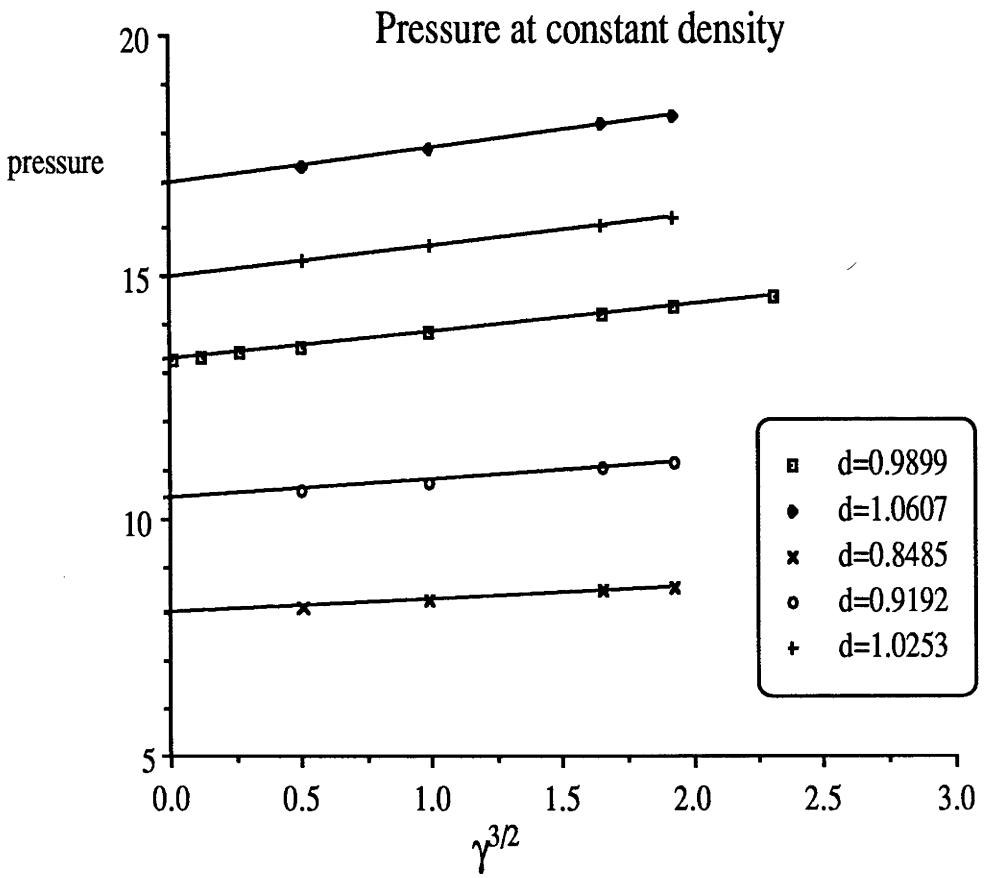


Figure 5.3 Plot of the pressure (p) against shear-rate (γ) for the dense soft sphere liquid at several densities (ρ=d). The freezing density is 1.15. The pressure varies as $\gamma^{3/2}$.

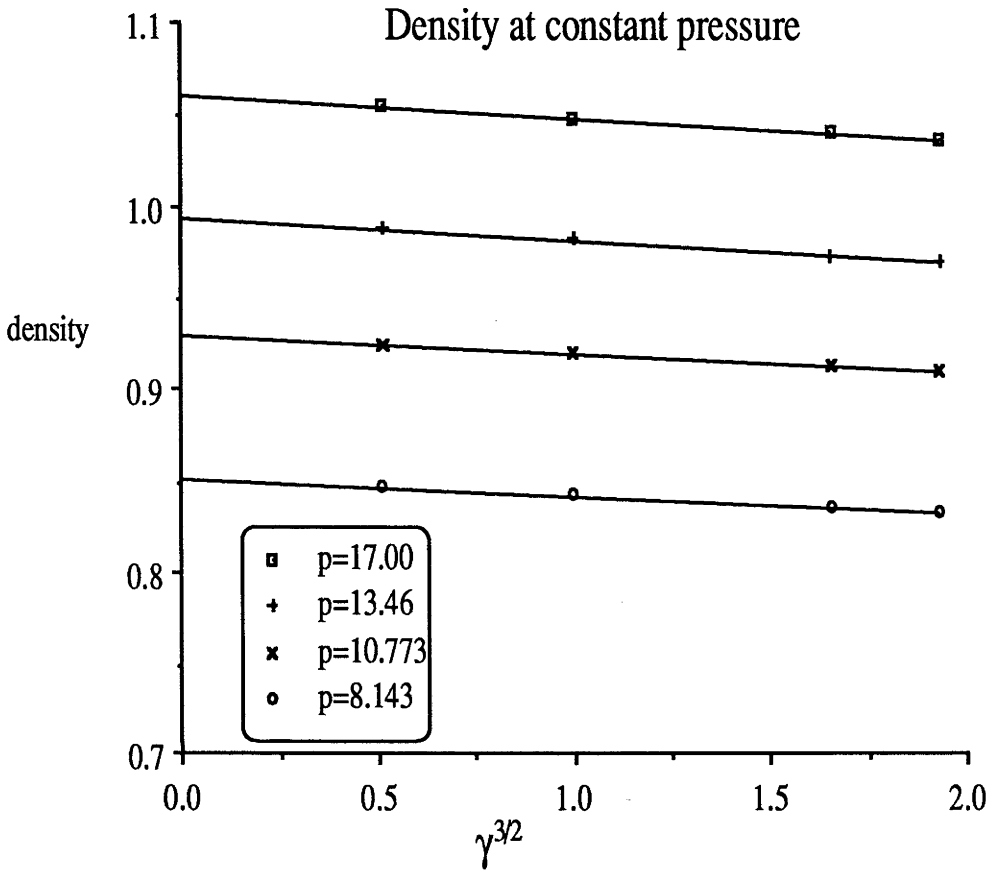


Figure 5.4 Plot corresponding to Figure 1 showing the variation of density with $\gamma^{3/2}$ at constant pressure. Note the derivative $(\partial\rho/\partial\gamma)_p$ is weaker than $(\partial p/\partial\gamma)_\rho$.

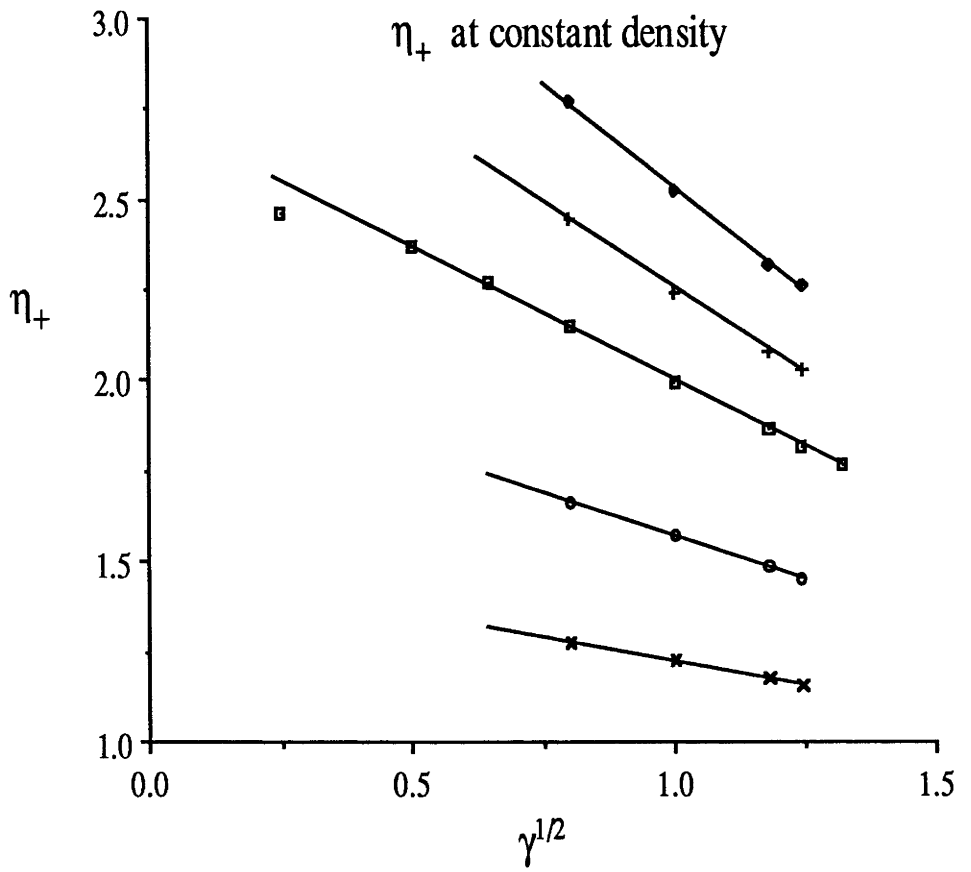


Figure 5.5 Variation of the shear viscosity, η_+ , with $\gamma^{1/2}$ at constant density. Legend as in Figure 5.3. For sufficiently small shear rates the viscosity turns over to a constant value. Thus for small γ the fluid is Newtonian.

