Computer Simulation of Nanorheology for Inhomogeneous Fluids

Junfang Zhang

A thesis submitted in fulfilment of requirements for the degree of

Doctor of Philosophy

Centre for Molecular Simulation
Faculty of Information and Communication Technologies

Swinburne University of Technology

2005
**Supervisors:** Associate Professor Billy Todd  
Centre for Molecular Simulation  
Faculty of Information and Communication Technologies  
Swinburne University of Technology, Vic 3122, Australia

Professor Richard Sadus  
Centre for Molecular Simulation  
Faculty of Information and Communication Technologies  
Swinburne University of Technology, Vic 3122, Australia
DECLARATION

I hereby declare that all work described in this thesis was done by the author, except where due reference is made in the text. The work contained in this thesis was carried out since the initial commencement date, and has not been submitted previously, in part or whole, for the award of any other degree or qualification at any university.

Junfang Zhang
June 2005
ACKNOWLEDGMENTS

I am very grateful that I have got the opportunity to carry out a PhD. It has been challenging, enjoyable and worthwhile years. I have enjoyed becoming absorbed by a topic that interests me and being in a continuous process of learning.

First and foremost I would like to express my great gratitude to my supervisors, Associate Professor Billy Todd and Professor Richard Sadus who accepted me as a PhD student. They have guided me very professionally through the project, given me room to do it my way, but always supported me. I owe great thanks to them for supervising my work, simulating me intellectually and all the time and effort they have put on my PhD project.

I wish to thank Karl P. Travis, Peter Cadusch, Peter J. Daivis and Ronald Blaak for their fruitful contributions. Many thanks go to Tom Hunt, Jaroslaw Bosko and all the other staff and students at the centre who have been very helpful and caring. I am also grateful to the staff of the Faculty of Information and Communication Technologies of Swinburne University of Technology for providing an appropriate setting and conditions and their assistance, and to all those who have kindly helped during the preparation and completion of this thesis.

My special thanks are expressed to the Australian Government for an International Postgraduate Research Scholarship (IPRS) and the Faculty of Information and Communication Technologies for a School Scholarship. I also thank the Australian Partnership for Advanced Computing National Facility (APAC), Victorian Partnership for Advanced Computing (VPAC), and the Swinburne Supercomputer Centre for providing me with generous allocations of computer time.

Finally, I reserve my deepest gratitude to my family for their constant source of love and support.
This thesis is partly based on the following papers:

**Chapter 2 and 5:**


**Chapter 6:**


**Chapter 6:**


**Chapter 7:**

VI

ABSTRACT

In this thesis, we use nonequilibrium molecular dynamics (NEMD) methods to investigate the structural and dynamic properties of highly confined atomic and polymeric fluids undergoing planar Poiseuille flow.

We derive “method of planes” expressions for pressure tensor and heat flux vector for confined inhomogeneous atomic fluids under the influence of three-body forces. Our derivation is validated against NEMD simulations of a confined atomic fluid acted upon by a two-body Barker-Fisher-Watts force coupled with the Axilrod-Teller three-body force. Our method of planes calculations are in excellent agreement with the equivalent mesoscopic route of integrating the momentum and energy continuity equations directly from the simulation data. Our calculations reveal that three-body forces have an important consequence for the isotropic pressure, but have negligible influence on the shear stress and heat flux vector for a confined simple fluid.

We use the non-local linear hydrodynamic constitutive model, proposed by Evans and Morriss [1] for computing a viscosity kernel, a function of compact support, for inhomogeneous nonequilibrium fluids. Our results show that the viscosity kernel, \( \eta(y) \), has a peak at \( y = 0 \), and gets smaller and decays to zero as \( y \) increases. Physically, it means that the strain rate at the location where we want to know the stress contributes most to the stress, and the contribution of the strain rate becomes less significant as the relative distance \( y \) increases. We demonstrate that there is a limitation in the model when it is applied to our confined fluids due to the effect of domain restriction on inverse convolution.

We study the nanorheology of simple polymeric fluids. Our NEMD simulation results show that sufficiently far from the walls, the radius of gyration for molecules under shear in the middle of the channel follows the power law, \( R_g \propto N^\nu \), where \( N \) is the number of bonds and the exponent has a value \( \nu = 0.60 \pm 0.04 \), which is larger than the melt value of 0.5 for a homogeneous equilibrium fluid. Under the conditions simulated, we find that viscous forces dominate the flow, resulting in the onset of plug-like flow velocity profiles with some wall slippage. An examination of the streaming angular velocity displays a strong correlation with the radius of gyration, being maximum in those regions where \( R_g \)
is minimum and vice-versa. The angular velocity is shown to be proportional to half the strain rate sufficiently far from the walls, consistent with the behaviour for homogeneous fluids in the linear regime.