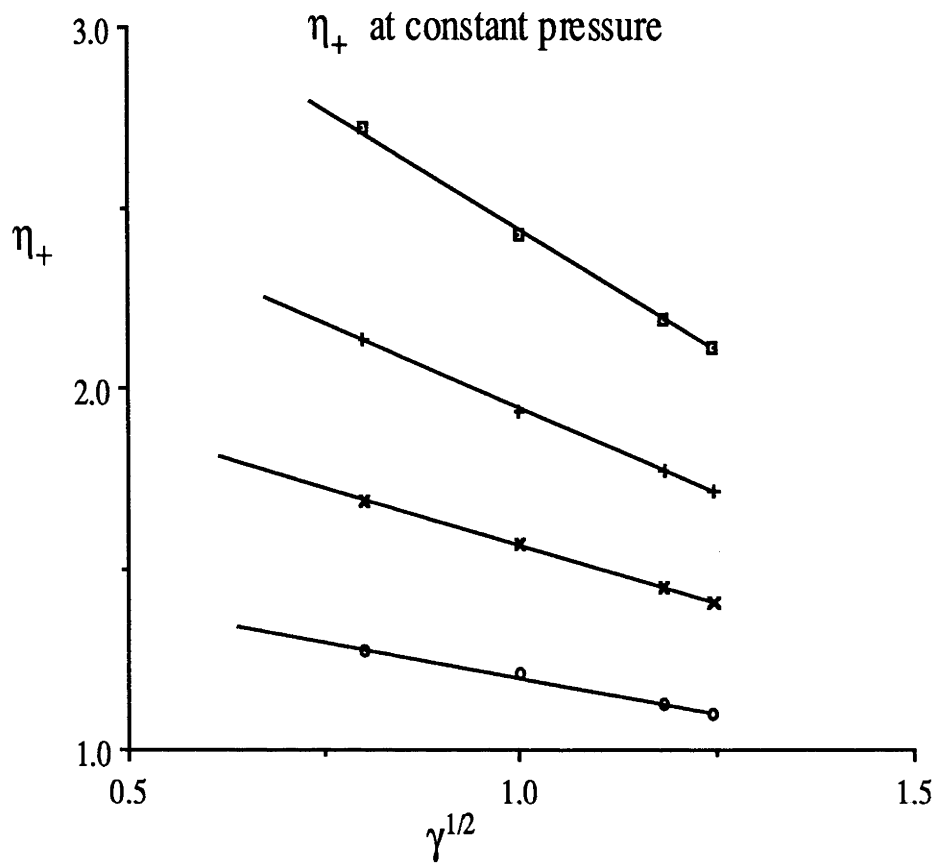


Figure 5.6 Variation of the shear viscosity with $\gamma^{1/2}$ at constant pressure. Legend as in Figure 5.4.

5.4.5 Normal Stress Differences

The coefficients defined by equations.(5.13) and (5.14) are displayed as Figures 5.7 - 5.10. They are well behaved functions of the shear-rate, but the low-γ behavior is uncertain. We have found repeatable negative values at low-γ for the 256 particle system (and for the 56 and 108 particle liquids) but it is unclear if corresponding negative values would be found for a larger system. Excluding these points, we estimate the uncertainty in the coefficients to be 6 and 3 percent for η_- and η_0, respectively. Our expressions for the coefficients are given in section 5.4.7. It is seen that η_- is a **very** weak function of the shear-rate, excluding the low-γ points.

5.4.6 Remarks

We remark that having numerical thermophysical property information is essential if any physical problem is to be understood. Of relevance here, for example, the statistical mechanical thermodynamics and transport properties of a non-equilibrium system has been a topic of interest [35] for some time but many investigations tended to be only formal and could not be verified until recently [47]. Also, as Rainwater *et al* [48] have pointed out, if the shear-rate dependent properties of a system are known *a priori* , one can approach fluid dynamics and rheology directly. The constant pressure data here completes the description of the shear behaviour of the soft sphere fluid. The differences between the isobaric and isochoric ensembles are significant.

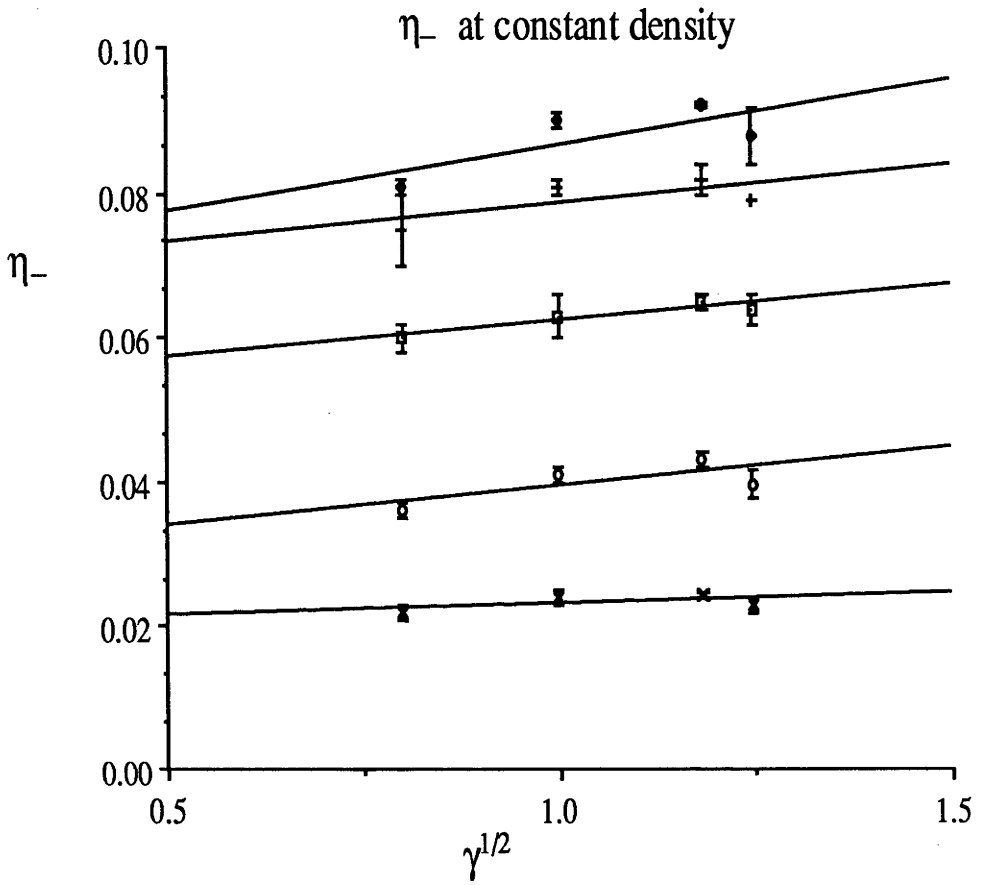


Figure 5.7 Variation of the normal pressure difference coefficient, η_- , with $\gamma^{1/2}$ at constant density. Legend as in Figure 5.3.

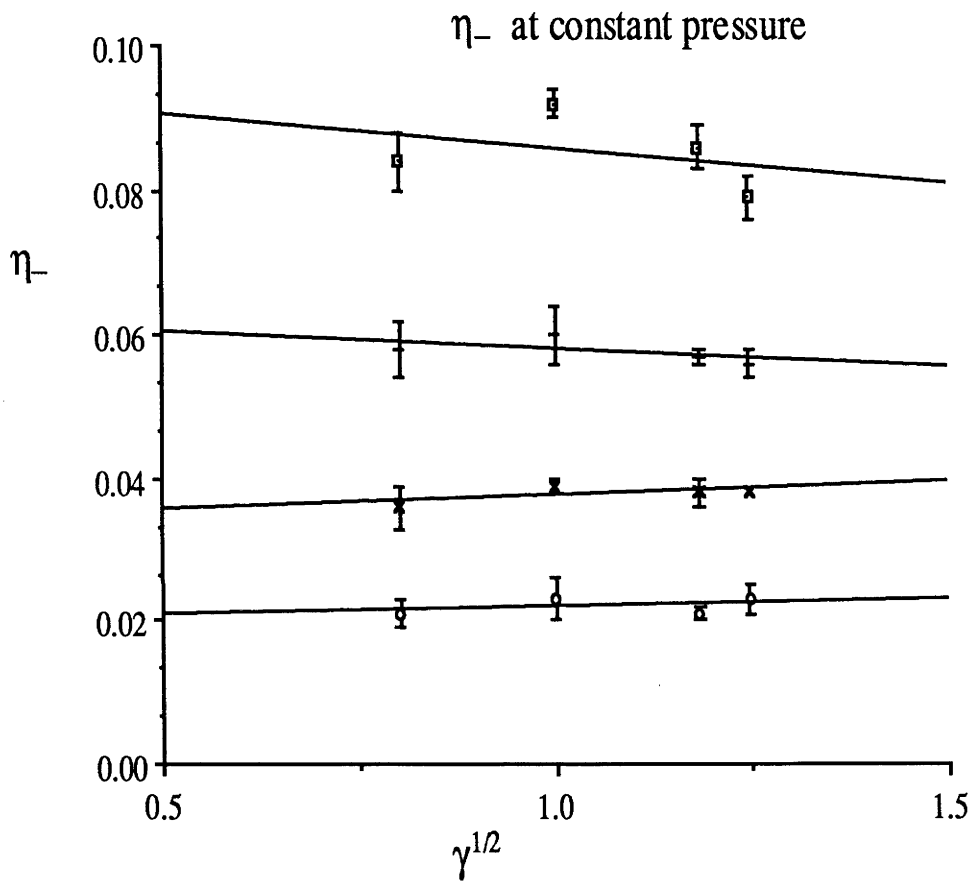


Figure 5.8 Variation of η_- with $\gamma^{1/2}$ at constant pressure. Legend as in Figure 5.3.

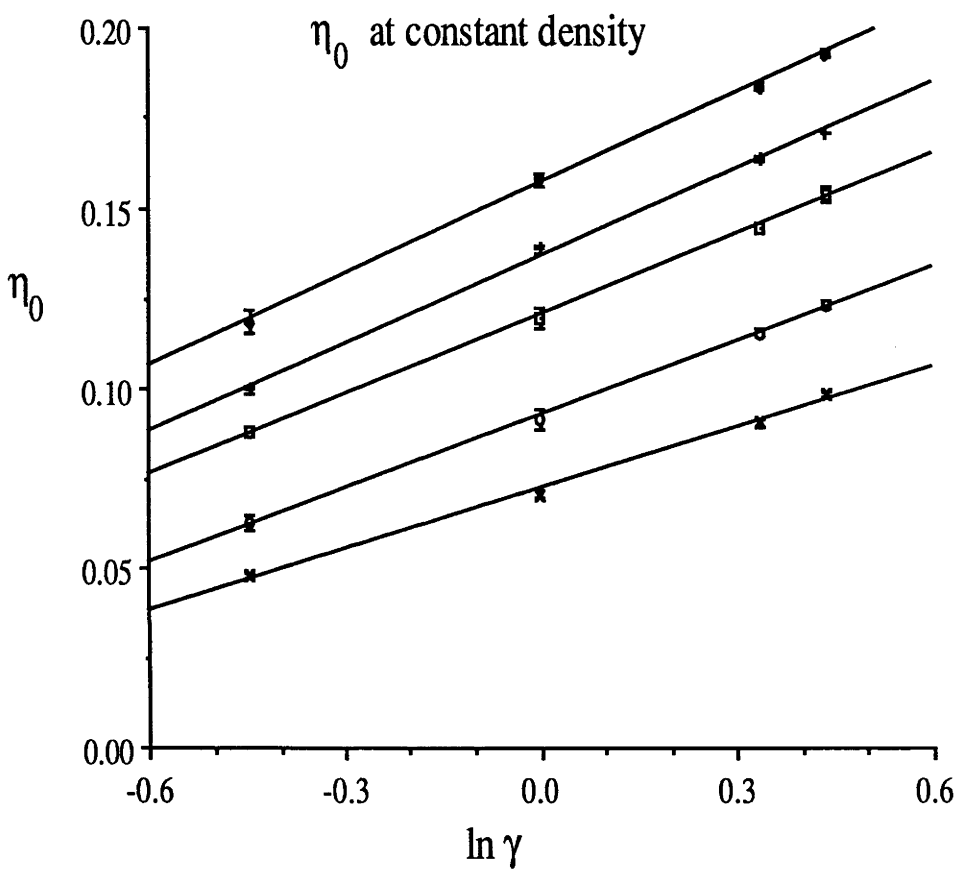


Figure 5.9 Variation of the normal pressure difference coefficient, η_0 , with ln γ at constant density. Legend as in Figure 5.3.

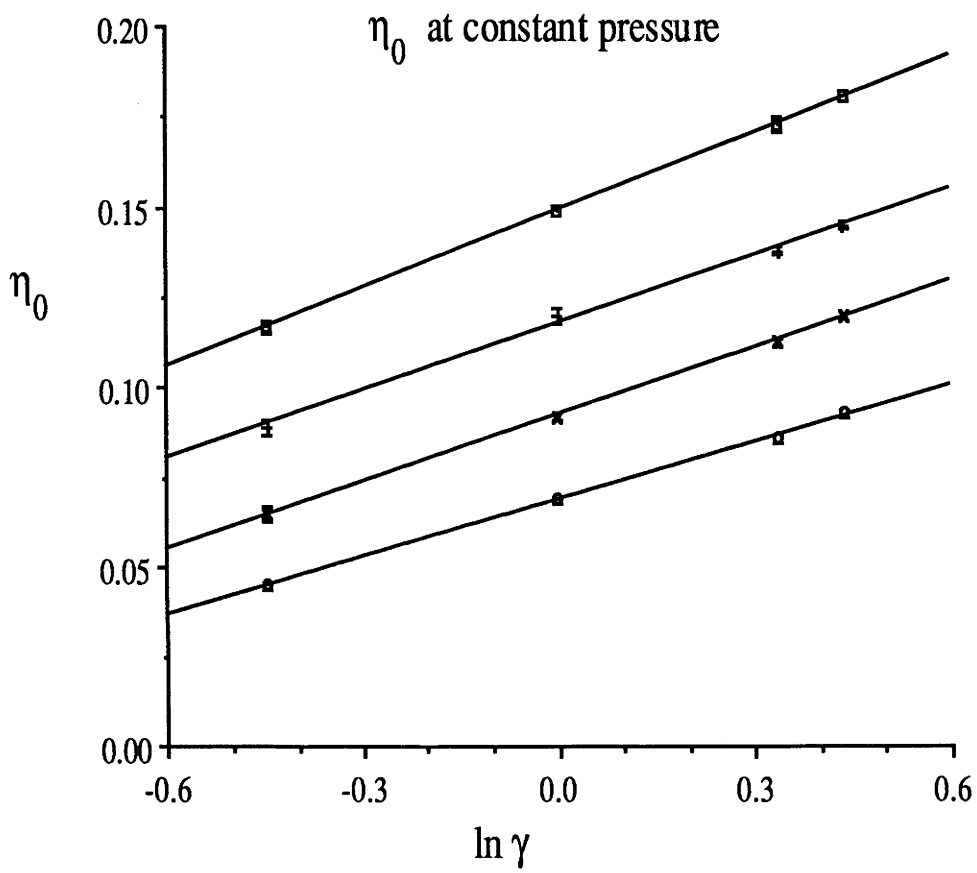


Figure 5.10 Variation of η_0 with $\ln \gamma$ at constant pressure. Legend as in Figure 5.4.

5.4.7 Numerically Fitted Equations

This section describes simple fitting equations of the soft sphere properties obtained from the simulation. The equations are empirical. All variables are in reduced units with the reducing parameters set equal to one.

Density-pressure

$p > 8.143$ and $0.64 < \gamma < 1.55$

$$\rho_{eq} = 0.54561 + 0.44763e\text{-}1 * p - 0.84681e\text{-}3 * p^2$$
$$\rho_\gamma = 0.1309e\text{-}2 + 0.13413e\text{-}2 * p - 0.399e\text{-}4 * p^2$$
$$\rho = \rho_{eq} - \rho_\gamma * \gamma^{3/2} \qquad (5.20)$$

Shear viscosity

isochoric ensemble, $\rho > 0.8485$ and $0.64 < \gamma < 1.55$

$$\eta^+_0 = 0.11738e+2 - 0.70755e+1 * \gamma^{1/2}$$
$$\eta^+_1 = -0.29903e+2 + 0.17672e+2 * \gamma^{1/2}$$
$$\eta^+_2 = 0.2103e+2 - 0.11390e+2 * \gamma^{1/2}$$
$$\eta_+ = \eta^+_0 + \eta^+_1 * \rho + \eta^+_2 * \rho^2 \qquad (5.21)$$

isobaric ensemble, $p > 8.143$, $0.64 < \gamma < 1.55$

$$\eta p^+_0 = -0.46187 + 0.51493 * \gamma^{1/2}$$
$$\eta p^+_1 = 0.25067 - 0.11014 * \gamma^{1/2}$$
$$\eta_+ = \eta p^+_0 + \eta p^+_1 * p \qquad (5.22)$$

Viscometric functions,

η_0, same experimental range as for the shear viscosity.

isochoric ensemble

$$\eta^0_1 = 0.40169 - 0.10138e+1*\rho + 0.73771 *\rho^2$$
$$\eta^0_2 = - 0.3943e-1 + 0.11376 *\rho$$
$$\eta_0 = \eta^0_1 + \eta^0_2 *\ln \gamma \qquad (5.23)$$

isobaric ensemble

$$\eta^0_1 = - 0.5152e-2 + 0.90903e-2 *p$$
$$\eta^0_2 = 0.3869e-1 + 0.19342e-2 *p$$
$$\eta_0 = \eta^0_1 + \eta^0_2 *\ln \gamma \qquad (5.24)$$

η_-, same experimental range

isochoric ensemble

$$\eta^-_0 = 0.56677e-1 + 0.32401e-1 *\gamma^{1/2}$$
$$\eta^-_1 = - 0.28629 - 0.10419 *\gamma^{1/2}$$
$$\eta^-_2 = 0.28446 + 0. 83314e-1 * \gamma^{1/2}$$
$$\eta_- = \eta^-_0 + \eta^-_1 *\rho + \eta^-_2 *\rho^2 \qquad (5.25)$$

isobaric ensemble

$$\eta^-_0 = -0.556e-1 + 0.16509e-1 \gamma^{1/2}$$
$$\eta^-_1 = 0.82843e-2 - 0.94301e-3 * \gamma^{1/2}$$
$$\eta_- = \eta^-_0 + \eta^-_1 *p \qquad (5.26)$$

5.5 A Remark on the Constant Pressure Ensemble - Relaxation Times

The differences between the isobaric and isochoric ensembles only begin to show for quite large shear rates. Thus one could argue that there is no need to be concerned about the differences because they only show up for extraordinarily large shear rates, for Argon $\gamma^* = 1.0$ corresponds to a shear rate of 10^{14} Hz. This argument however neglects an important point raised in chapter 1; that of scaling.

Hanley has argued that the shear rate alone is not sufficient to determine a fluid's behaviour [18]. Instead the product $\tau\gamma$ should be used; τ is the Maxwell relaxation time for the stress autocorrelation function, $\tau = \eta/G$, where G is the shear modulus. Simple fluids such as Argon have τ of the order of 10^{-12} seconds or less, whereas for polymers τ may be greater than 10^2 seconds. Hanley argues that when the product $\tau\gamma$ is greater than about 0.01 non-Newtonian behaviour can be observed for any fluid. Thus for Argon non-Newtonian behaviour is never observed, as we would require a shear rate of 10^{10} Hz. For complex fluids like polymers or supercooled glycerol non-Newtonian behaviour is easily observed.