Finally, we make some concluding remarks and suggestions for future work in the final chapter.
## CONTENTS

<table>
<thead>
<tr>
<th>Chapter</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>General Introduction</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td>1.1 Motivation</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td>1.2 Review of previous work</td>
<td>2</td>
</tr>
<tr>
<td>2</td>
<td>Hydrodynamic Theory</td>
<td>9</td>
</tr>
<tr>
<td></td>
<td>2.1 Macroscopic hydrodynamics</td>
<td>9</td>
</tr>
<tr>
<td></td>
<td>2.2 Microscopic expressions for the continuity equations</td>
<td>12</td>
</tr>
<tr>
<td></td>
<td>2.3 Derivation of pressure tensor and heat flux for three-body forces</td>
<td>13</td>
</tr>
<tr>
<td>3</td>
<td>Simulation Methodology</td>
<td>26</td>
</tr>
<tr>
<td></td>
<td>3.1 Molecular Dynamics</td>
<td>26</td>
</tr>
<tr>
<td></td>
<td>3.2 SLLOD algorithm</td>
<td>37</td>
</tr>
<tr>
<td>4</td>
<td>Parallel Programming with MPI</td>
<td>39</td>
</tr>
<tr>
<td></td>
<td>4.1 Parallel architectures</td>
<td>39</td>
</tr>
<tr>
<td></td>
<td>4.2 Introduction to MPI</td>
<td>42</td>
</tr>
<tr>
<td></td>
<td>4.3 Sample of parallelized program for Poiseuille flow</td>
<td>51</td>
</tr>
<tr>
<td></td>
<td>4.4 Running a parallelized program</td>
<td>57</td>
</tr>
<tr>
<td>5</td>
<td>Simulation of Atomic Poiseuille Flow under the Influence of Three-Body Forces</td>
<td>61</td>
</tr>
<tr>
<td></td>
<td>5.1 General simulation considerations</td>
<td>61</td>
</tr>
<tr>
<td></td>
<td>5.2 Simulation results</td>
<td>65</td>
</tr>
<tr>
<td>6</td>
<td>Non-local Linear Hydrodynamic Constitutive Model</td>
<td>76</td>
</tr>
<tr>
<td></td>
<td>6.1 Navier-Stokes theory for fluids in slit pores</td>
<td>76</td>
</tr>
<tr>
<td></td>
<td>6.2 Non-local constitutive equation</td>
<td>79</td>
</tr>
<tr>
<td></td>
<td>6.3 The method employed for solving the model</td>
<td>81</td>
</tr>
<tr>
<td></td>
<td>6.4 Discrete Fourier Transforms</td>
<td>83</td>
</tr>
<tr>
<td>Section</td>
<td>Page</td>
<td></td>
</tr>
<tr>
<td>------------------------------------------------------------------------</td>
<td>------</td>
<td></td>
</tr>
<tr>
<td>6.5 Simulation details</td>
<td>85</td>
<td></td>
</tr>
<tr>
<td>6.6 Simulation and DFT results</td>
<td>89</td>
<td></td>
</tr>
<tr>
<td>6.7 Effect of domain restriction on inverse convolution</td>
<td>106</td>
<td></td>
</tr>
<tr>
<td>6.8 Analytic Fourier Transform</td>
<td>117</td>
<td></td>
</tr>
<tr>
<td>Chapter 7 Simulation of Confined Polymeric Fluids Undergoing Poiseuille Flow</td>
<td>124</td>
<td></td>
</tr>
<tr>
<td>7.1 Computational methodology</td>
<td>124</td>
<td></td>
</tr>
<tr>
<td>7.2 Simulation results</td>
<td>127</td>
<td></td>
</tr>
<tr>
<td>Chapter 8 Summary and Conclusions</td>
<td>160</td>
<td></td>
</tr>
<tr>
<td>Bibliography</td>
<td>164</td>
<td></td>
</tr>
</tbody>
</table>
1 General Introduction

In this introduction we start with our research motivation and then give a brief overview of the history of the rheological study of flows, and more specifically nano-scale flows.

1.1 Motivation

The study of the rheology of fluids in nanoporous materials is very important because of the technological applications as well as the fundamental scientific interest. For example, a lubricant is applied to the disk surface to protect the disk from being damaged during operation [2]. Many applications exist where the situation of boundary lubrication is encountered, in which the spacing between the confining solid surfaces reaches nanometer length scales. When fluid is narrowly confined within spacing comparable to the molecular dimension, its structural and dynamic properties deviate strongly from the bulk behaviour in various respects. It was shown that a deviation from typical quadratic velocity profiles predicted by classical Navier-Stokes dynamic theory can be observed for planar Poiseuille flow [3]. The shear stress exhibits oscillations about the classical liner profile and highly layer-structured density profiles are present. Strong density variations across the fluid channel result in spatial inhomogeneities in all transport properties [4, 5, 6, 7] that deviate substantially from their bulk properties.

An understanding of these properties can be very important in adhesion, lubrication, colloidal stability, separations, and materials synthesis. In fact, these properties
of confined fluids, such as viscosity, heat flux, and boundary conditions are far from fully understood, primarily due to the failure of continuum theories at such small length scales, and the apparent difficulty of current real experimental techniques to probe the transport properties at the atomic and molecular level, since most experiments only provide macroscopic manifestations of the actual microscopic phenomena. This leads to a loss of crucial microscopic information. Even the most advanced experimental technique such as the surface force apparatus (SFA) provides only very limited information. Therefore, molecular simulation [8, 9, 10], the most powerful tool, has mostly been used to investigate the behaviour of such a confined system, as it has the advantage of providing more detailed information on the microscopic level.

The goal of this study is to fundamentally understand the spatially dependent structural and dynamic properties for inhomogeneous nonequilibrium fluids using nonequilibrium molecular dynamics. Molecular simulations not only provide the capability to probe the behaviour of confined fluids at a microscopic level, thus providing the most fundamental understanding of the rheological properties which cannot be captured from existing theories, but also enable us to probe nonlinear phenomena at extreme nanoscale conditions which is impossible to achieve using existing experimental equipment.

1.2 Review of previous work

Rheological behaviours are a subject of practical and long-lasting interest. At a fundamental level, they are usually studied in the framework of the Boltzmann equation and related kinetic equations. To first order in Knudsen number (mean free path divided by hydrodynamic length scale), the Chapman-Enskog expansion leads to the Navier-Stokes equations [11]. Higher order expansions (i.e., Burnett and super Burnett) however, have had limited success, particularly with regard to convergence and boundary conditions [12].
The moment method, introduced by Maxwell, proves to be more successful in theoretical work but is not yet widely used as the basis for numerical calculations. While theoretical advances continue to be made [13, 14, 15], a general approach to solving the Boltzmann equation has not been established.

In this regard, numerical experiments have played an important role in modern kinetic theory. Historically, computational simulations of dilute systems have been done by two groups. The first is the statistical physics community [16], which has focused primarily on molecular dynamics simulations of gases and lattice-based schemes with simplified collision dynamics. The second branch is in the engineering community. Stochastic particle methods such as direct simulation Monte Carlo (DSMC) [17] and its variants were introduced by aerospace engineers during the 1960’s and 70’s, when space exploration was the primary application for rarefied flows. The two branches developed separate numerical approaches since they focused on different problems. In statistical physics the work was primarily in equilibrium and simple non-equilibrium states, such as Couette and Poiseuille flow. The engineering community was interested in complex systems such as hypersonic flight and extreme non-equilibrium flows. In recent years, the two communities have discovered the utility of the algorithms developed by each, especially as their interests turn to meso- or nano-scale applications. Some of the new directions in computational study of flows include [18]:

**Particle/Continuum hybrids:**

Many interesting flows that require the use of microscopic simulation have not been studied due to their computational expense, which is several orders of magnitude greater than that of continuum hydrodynamic methods. However, one needs to simulate fluids at the atomic level to understand their chemical and physical behaviour. Hybrid schemes, which combine a microscopic algorithm with a continuum hydrodynamic calculation have proved useful. The need for such hybrid methods arises from the fact that hydrodynamics
modelled by continuum representations are often inaccurate while solutions generated using molecular resolution globally are not feasible. For example, the Adaptive Mesh and Algorithm Refinement (AMAR) scheme uses direct simulation Monte Carlo at the finest grid scale in an adaptive mesh refinement hierarchy and a Navier-Stokes solver at other scales [19]. The atomistic model is applied locally in regions where the continuum description is invalid or inaccurate. The continuum and atomistic representations are coupled by proper averaging and interpolation of data between scales, adjusting grids as the flow solution evolves, adaptively deciding where and when the calculation should switch from a continuum algorithm to a microscopic algorithm.

**Plasma flows:**

The properties of plasma flow has been a very interesting topic to study due to the many important applications for plasmas such as semiconductor processing, surface treatments and pollution control. Molecular dynamic simulations of charged fluids are extremely time consuming, mainly because of the long-ranged Coulomb interactions. The question then arises as to the possibility of setting up alternative mesoscopic methods. Today, there are some plasma problems that have been analysed by direct simulation Monte Carlo (DSMC) and Particle-In-Cell techniques [20], where charged plasma species are modelled as individual macro-particles (each macro-particle represents a large number of real particles). Particles move in the plasma using Newton’s laws and self-consistently calculated electric fields resulting from applied voltages and inter-particle Coulomb forces.

**Nano-scale flows:**

Nano-scale flows play an increasingly important role in fluid dynamics, thanks to advances in computer technology. When the characteristic length scale of a flow is comparable to the mean free path of the constituent molecules, the traditional macroscopic description breaks down. Molecular simulations, primarily molecular dynamics [10, 21]
and direct simulation Monte Carlo [16], are at present the main computational tools in the study of these transitional flows. An example is the simple Poiseuille flow confined to a channel only several molecular diameters wide. Perhaps the first molecular dynamics simulation of Poiseuille flow is the one reported in [22]. Possibly the first study of Poiseuille flow driven by an external force was carried out by Bitsanis et al. [3] and Kadanoff et al. [23], who simulated laminar pipe flow with lattice-gas automata to verify the parabolic profile of momentum density for this system and observe the well-known logarithmic divergence in the viscosity. Esposito et al. [24] studied the Boltzmann equation and found that if the force is sufficiently weak there is a solution which converges, in the hydrodynamic limit, to the local equilibrium distribution with parameters given by the stationary solution of the Navier-Stokes (NS) equations. Alaoui and Santos [25] carried out kinetic theory analysis of Poiseuille flow clearly exhibiting non-Newtonian behaviour. They found an exact solution of the Bhatnagar-Gross-Krook (BGK) kinetic equation for a gas of Maxwell molecules. An exact solution was found for a particular value of the force parameter. At a hydrodynamic level, the solution is characterized by a parabolic profile of the flow velocity with respect to a space variable scaled with the local collision frequency. The general solution of the BGK model under the form of an expansion in powers of acceleration through fifth order was obtained by Tij and Santos [26].