6. 2-dimensional Shear Flow

Two dimensional simulations have been compared, at least qualitatively, with experimental situations occurring on planes, for example diffusion of gas across a metal surface. Although such an analogy is limited at best, two dimensional simulations are still useful in understanding fluid properties. Calculating the forces in two dimensions is substantially faster than in three because there are less neighbours. Another major advantage of two dimensional systems is the ability to visualize them; we can easily produce snapshots of particle configurations or of flow patterns, such aids to understanding processes are clearly vastly more difficult in three dimensions. A rather more profound reason for studying systems other than three dimensions is that various theories predict different functional forms for various quantities depending on the dimensionality of the system. If the predicted functional forms are actually observed for other dimensional systems then we have an extra degree of confidence in the theory on which the predictions are based. The PUT thermostat work was all derived from studies of two dimensional systems. In the previous chapter a fairly exhaustive set of data for the soft sphere system was described. In this chapter a detailed study of shear flow in two dimensions is given. The soft disk system shows a large dependence on system size.

6.1 Early Work

Earlier work by Evans and others [49,50] had shown a marked difference in behaviour between small systems, $N < 100$ and larger systems $N \geq 896$. In 1980 Evans performed simulations for soft disk systems with $N = 32$, $N = 50$ and $N = 98$ [49]. He found that the shear viscosity had a logarithmic dependence on γ, with a slope related to $1/N$ in such a way that it was possible to extrapolate to an infinite system result. There appeared to be no evidence of a finite limit to the shear viscosity at equilibrium, thus confirming the predictions of some long time tail theories that the Navier-Stokes transport coefficients diverged [51,52,53]. He also found that the fluid exhibited negative shear dilatancy for smaller shear rates, a result in conflict with thermodynamics.

Although the small system results were consistent with each other and extrapolated to an infinite system size, when simulations were performed on larger systems with $N = 896$ and $N = 3584$, quite different results were obtained [54]. Over the range of shear rates studied the small system results showed no deviation from logarithmic behaviour. However the large system showed a distinct turnover; the viscosity was approximately constant for $\gamma < 0.1$. The turnover point also appeared to be very sharp, although the errors in the data did allow some doubt in this interpretation. Evans and Morriss argued that the turnover was caused by thermodynamic instabilities [54]; essentially at the shear rate where Δp was going negative for the small systems, the large systems exhibited this turnover.

Although thermodynamics could provide a stability argument for the existence of the turnover it was unable to provide any insight into the microscopic processes responsible for the turnover. For a variety of reasons, Evans and Morriss believed that the turnover was caused by transverse momentum currents [54]. They constructed a Maxwell demon that clamped the first three of the transverse modes of the momentum field to zero. The effect of this clamping was that the viscosity followed the logarithmic behaviour, or log line, down to the lowest shear rate studied, $\gamma = 0.01$, the same result as for the small systems.

This then represented the current knowledge of small field shear flow in 2 dimensions but several questions remained. Although the N = 896 and N = 3584 results were statistically indistinguishable, is there a system size effect for these larger systems ? When does the transition from large system to small system behaviour occur ? Is the turnover really sharp ? If as was argued, that secondary flows were responsible for the turnover, would the PUT thermostat show anything different ? To answer these questions and to provide an accurate set of results for future reference it was decided to conduct a thorough study of 2 dimensional shear flow for a number of system sizes.

6.2 Simulation Results

The reference system chosen was N = 14336 soft disks, with $\phi = (\sigma/r)^{12}$, and the potential was cut at $r = 1.5\sigma$. The temperature chosen was $T^* = 1.0$ with a fixed density, $\rho = 0.9238$. The Sllod algorithm was used. The N = 14336 results agree with the n = 896 results of Evans and Morriss [54], although in the current study the errors are substantially smaller. To check any system size dependency, some accurate N = 896, N = 57 344 and N = 224 calculations were performed. It was found that there was no statistically significant difference for the systems with $N \geq 896$, but the N = 224 results were slightly different although the same qualitative behaviour is observed. Thus for large systems, with N ~1000 and greater, there is no N-dependence.

The turnover regime is difficult to simulate because fluctuations are large and the system takes a long time to move from logarithmic to turnover behaviour. The early small system simulations of Evans were relatively short runs, and thus it was possible that these early results were wrong and the system did in fact exhibit a turnover if run for a sufficient time. To test this idea an accurate set of data was obtained for N = 56 and N = 32 particle systems. The Sllod algorithm is slightly different to the homogeneous shear algorithm used by Evans and so the results were expected to be slightly different. Also the very strong N-dependence observed for the small systems meant that the N = 50 results of Evans and the N = 56 results here would not be directly comparable.

The small system results obtained agreed closely, at least qualitatively with the earlier work of Evans, and did not exhibit any turnover, even after runs of up to 20 million timesteps. The transition from the large system behaviour to small system behaviour must occur for a system with about 100 particles. An accurate set of data for a 98 and a 72 particle system was obtained. The shear viscosity and shear dilatancy obtained for all state points and system sizes are shown in figure 6.1 and 6.2 and tabulated in table 6.1.

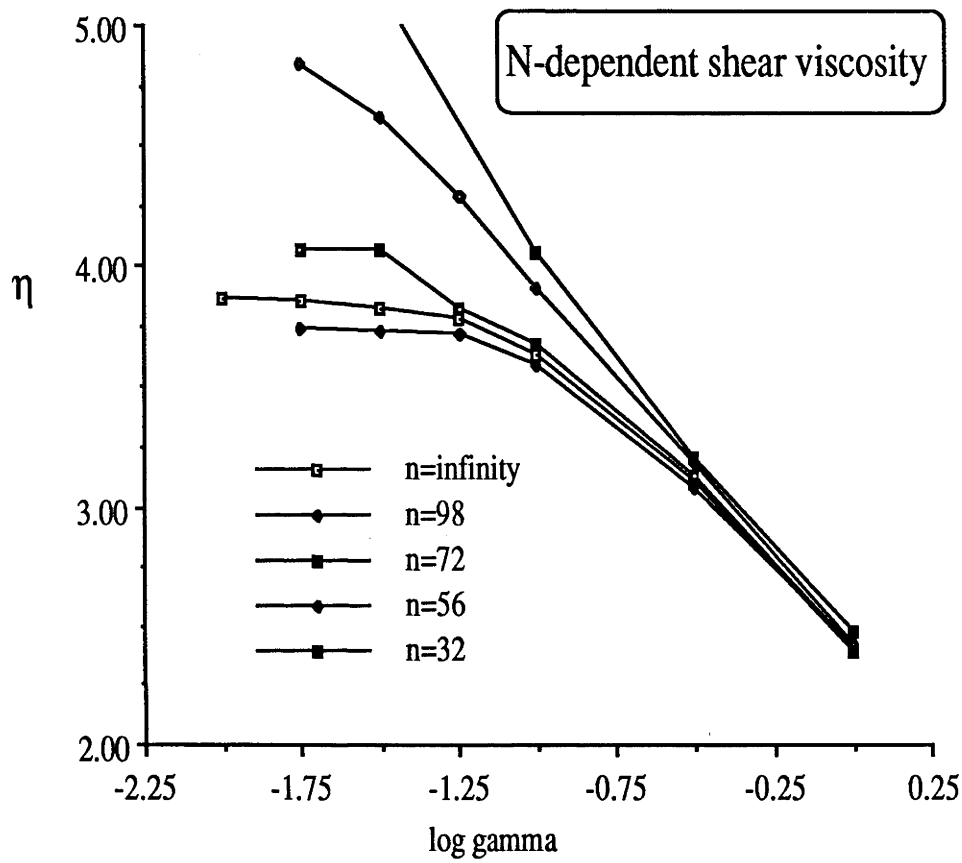


Figure 6.1 Shear viscosity for various system sizes. The N=infinity results are for the N=14336 system. N = 896, 3584 and 57344 all have the same results within statistical error (N = n on the graph).

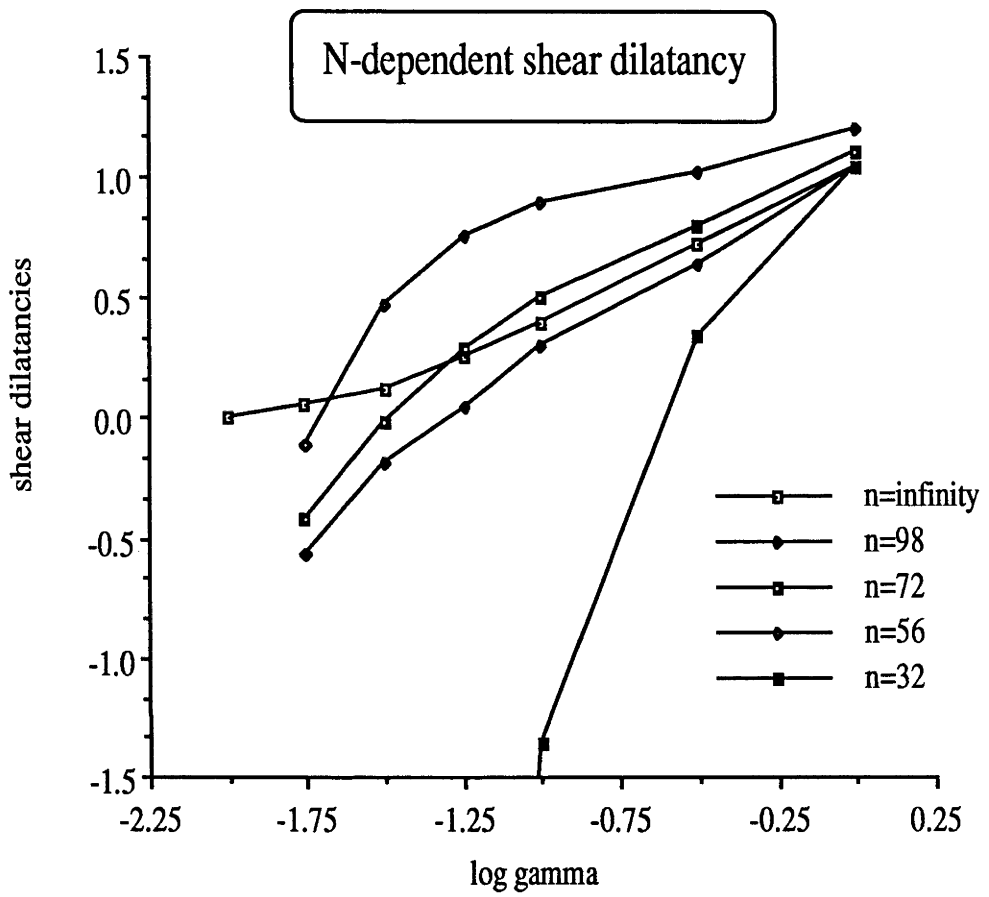


Figure 6.2 Shear dilatancy for various system sizes. The N=infinity results are for the N=14336 system. N = 896, 3584 and 57344 all have the same results within statistical error (N = n on the graph).