Contrary to bulk fluids, interfacial or confined fluids exhibit strong density variations on a microscopic, molecular scale, which has been observed in many molecular simulations [27, 28, 29] and has been deduced experimentally [30]. This fact introduces serious conceptual and operational difficulties in the description of their transport behaviour. Recently, Sokhan and Quirke studied the influence of the boundary conditions at the solid-fluid interface on tracer diffusivity and collective diffusivity. They obtained a relationship between the collective diffusivity and the Maxwell coefficient describing wall
Supple and Quirke carried out molecular dynamics simulations of carbon nanotubes imbibing oil at an oil/vapour interface [32]. They derived expressions for the penetration length and the velocity of the imbibing oil and related both to the solid-fluid surface tensions and interfacial friction via the Maxwell coefficient.

For interfacial or confined fluid systems, due to the lack of uniformity on a molecular scale, the traditional microscopic identification of the mass flux, stress tensor, streaming velocity and energy flux is inappropriate for fluids in nano dimension space. Due to the difficulties in obtaining information from real experiments, the molecular dynamics (MD) method has been used for the calculation of transport properties of fluids. Equilibrium MD calculations employ the well-known formula of linear response theory, the so-called Green-Kubo relations [33, 34], which are not expected to be applicable for systems far from equilibrium. Despite the success of this approach in diffusivity calculations [35, 36], it runs into difficulties when applied for the calculation of viscosity [37, 38]. In order to overcome these difficulties a number of nonequilibrium simulation techniques were developed which calculate the shear viscosity directly from a constitutive relation [39]. According to usual hydrodynamics, local transport coefficients can be assigned at each point inside an inhomogeneous fluid. These are the same as the transport coefficients of a homogeneous fluid at the local density, i.e., they are point functions of the density. Usual hydrodynamics is of course applicable only if the fluid density, despite its macroscopic variation, hardly changes over molecular distances. Bitsanis et al. [29] have noticed that the effects of strong density inhomogeneities on the flow and transport behaviour are much weaker than what they would have been, had the local transport coefficients been literally point functions of the density. Therefore they presented a model, the local average density model (LADM), which allows for viscosity variations over the pore cross section by evaluating the local viscosity at a coarse-grained density obtained upon local averaging of the density over a sphere of radius half the molecular diameter.
However, their analysis assumes a uniform pressure over the cross section, while using the equilibrium density profiles in violation of the Gibbs-Duhem relation.

The most successful approach to the confined system was first attempted by Davis and co-workers [4] who have generalized Enskog’s kinetic theory to fluids with strong density inhomogeneities over molecular distances. The constitutive equations resulting from such a generalization are different and more complicated than those of bulk fluids due to the inherent anisotropy of the strongly inhomogeneous medium. The corresponding hydrodynamic equations, therefore, are much more complex than the usual Navier-Stokes equations, which makes their application tedious to all but the simplest flows.

The generalized Enskog’s kinetic theory was applied to self-diffusion in fluids confined between flat, solid surfaces, i.e., fluids in a slit pore [6]. Such an approach follows the idea that local transport coefficients in an inhomogeneous fluid can be set equal to those for a homogeneous fluid whose density is set equal to some density obtained by averaging over densities in the immediate region of the point of interest in the inhomogeneous system. Although the use of such averaged density rests on rigorous foundations for the equilibrium case, no analogous foundations exists for their use for nonequilibrium fluids and the expressions for the transport coefficients were found only for the cases of a local equilibrium velocity distribution for weakly inhomogeneous fluids, with inhomogeneity in one direction. Pozhar and Gubbins [40] have developed a microscopic theory, a functional perturbation theory, describing nonequilibrium behaviour of dense, strongly inhomogeneous fluids and have shown that generalized Langevin equations should be regarded as the first order form of the functional perturbation theory. Subsequently, the generalized Langevin equations were used to obtain linearized kinetic equations which were solved [41] to obtain linearized Navier-Stokes equations and the associated transport coefficients. The major limitation to this theory is that it relates local values of
transport coefficients to integrals over equilibrium inhomogeneous singlet and pair correlation functions. It is unlikely that the theory will hold when systems are far from equilibrium. A comparison of the Pozhar-Gubbins (PG) theory was made with the local viscosity from NEMD simulations of a fluid undergoing planar Poiseuille flow [42]. However, larger numerical uncertainty is observed in the computation of local NEMD viscosities. Strong density variations across the fluid channel results in spatial inhomogeneities in all transport properties, which were also observed by many experimental investigations using the surface force apparatus [30, 43].

In this work, we apply NEMD techniques to study the rheological behaviour of fluids undergoing Poiseuille flow in a highly confined slit pore. Structural and dynamic properties in terms of the density profile, streaming velocity, shear stress, strain rate and shear viscosity are investigated.

This thesis is structured as follows. Chapter 1 gives a general introduction to the history of the rheological study of flows, and more specifically nano-scale flows. Chapter 2 describes the theoretical background and derives the pressure tensor and heat flux vector for fluids under the influence of three-body force. Our derivation is validated against nonequilibrium molecular dynamics simulations of a confined fluid acted upon by a two-body Barker-Fisher-Watts force coupled with the Axilrod-Teller three-body force in Chapter 5. In Chapter 3, an introduction to the basic simulation methodologies used in this work is given. In Chapter 4 we describe the parallel programming scheme used to implement our code with message passing interface (MPI). In Chapter 6 we use the non-local linear hydrodynamic constitutive model proposed by Evans and Morris [1] for computing the viscosity kernel for inhomogeneous nonequilibrium fluids. In Chapter 7 we study the rheology of highly confined polymeric fluids undergoing Poiseuille flow.
In this chapter the basic theoretical framework of this thesis is introduced, in particular the microscopic expressions for the continuity equations. We then proceed to derive the pressure tensor and heat flux vector for fluids under the influence of three-body forces. Our derivation is validated against nonequilibrium molecular dynamics simulations of a confined fluid acted upon by a two-body Barker-Fisher-Watts force coupled with the Axilrod-Teller three-body force in Chapter 5. This chapter is partly based on work published in [44].