Viscosity, $\eta = -P_{xy}/\dot{\gamma}$

	N					
$\dot{\gamma}$	14336	224	98	72	56	32
0.01	3.87±0.03					
0.01778	3.85±0.02		3.74±0.05	4.07±0.03	4.84±0.09	
0.0316	3.82±0.02	3.75±0.06	3.73±0.04	4.07±0.04	4.62±0.05	5.17±0.06
0.057	3.78±0.01		3.72±0.03	3.82±0.05	4.29±0.05	
0.1	3.63±0.02	3.58±0.01	3.59±0.04	3.67±0.02	3.91±0.02	4.06±0.01
0.1778	3.42±0.01	3.41±0.01				
0.316	3.11±0.01	3.12±0.02	3.08±0.01	3.13±0.01	3.19±0.01	3.21±0.02
1.0	2.392±0.003		2.47±0.01	2.41±0.01	2.42±0.01	2.47±0.01

pressure

	N					
$\dot{\gamma}$	14336	224	98	72	56	32
0.0	10.895±0.001	10.913±0.003	10.952±0.001	10.907±0.001	10.844±0.005	11.099±0.001
0.01	10.895±0.000					
0.01778	10.896±0.000		10.942±0.000	10.899±0.001	10.842±0.003	
0.0316	10.899±0.001	10.917±0.002	10.946±0.001	10.906±0.003	10.859±0.002	10.864±0.003
0.057	10.910±0.001		10.955±0.001	10.923±0.001	10.887±0.003	
0.1	10.935±0.000	10.953±0.001	10.982±0.001	10.957±0.001	10.934±0.002	10.963±0.002
0.1778	10.995±0.001	11.009±0.002				
0.316	11.124±0.002	11.133±0.004	11.155±0.001	11.160±0.002	11.168±0.001	11.209±0.002
1.0	11.941±0.001		11.994±0.002	12.013±0.004	12.043±0.002	12.145±0.005

Table 6.1 N-dependent shear viscosity and pressure.

6.3 Discussion of the Results

For large shear rates, $\gamma > 0.1$, the viscosity and shear dilatancy follow a logarithmic behaviour with shear rate, with the exception of the shear dilatancy for N = 56 which follows no simple analytic function. For smaller shear rates the larger system sizes show a turn over in the viscosity to a fairly constant value. This turnover is gradual and not sharp as was suggested by Evans and Morriss. For the larger systems, the turnover in the shear viscosity is matched by a turnover in the shear dilatancy. Essentially where Δp would be going negative if the log line were followed is where the turnover occurs, and so Δp is always positive. The transition from the log line to the turnover is so gradual that the asymptotic logarithmic behaviour does not appear until $\gamma \sim 0.5$. From the graphs it might be argued that the turnover is not exactly flat. However for the existence of the Navier-Stokes transport coefficients the viscosity must eventually turn over to a constant value. The results here are in agreement with the differential transient results of Morriss [55] who showed that the viscosity was constant down to the smallest field studied, $\gamma = 10^{-8}$. Thus it is fair to say that the turnover is flat and that the Navier-Stokes transport coefficients do exist for two dimensional shear flow.

The smaller system size results however are rather different. The N = 98 shear viscosity closely matches that of the larger systems, but the shear dilatancy, which is a non-linear effect does not. Clearly for smaller systems the transverse momentum currents which cause the turnover are of too long a wavelength to fit in the simulation cell commensurate with the periodic images. The N = 98 results suggest that some form of transverse current is formed which causes the viscosity to turn over, but is not sufficient to cause the shear dilatancy to turn over. The N = 72 results also show a slight turnover for the shear viscosity but not the shear dilatancy. It is also significant that for $\gamma = 1.0$ all system sizes show essentially the same results. Clearly any transverse currents are only important for smaller shear rates. The smaller system size results are also unphysical, as thermodynamics predicts that Δp cannot be negative for such systems. For the purposes of

doing simulations, it would appear that 224 particles is enough to obtain the large system behaviour, although numerical values are not exactly the same within statistical error.

The rather dramatic variation of viscosity and shear dilatancy with system size is somewhat surprising when compared to the three dimensional case. Simulations of the Lennard-Jones system in three dimensions by Evans *et al* [56] do show a very slight system size effect but it is far less extreme than is the case here. The characteristic length scale for the simulation is the length of the primitive cell, L. For the two dimensional systems L (α $N^{1/2}$) is substantially larger than L (α $N^{1/3}$) for the corresponding three dimensional system. Why the three dimensional system should be so insensitive to system size when compared with the two dimensional system is not known.

6.4 Flow Patterns in the Simulations

If secondary, transverse momentum currents are the cause of the turnover as was argued by Evans and Morriss, then the PUT thermostatted system should show some differences when compared to the PBT system. To test this hypothesis simulations for 57344 particles using both PBT and PUT thermostats for γ = 0.01, 0.1 and 1.0 were performed. It was found that there was no statistically significant difference in shear viscosity and shear dilatancy between the two systems for any of the shear rates used. Thus the magnitude of the transverse currents must be far less than the thermal motion of the particles in the system.

To show the flow patterns within the simulation, the primitive cell was divided into a 40 x 40 grid and the particle momenta within these cells were time averaged over 2500 timesteps to produce a smooth distribution of fluid element velocities. This method assumes that any flow patterns that form exist for substantially longer than 2500 timesteps and further are relatively stationary within the cell. From the 40 x 40 cell description was produced a 20 x 20 and a 10 x 10 description by simple spatial averaging. The flow patterns, relative to the imposed linear velocity field, for the PBT and PUT thermostats are shown in figures 6.3 and 6.4 respectively. In these figures the length of the arrows have all been set equal, thus only the direction and not the magnitude of the fluid flow is shown. It was found that the flow velocity was typically around 0.1, whereas the thermal velocity was typically 1.0. Thus the kinetic energies involved in the secondary flow patterns are 1% of the thermal energy and explains why the two thermostats show essentially the same results. It is however worth pointing out that the PUT system **does** show more structure in the flow patterns than the PBT system.

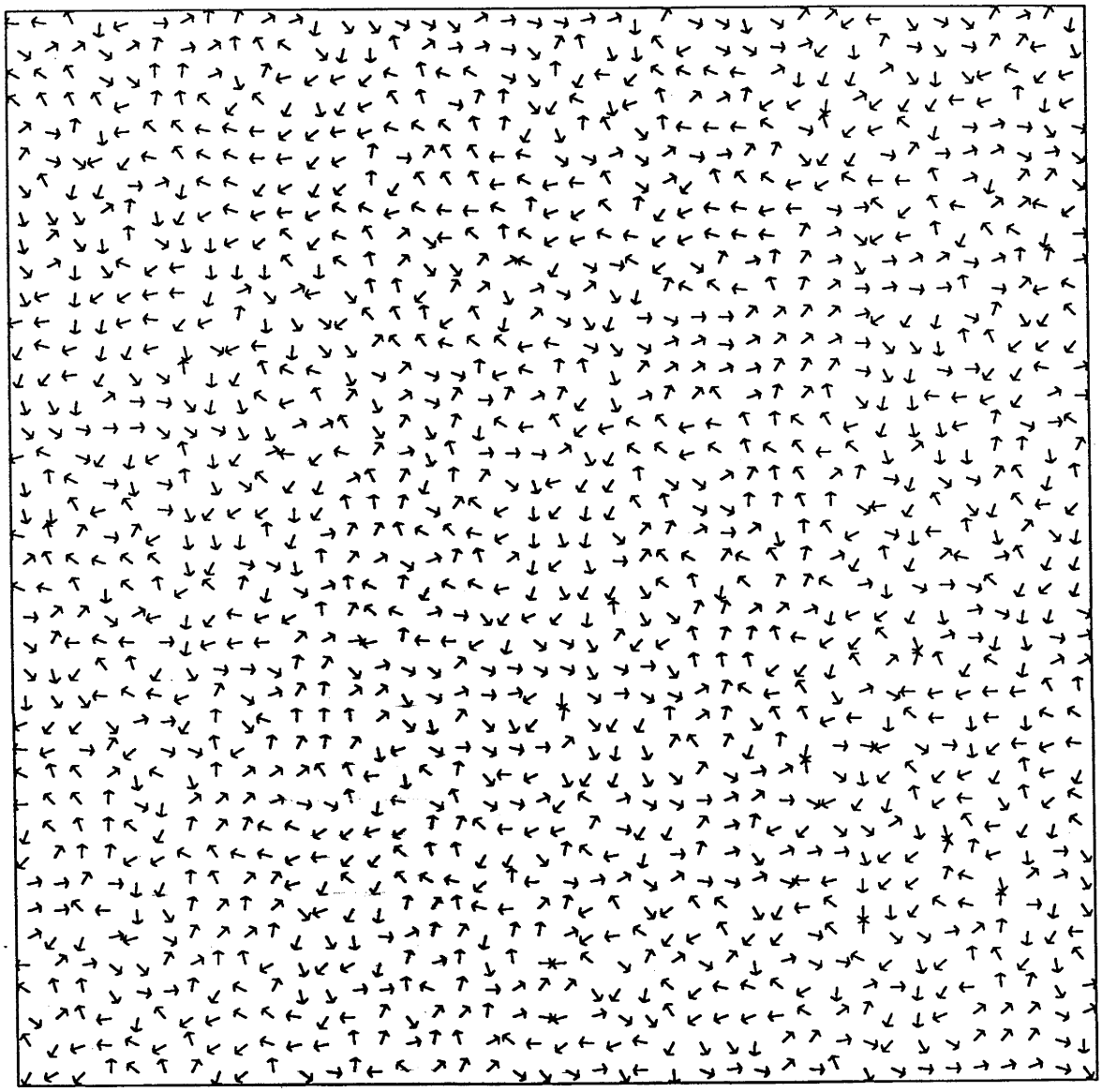


Figure 6.3a Flow pattern for 57 344 particles on a 40 x 40 grid for the PBT thermostat.