2.1 Macroscopic hydrodynamics

A central problem in the study of hydrodynamics is the computation of transport coefficients. Therefore, we are interested in the macroscopic process of mass, energy and momentum transfer. Because of conservation of these quantities, they can only change by a process of redistribution. If the process occurs on a molecular time scale, it would be unobservable at a macroscopic level. But if it is slow, it is observable and plays a macroscopic role. The macroscopic equations of motion for the densities of conserved quantities are called the Navier-Stokes equations [1]. We will now give a brief description of how these equations are derived. It is important to understand this derivation because based on these equations the microscopic expressions for the mass,
Hydrodynamic Theory

Momentum and energy densities are defined and the pressure tensor and heat flux vector are derived.

If \( M(t) \) is the total mass contained in an arbitrary volume \( V \), then

\[
M = \int_V \rho(r, t) \, dr \tag{2.1}
\]

where \( \rho(r, t) \) is the mass density at position \( r \) and time \( t \). Because of conservation of mass, mass in the volume \( V \) changes by flowing through an enclosing surface, \( S \).

\[
\frac{dM}{dt} = -\int_S \rho(r, t) u(r, t) \, dS = -\int_V \nabla \cdot [\rho(r, t) u(r, t)] \, dr \tag{2.2}
\]

where \( u(r, t) \) is the fluid streaming velocity at position \( r \) and time \( t \), and \( dS \) denotes an area element of the surface \( S \). The rate of change of the enclosed mass can also be expressed in terms of the change in mass density \( \rho(r, t) \), as

\[
\frac{dM}{dt} = \int_V \rho \frac{\partial \rho(r, t)}{\partial t} \, dr \tag{2.3}
\]

Equating (2.2) and (2.3) gives

\[
\frac{\partial \rho(r, t)}{\partial t} = -\nabla \cdot [\rho(r, t) u(r, t)] \tag{2.4}
\]

which is the mass continuity equation governing the conservation of mass.

Similarly we can derive a momentum continuity equation. If \( G(t) \) is the total momentum of an arbitrary volume \( V \), then

\[
G = \int_V J(r, t) \, dr \tag{2.5}
\]

where \( J(r, t) = \rho(r, t) u(r, t) \) is the momentum density at position \( r \) and time \( t \). The rate of change of momentum is given by

\[
\frac{dG}{dt} = \int_V \rho \frac{\partial J(r, t)}{\partial t} \, dr = \int_V \rho \frac{\partial [\rho(r, t) u(r, t)]}{\partial t} \tag{2.6}
\]
The total momentum of volume $V$ can change firstly by convection, that is, by flowing through a closed surface $S$ enclosing the volume $V$. This convective term is

$$\frac{dG_c}{dt} = -\int_S dS.\left[\mathbf{J}(\mathbf{r}, t)\mathbf{u}(\mathbf{r}, t)\right]$$

(2.7)

Momentum can also change by the pressure exerted on the fluid on volume $V$ by the surrounding fluid. The force $d\mathbf{F}$, exerted on an elementary area $dS$ is,

$$d\mathbf{F} = dS.\mathbf{P}$$

(2.8)

where $\mathbf{P}$ is the pressure tensor. We note here that $\mathbf{P} = -\sigma$, where $\sigma$ is the stress tensor. Thus the stress contribution to the change in momentum can be written as,

$$\frac{dG_s}{dt} = -\int_S dS.\mathbf{P}$$

(2.9)

Combining (2.7) and (2.9), we have,

$$\frac{dG}{dt} = \frac{dG_c}{dt} + \frac{dG_s}{dt} = -\int_S dS.\left[\rho(\mathbf{r}, t)\mathbf{u}(\mathbf{r}, t)\mathbf{u}(\mathbf{r}, t) + \mathbf{P}\right]$$

(2.10)

Equating the two expressions (2.6) and (2.10) gives,

$$\frac{\partial \mathbf{J}(\mathbf{r}, t)}{\partial t} = \frac{\partial [\rho(\mathbf{r}, t)\mathbf{u}(\mathbf{r}, t)\mathbf{u}(\mathbf{r}, t)]}{\partial t} = -\nabla.\left[\rho(\mathbf{r}, t)\mathbf{u}(\mathbf{r}, t)\mathbf{u}(\mathbf{r}, t) + \mathbf{P}\right]$$

(2.11)

which is known as the momentum continuity equation.