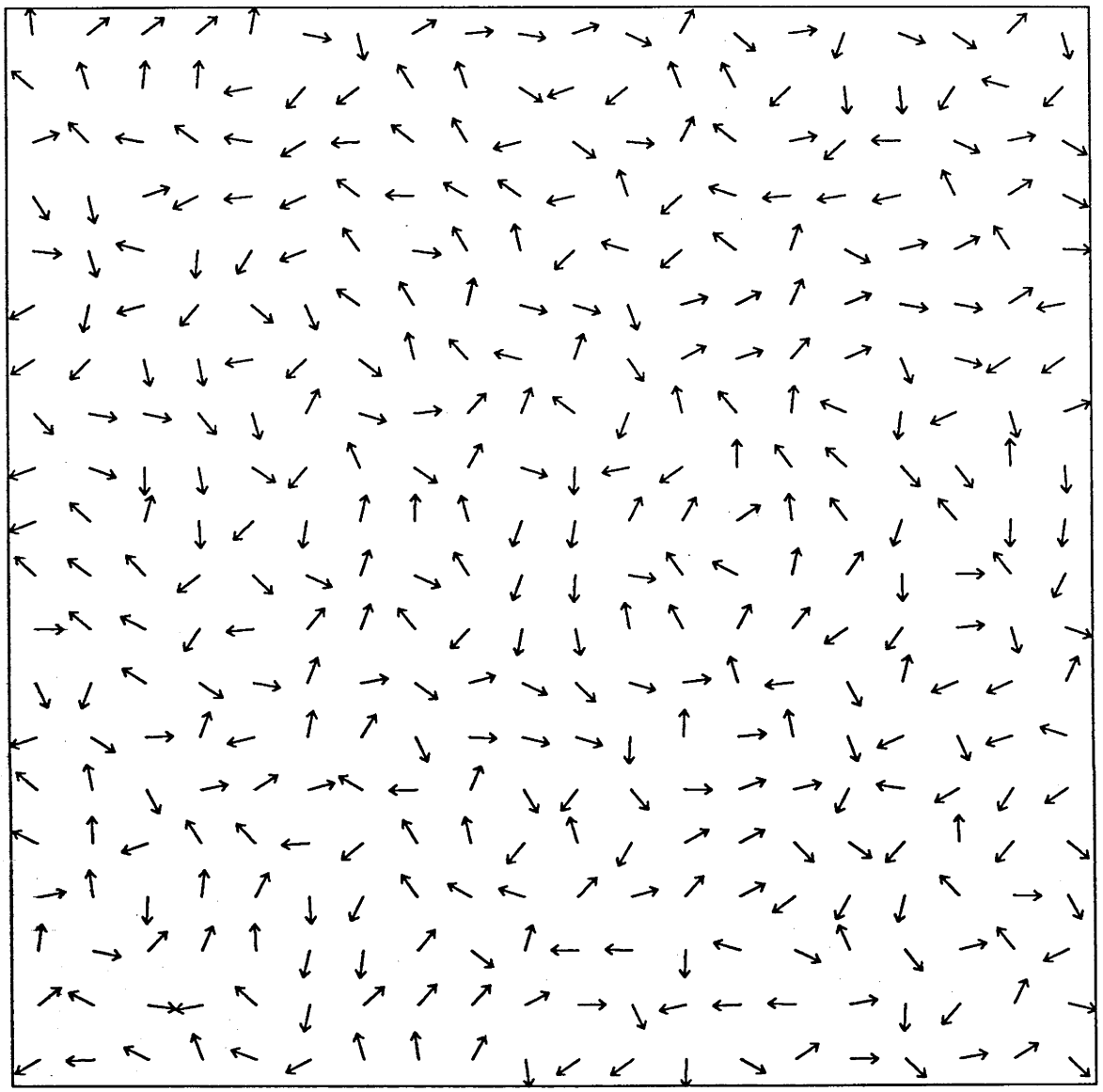


Figure 6.3b Flow pattern for 57 344 particles on a 20 x 20 grid for the PBT thermostat.

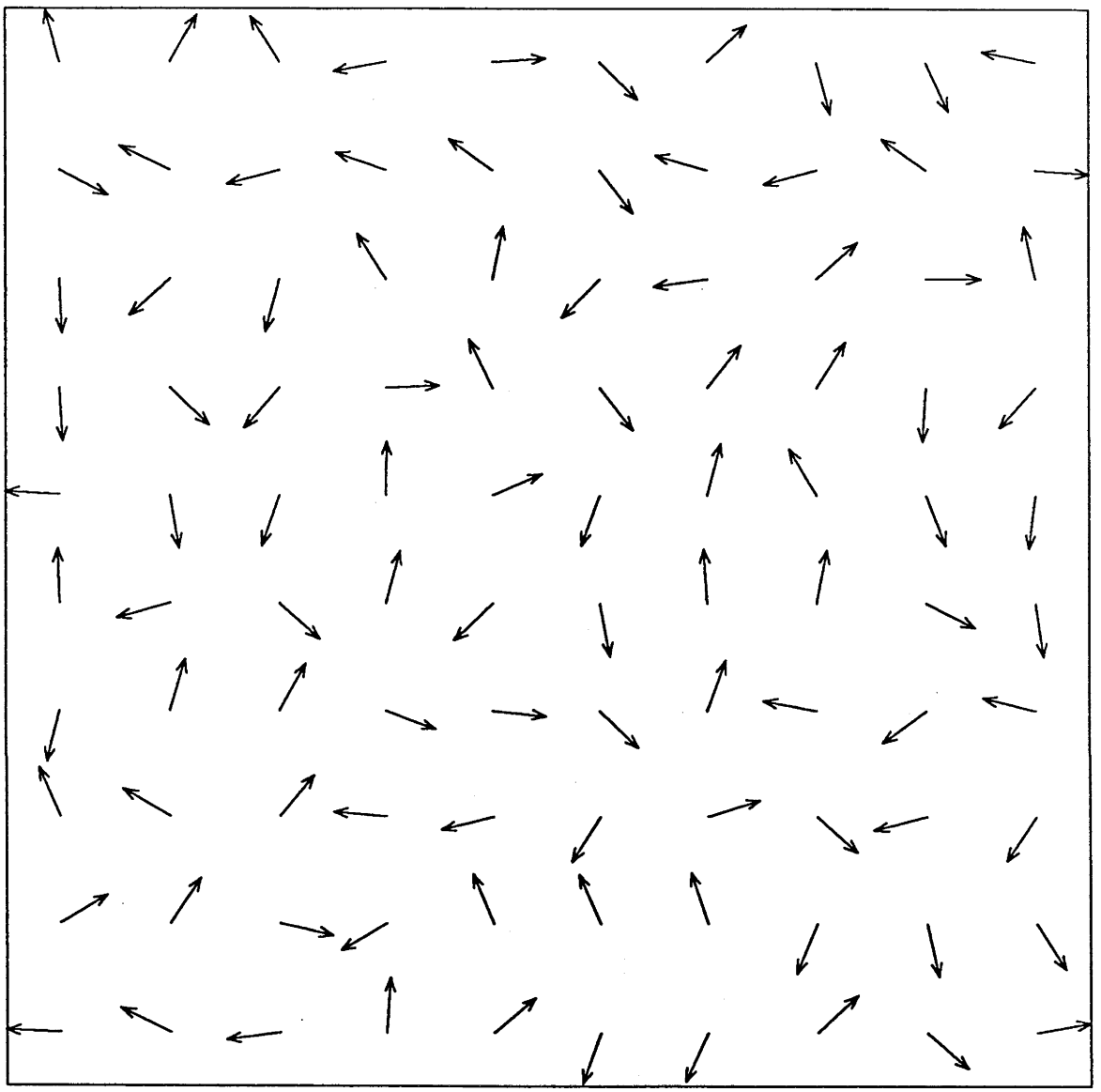


Figure 6.3c Flow pattern for 57 344 particles on a 10 x 10 grid for the PBT thermostat.