We now consider the energy continuity equation. Let the total energy per unit mass be $e(\mathbf{r}, t)$, and the total energy density be $\rho(\mathbf{r}, t)e(\mathbf{r}, t)$. For a convecting fluid, the energy density has two components, one is the kinetic energy part, and the other is the thermodynamic internal energy density, $\rho(\mathbf{r}, t)U(\mathbf{r}, t)$. Let $E(t)$ be the total energy inside an arbitrary volume $V$, then

$$E = \int_V d\mathbf{r}\rho(\mathbf{r}, t)e(\mathbf{r}, t).$$

(2.12)
The rate of change of energy is given by
\[
\frac{dE}{dt} = \int_V d\mathbf{r} \frac{\partial \rho(\mathbf{r}, t) \mathbf{e}(\mathbf{r}, t)}{\partial t} \quad (2.13)
\]

The total energy inside volume \( V \) can change via convection through the containing surface, diffusion through the surface and the work done on the volume \( V \) by the surface stresses. In order, these terms can be written,
\[
\frac{dE}{dt} = -\int_S d\mathbf{S} \left[ \rho(\mathbf{r}, t) \mathbf{e}(\mathbf{r}, t) \mathbf{u}(\mathbf{r}, t) + \mathbf{J}_Q \right] - \int_S (d\mathbf{S} \cdot \mathbf{P}(\mathbf{r}, t)) \cdot \mathbf{u}(\mathbf{r}, t)
\]
\[
= -\int_V d\mathbf{r} \nabla \cdot \left[ \rho(\mathbf{r}, t) \mathbf{e}(\mathbf{r}, t) \mathbf{u}(\mathbf{r}, t) + \mathbf{J}_Q + \mathbf{P}(\mathbf{r}, t) \cdot \mathbf{u}(\mathbf{r}, t) \right] \quad (2.14)
\]

where \( \mathbf{J}_Q \) is the heat flux vector. Equating the two expressions (2.13) and (2.14) gives,
\[
\frac{\partial \rho(\mathbf{r}, t) \mathbf{e}(\mathbf{r}, t)}{\partial t} = -\nabla \cdot \left[ \rho(\mathbf{r}, t) \mathbf{e}(\mathbf{r}, t) \mathbf{u}(\mathbf{r}, t) + \mathbf{J}_Q + \mathbf{P}(\mathbf{r}, t) \cdot \mathbf{u}(\mathbf{r}, t) \right] \quad (2.15)
\]

This is the form of the energy continuity equation.

2.2 Microscopic expressions for the continuity equations

In section 2.1 we briefly outlined the continuity equations of macroscopic hydrodynamics. In this section we will give an introduction to the microscopic description for the local densities of mass, momentum and energy and the three continuity equations.

If the mass of the individual atoms in our system is \( m \), then the mass density at a position \( \mathbf{r} \) and time \( t \) can be calculated by sitting at a particular point in phase space and calculating the density of ensemble points as a function of time. The mass density can now be calculated by summing the values of mass at at a position \( \mathbf{r} \), \( \sum_i m \delta(\mathbf{r} - \mathbf{r}_i) \), but weighting these values by the current value of the \( N \)-particle distribution function \( f(\mathbf{\Gamma}, t) \) at that place in phase space. This is the Schrödinger representation of the mass density, \( \rho(\mathbf{r}, t) \)
\[
\rho(\mathbf{r}, t) = \int d\mathbf{\Gamma} \ f(\mathbf{\Gamma}, t) \sum_i m \delta(\mathbf{r} - \mathbf{r}_i)
\]
Alternatively, we can calculate the value at time \( t \) by following the mass, \( \sum_i m \delta(r - r_i(t)) \), as it changes along a single trajectory in phase space. The mass density can then be calculated by summing the values of the mass, \( \sum_i m \delta(r - r_i(t)) \), with a weighting factor determined by the probability of starting from each initial phase \( \Gamma \). These probabilities are chosen from an initial distribution function \( f(\Gamma, 0) \). This is the Heisenberg representation of mass density,

\[
\rho(r, t) = \int d\Gamma \ f(\Gamma, 0) \ \sum_i m \ \delta(r - r_i(t)) = \langle \sum_i m \ | \ r_i(t) = r \rangle \quad (2.16)
\]

The momentum density, \( \rho u(r, t) \), and total energy density, \( \rho e(r, t) \), are defined similarly.

\[
\rho(r, t) u(r, t) = \int d\Gamma \ f(\Gamma, t) \ \sum_i m v_i \ \delta(r - r_i) \\
= \int d\Gamma \ f(\Gamma, t) \ \sum_i p_i \ \delta(r - r_i) \\
= \langle \sum_i p_i(t) \ | \ r_i(t) = r \rangle \quad (2.17)
\]

\[
\rho(r, t) e(r, t) = \int d\Gamma \ f(\Gamma, t) \left[ \frac{1}{2} \sum_i m v_i^2 + \frac{1}{2} \sum_{ij} \phi_{ij}^{(2)} + \frac{1}{3} \sum_{ijk} \phi_{ijk}^{(3)} \right] \delta(r - r_i) \\
= \langle \frac{1}{2} \sum_i m v_i^2 + \frac{1}{2} \sum_{ij} \phi_{ij}^{(2)} + \frac{1}{3} \sum_{ijk} \phi_{ijk}^{(3)} \ | \ r_i(t) = r \rangle \quad (2.18)
\]

In these equations \( v_i \) is the velocity of particle \( i \), \( p_i \) is its momentum. \( \phi_{ij}^{(2)} = \phi^{(2)}(r_i, r_j) \) is the two-body potential and \( \phi_{ijk}^{(3)} = \phi^{(3)}(r_i, r_j, r_k) \) is the three-body potential.

### 2.3 Derivation of pressure tensor and heat flux for three-body forces

In order to derive expressions for the pressure tensor and heat flux vector, we should give a brief introduction to the microscopic expressions for the mass, momentum and energy densities in \( k \)-space.