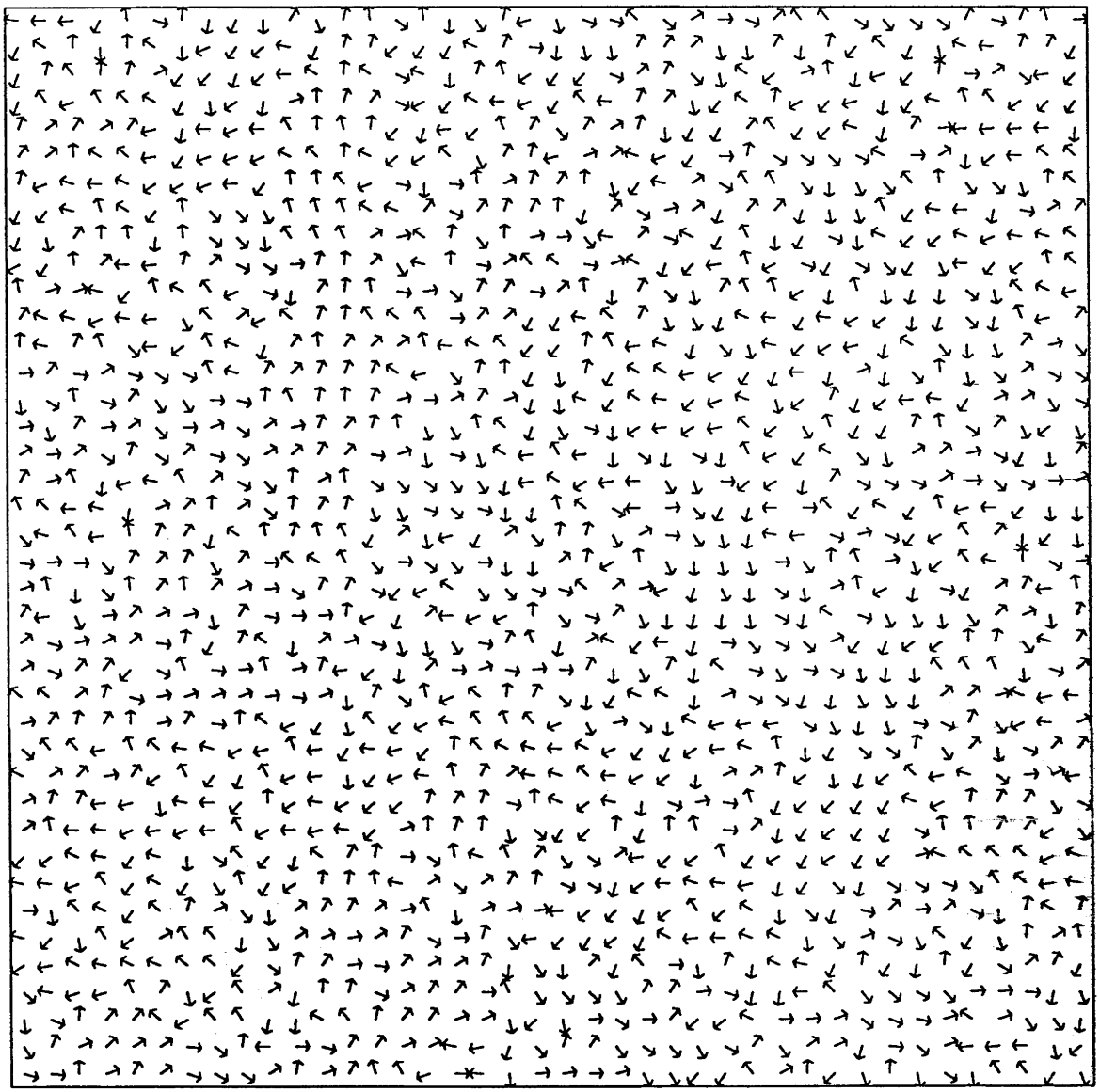


Figure 6.4a Flow pattern for 57 344 particles on a 40 x 40 grid for the PUT thermostat.

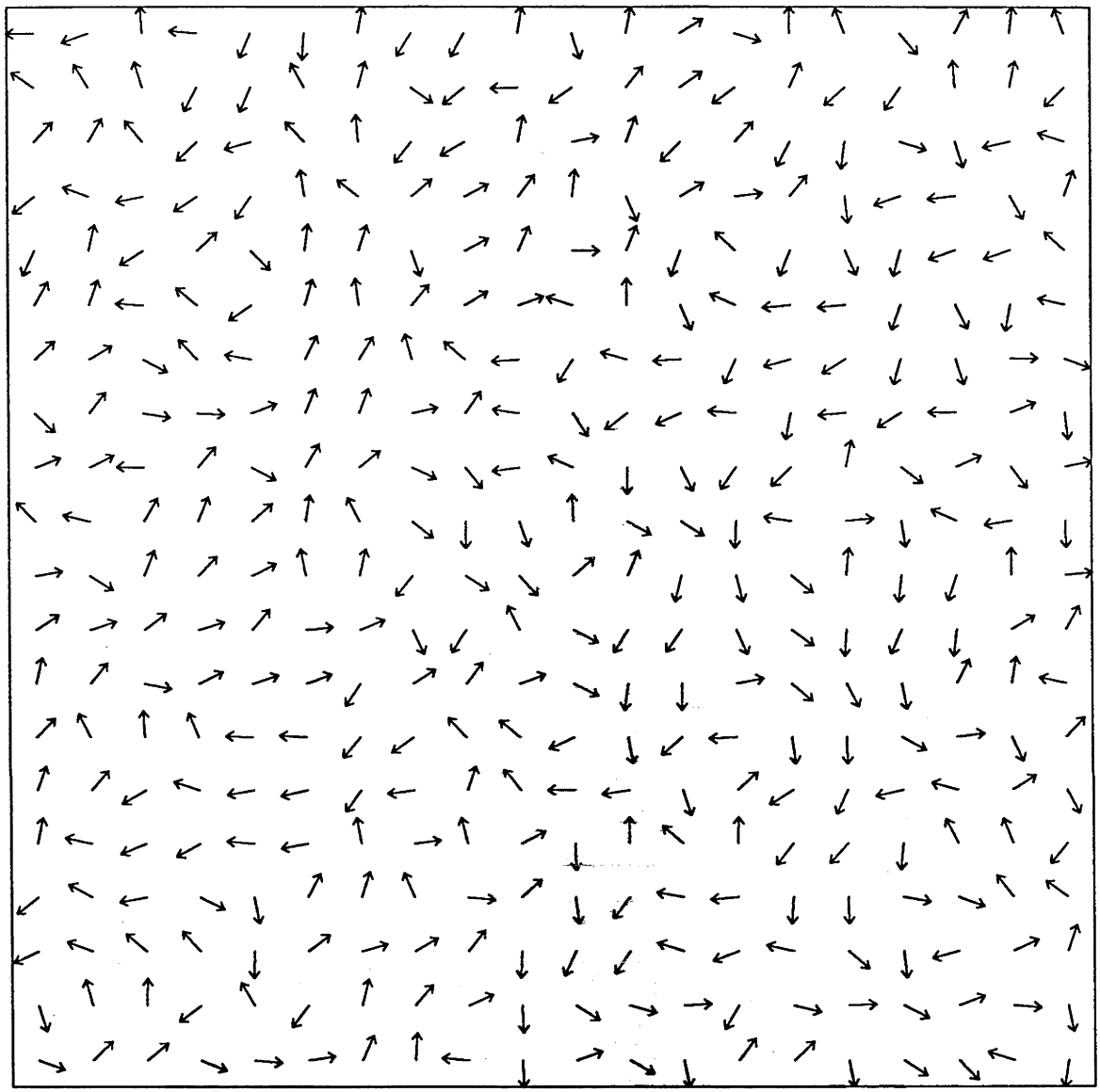


Figure 6.4b Flow pattern for 57 344 particles on a 20 x 20 grid for the PUT thermostat.

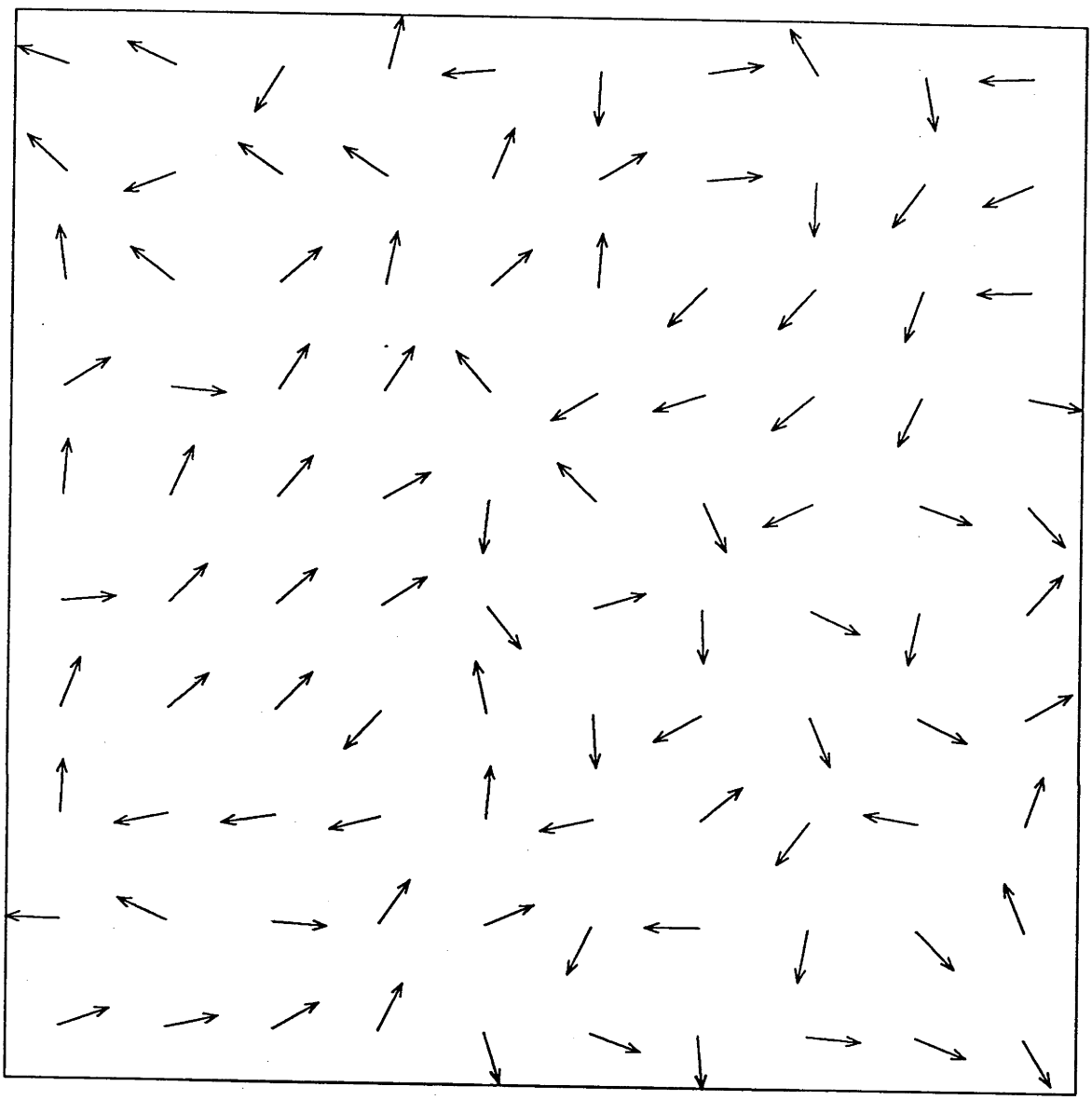


Figure 6.4c Flow pattern for 57 344 particles on a 10 x 10 grid for the PUT thermostat.