We define the Fourier transform pair by

\[
f(k) = \int dr \ e^{ik \cdot r} \ f(r)
\]
\[ f(r) = \frac{1}{(2\pi)^3} \int d\mathbf{k} \, e^{-i\mathbf{k} \cdot \mathbf{r}} \, f(\mathbf{k}) \]  

(2.19)

The instantaneous \( r \)-space mass density is defined as follows,

\[ \rho(\mathbf{r}, t) = \sum_{i=1}^{N} m \, \delta(\mathbf{r} - \mathbf{r}_i(t)) \]  

(2.20)

The \( \mathbf{k} \)-space instantaneous mass density is then,

\[ \rho(\mathbf{k}, t) = \int d\mathbf{r} \, \sum_{i=1}^{N} m \, \delta(\mathbf{r} - \mathbf{r}_i(t)) \, e^{i\mathbf{k} \cdot \mathbf{r}} \]

\[ = \sum_{i=1}^{N} m \, e^{i\mathbf{k} \cdot \mathbf{r}_i(t)} \]  

(2.21)

The instantaneous \( r \)-space momentum density is defined as follows,

\[ \mathbf{J}(\mathbf{r}, t) = \sum_{i=1}^{N} m \mathbf{v}_i \, \delta(\mathbf{r} - \mathbf{r}_i(t)) \]  

(2.22)

The \( \mathbf{k} \)-space instantaneous momentum density is then,

\[ \mathbf{J}(\mathbf{k}, t) = \int d\mathbf{r} \, \sum_{i=1}^{N} m \mathbf{v}_i \, \delta(\mathbf{r} - \mathbf{r}_i(t)) \, e^{i\mathbf{k} \cdot \mathbf{r}} \]

\[ = \sum_{i=1}^{N} m \mathbf{v}_i \, e^{i\mathbf{k} \cdot \mathbf{r}_i(t)} \]  

(2.23)

The instantaneous \( r \)-space energy density is

\[ \rho c(\mathbf{r}, t) = \left[ \frac{1}{2} \sum_i m \mathbf{v}_i^2 + \frac{1}{2} \sum_{ij} \phi^{(2)}_{ij} + \frac{1}{3} \sum_{ijk} \phi^{(3)}_{ijk} \right] \, \delta(\mathbf{r} - \mathbf{r}_i(t)) \]  

(2.24)

The \( \mathbf{k} \)-space instantaneous energy density is then,

\[ \rho c(\mathbf{k}, t) = \sum_i \left[ \frac{1}{2} m \mathbf{v}_i^2 + \frac{1}{2} \sum_j \phi^{(2)}_{ij} + \frac{1}{3} \sum_{jk} \phi^{(3)}_{ijk} \right] \, e^{i\mathbf{k} \cdot \mathbf{r}_i(t)} \]  

(2.25)

2.3.1 Pressure tensor

Since we are interested in planar interfaces with a normal parallel to the \( y \) axis, which is perpendicular to the surface of the wall, it is convenient to consider a partial Fourier
transform over the $y$ coordinate,

\[ J_\alpha(k_y, x, z) \equiv \int_{-\infty}^{\infty} \sum_i m v_{\alpha i} \delta(r - r_i) e^{ik_y y} dy \]

\[ = \sum_i m v_{\alpha i} \delta(x - x_i) \delta(z - z_i) e^{ik_y y} \]  

(2.26)

If the fluid is assumed to be uniform in the $x$, $z$ directions we can average over them and write the transformed momentum density as

\[ J_\alpha(k_y) = \frac{1}{A} \sum_i m v_{\alpha i} e^{ik_y y_i} \]  

(2.27)

where $A$ is the magnitude of the area of the $x - z$ surface that has its normal in the $y$ direction. $m$ and $v_{\alpha i}$ are the mass and laboratory velocity of particle $i$, respectively, and $\alpha = x, y, z$.

Now we perform the Fourier transform on the momentum continuity equation,

\[ \frac{\partial J(r, t)}{\partial t} = -\nabla \cdot \left[ P + \rho(r, t) u(r, t) u(r, t) \right]. \]  

(2.28)

In $k$-space the momentum continuity equation is

\[ \frac{\partial J_\alpha(k_y)}{\partial t} = ik_y [P_{\alpha y}(k_y) + \mathcal{F}\{\rho(y)u_\alpha(y)u_y(y)\}] , \]  

(2.29)

where $\mathcal{F}\{}$ denotes the Fourier transform of the quantity in brackets. Substituting (2.27) into (2.29) gives the potential contribution to the wave-vector-dependent pressure tensor as

\[ P_{\alpha y}^U(k_y) = \frac{1}{A} \sum_i \frac{F_{\alpha i}}{ik_y} e^{ik_y y_i} , \]  

(2.30)

while the kinetic contribution is

\[ P_{\alpha y}^K(k_y) = \frac{1}{A} \sum_i \frac{mv_{\alpha i}}{ik_y} \frac{d}{dt} e^{ik_y y_i} - \mathcal{F}\{\rho u_\alpha u_y\} . \]  

(2.31)
We make an inverse Fourier transform of the configurational component of the pressure tensor (2.30), to find that

\[ P_{\alpha y}^{U}(y) = \frac{1}{2\pi A} \sum_{i} F_{\alpha i} e^{ik_{y}y_{i}} e^{-ik_{y}y} \, dk_{y} \]  