6.5 Summary of N-dependent 2-d Shear Flow

The results described in this chapter hint at a couple of interesting questions. The viscosity and shear dilatancy has been calculated to about 1% error level or less. It would be useful to be able to explore the nature of the turnover for smaller fields. This however would require a vast increase in the amount of computational resources needed. To achieve 1% error at $\gamma = 0.001$ would require 10 or more times the computation required to achieve 1% error at $\gamma = 0.01$. This would then show up some interesting behaviour which needed investigation at $\gamma = 0.0001$. Thus the computational resources required are always just beyond reach. Also, a 1% error is too large, and we would really prefer 0.1% error which is 100 times more expensive. The N-dependence problem will possibly never be more fully resolved than it has been here, and it is certainly not clear that it needs to be.

The PUT and PBT thermostats exhibit very similar results for the small shear rates, typically around 1% difference being observed. However, the transverse momentum currents appear to have an energy of around 1% of the thermal motion and thus it might be possible for some difference to be discernible if we can obtain results accurate to the 0.1% level. As this would be 100 times more expensive to compute than the current results, we are unlikely to know until the next generation of supercomputers.

References

1. An excellent review of classical statistical mechanics is given by D. A. Macquarie "Statistical Mechanics", published by Harper & Row (1976).
2. J. W. Dufty and M. J. Lindenfeld, *J. Stat. Phys.*, **20**, 259 (1979).
   E. G. D. Cohen in "Nonlinear Fluid Behaviour" edited by H. J. M. Hanley, published by North-Holland Amsterdam (1983).
   D. J. Evans and G. P. Morriss, *Phys. Rev. A*, **30**, 1528 (1984).
3. R. Kubo, *J. Phys. Soc. Japan*, **12**, 570 (1957).
4. D. J. Evans and G. P. Morriss, *Mol. Phys.*, **64**, 521 (1988).
5. G. P. Morriss and D. J. Evans, *Mol. Phys.*, **54**, 629 (1985).
6. T. Yamada and K. Kawasaki, *Prog. Theor. Phys.*, **38**, 1031 (1967).
7. G. P. Morriss and D. J. Evans, *Phys. Rev. A*, **35**, 792 (1987).
8. B. J. Alder and T. E. Wainwright, *J. Chem. Phys.*, **27**, 1208 (1957).
   A. Rahman, *Phys. Rev. A*, **136**, 405 (1964);
   J. Chem. Phys., **45**, 258 (1966).
9. D. J. Evans and G. P. Morriss, *Computer Physics Reports*, **1**, 297 (1984).
10. The Sllod equations of motion were first proposed by W. G. Hoover and A. J. C. Ladd in a private communication to D. J. Evans.
11. Q. W. Lees and S. F. Edwards, *J. Phys. C*, **5**, 1921 (1972).
12. H. J. C. Berendsen, J. P. M. Postma, W. F. van Gunsteren, A. DiNola and J. R. Haak, *J. Chem. Phys.*, **81**, 3684 (1984).
13. W. G. Hoover, A. J. C. Ladd and B. Moran, *Phys. Rev. Lett.*, **48**, 1818 (1982);
    D. J. Evans, *J. Chem. Phys.*, **78**, 3297 (1983).
14. S. Nosé, *Mol. Phys.*, **52**, 255 (1984);
    J. Chem. Phys., **81**, 511 (1984).
15. W. G. Hoover, *Phys. Rev. A*, **31**, 1695 (1985).
16. D. M. Heyes, G. P. Morriss and D. J. Evans, *J. Chem. Phys.*, **83**, 4760 (1985).

17. D. J. Evans and G. P. Morriss, *Phys. Rev. Lett.*, **56**, 2172 (1986).

18. H. J. M. Hanley, G. P. Morriss, T. R. Welberry and D. J. Evans, *Physica*, **149**, 406 (1988).

19. D. J. Evans, unpublished results.

20. W. Schönauer, "Scientific Computing on Vector Computers", published by Horth-Holland (1987).

21. D. Fincham and B. J. Ralston, *Computer Physics Comms.*, **23**, 127 (1981).

22. R. W. Hockney and J. W. Eastwood, "Computer Simulation Using Particles", published by McGraw-Hill, New York (1981);
R. W. Hockney and C. R. Jesshope, "Parallel Computers", published by Adam Hilger, Bristol (1981).

23. J. Boris, *J. Computational Physics*, **66**, 1 (1986).

24. S. G. Lambrakos and J. P. Boris, *J. Computational Physics*, **73**, 183 (1987).

25. D. J. Evans in "Molecular Dynamics Simulation of Statistical-Mechanical Systems" edited by G. Ciccotti and W. G. Hoover, published by North-Holland (1987).

26. D. Brown and J. H. R. Clarke., *Phys. Rev. A*, **34**, 2093 (1986).

27. D. J. Evans and J. F. Ely, *Mol. Phys.*, **59**, 1 (1986).

28. L. M. Hood, D. J. Evans and G. P. Morriss, *Mol. Phys.*, **62**, 419 (1987).

29. D. J. Evans and B. L. Holian, *J. Chem. Phys.*,**83**, 4069 (1985).

30. D. J. Evans, *Phys. Rev. A*, **23**, 1988 (1981).

31. W. G. Hoover, D. J. Evans, R. B. Hickman, A. J. C. Ladd, W. T. Ashurst and W. Moran, *Phys. Rev. A*, **22**, 1690 (1980).

32. D. J. Evans and G. P. Morriss, *Phys. Rev. A*, **31**, 3817 (1985).

33. D.J. Evans and W.G.Hoover, *Ann. Rev. Fluid Mech.*, **18**, 243 (1986).

34. H. J. M. Hanley and D. J. Evans, *Int. J. Thermophys.* (in press ,1988).

35. D.J. Evans and G.P.Morriss, "Dynamics of Classical Liquids" (Academic Press, New York, in press) .

36. D. J. Evans, H. J. M. Hanley and S. Hess, *Phys. Today* , **37**, 26 (1984).

37. *Physica* **118A** (1983). This volume reports the proceedings of a conference on nonlinear phenomena held in Boulder, Colorado, 1982.

38. D. J. Evans and G. P. Morriss, *Chem. Phys.*, **77**, 63 (1983).

39. H. J. M. Hanley, J. C. Rainwater, D. J. Evans and L. M. Hood, *Proc. Xth Int.Cong. Rheology* , edited by P.H.T.Uhlherr. (Sydney,1988).

40. S. Hess, *Phys. Rev. A*, **25**, 614 (1982).

41. S. Hess and H. J. M. Hanley, *Int. J. Thermophys.*, **4**, 97 (1983).

42. W. G. Hoover, M. Ross, K. W. Johnson, D. Henderson, J. A. Barker and B. C. Brown, *J.Chem.Phys.*, **52**, 4931 (1970).

43. F. J. Rogers and D. A. Young, *Phys. Rev. A*, **30**, 999 (1984).

44. J. C. Rainwater, H. J. M. Hanley, T. Paszkiewicz and Z. Petru, *J. Chem. Phys.*, **83**, 339 (1985).

See Equations (44-47) . Note there is a misprint in Eq.(45); for 0.2982 read 0.02982.

45. J. C. Rainwater, *J.Stat. Phys.*, **19**, 177 (1978).

46. W. T. Ashhurst and W. G. Hoover, *Phys. Rev. A*, **11**, 658 (1975).

47. D. J. Evans and E. G. D. Cohen, *Int. J. Thermophysics.*, (submitted).

48. J. C. Rainwater, *J. Chem. Phys.*, **71**, 5171 (1979); **74**, 4130 (1981).

49. D. J. Evans, *Phys. Rev. A*, **22**, 290 (1980).

50. D. J. Evans and H. J. M. Hanley, *Physica*, **108A**, 569 (1981).

51. K. Kawasaki and J. D. Gunton, *Phys. Rev. A*, **8**, 2048 (1973).

52. T. Yamada and K. Kawasaki, *Prog. Theor. Phys.*, **53**, 111 (1975).

53. M. H. Ernst, B. Cichocki, J. R. Dorfman, J. Sharma and H. van Beijeren, *J. Stat. Phys.*, **18**, 237 (1978).

54. D. J. Evans and G. P. Morriss, *Phys. Rev. Lett.*, **51**, 1776 (1983).

55. G. P. Morriss and D. J. Evans, *Phys. Rev. A.*, (in press).

56. D. J. Evans, G. P. Morriss and L. M. Hood, *Mol. Phys.*, (submitted).