(2.32)

Utilizing the fact that

\[ \frac{1}{\pi} \int_{-\infty}^{\infty} \frac{\exp(iky)}{ik} \, dk = \text{sgn}(y) \]  

(2.33)

it is straightforward to show

\[ P_{\alpha y}^{U}(y) = \frac{1}{2A} \sum_{i} F_{\alpha i} \text{sgn}(y_{i} - y) \]  

(2.34)

where \( F_{\alpha i} \) is the \( \alpha \) component of the total force on atom \( i \) including both two- and three-body contributions. For two-body contributions, Todd et al [45] have detailed the derivations. In what follows we specifically consider the case of three-body forces and use symmetry relations to generate a useful expression for the potential contribution to the three-body pressure. The kinetic contribution remains unchanged as it implicitly contains the full two- and three-body force contributions in the particle momenta.

We now separate out the two- and three-body contributions to the pressure

\[ P_{\alpha y}^{U}(y) = \frac{1}{2A} \sum_{i} F_{\alpha i} \text{sgn}(y_{i} - y) \]

\[ = \frac{1}{2A} \sum_{i} (F_{\alpha i}^{(2)} + F_{\alpha i}^{(3)}) \text{sgn}(y_{i} - y) \]

\[ = P_{\alpha y}^{(2)U}(y) + P_{\alpha y}^{(3)U}(y) \]  

(2.35)
where \( P_{y}^{(2)}(y) \) and \( P_{y}^{(3)}(y) \) are contributions from two-body and three-body forces, respectively. In [45] it was shown that

\[
P_{y}^{(2)}(y) = \frac{1}{2A} \sum_{i} F_{\alpha i}^{(2)} sgn(y_{i} - y)
\]

\[
= \frac{1}{4A} \left[ \sum_{ij} F_{\alpha ij}^{(2)} sgn(y_{i} - y) + \sum_{ij} F_{\alpha ji}^{(2)} sgn(y_{j} - y) \right]
\]

\[
= \frac{1}{4A} \sum_{ij} F_{\alpha ij}^{(2)} [sgn(y_{i} - y) - sgn(y_{j} - y)]
\]

\[
= \frac{1}{2A} \sum_{ij} F_{\alpha ij}^{(2)} \left[ \Theta(y_{i} - y)\Theta(y - y_{j}) - \Theta(y_{j} - y)\Theta(y - y_{i}) \right],
\]

(2.36)

where \( \Theta \) is the Heaviside step function. \( F_{\alpha i}^{(2)} \) is the \( \alpha \) component of two-body force on atom \( i \). \( F_{ij}^{(2)} \) is defined here to be the contribution to the total two-body force on atom \( i \) due to atom \( j \). If \( \phi_{ij}^{(2)} = \phi^{(2)}(r_{i}, r_{j}) \) is the two-body potential, then

\[
F_{i}^{(2)} \equiv \sum_{j} F_{ij}^{(2)} = -\sum_{j} \frac{\partial \phi_{ij}^{(2)}}{\partial r_{i}}
\]

Making similar use of particle exchange symmetry, the three-body contribution to the pressure tensor can be expressed as

\[
P_{y}^{(3)}(y) = \frac{1}{2A} \sum_{i} F_{\alpha i}^{(3)} sgn(y_{i} - y)
\]

\[
= \frac{1}{6A} \left[ \sum_{i} F_{\alpha i}^{(3)} sgn(y_{i} - y) + \sum_{j} F_{\alpha j}^{(3)} sgn(y_{j} - y) + \sum_{k} F_{\alpha k}^{(3)} sgn(y_{k} - y) \right],
\]

(2.37)

where

\[
F_{i}^{(3)} \equiv \sum_{jk} (F_{ij}^{(3)} + F_{ik}^{(3)}) = -\sum_{jk} \left( \frac{\partial \phi_{ijk}^{(3)}}{\partial r_{ij}} + \frac{\partial \phi_{ijk}^{(3)}}{\partial r_{ik}} \right)
\]

(2.38)

\[
r_{ij} \equiv r_{i} - r_{j}, \quad F_{ij}^{(3)} \equiv -\frac{\partial \phi_{ijk}^{(3)}}{\partial r_{ij}}
\]

\( F_{i}^{(3)} \) is the three-body force on atom \( i \). \( F_{ij}^{(3)} \) is defined here to be the contribution to the total three-body force on atom \( i \) due to atom \( j \). Substitution of Eq. (2.38) into (2.37)
yields
\[
P^{(3)T}_{\alpha y}(y) = \frac{1}{6A} \left[ \sum_{ij} F^{(3)}_{\alpha ij} \text{sgn}(y_i - y) + \sum_{ik} F^{(3)}_{\alpha ik} \text{sgn}(y_i - y) \\
+ \sum_{ji} F^{(3)}_{\alpha ji} \text{sgn}(y_j - y) + \sum_{jk} F^{(3)}_{\alpha jk} \text{sgn}(y_k - y) \\
+ \sum_{ki} F^{(3)}_{\alpha ki} \text{sgn}(y_k - y) + \sum_{kj} F^{(3)}_{\alpha kj} \text{sgn}(y_k - y) \right] \\
= \frac{1}{6A} \left\{ \sum_{ij} F^{(3)}_{\alpha ij} [\text{sgn}(y_i - y) - \text{sgn}(y_j - y)] \\
+ \sum_{ik} F^{(3)}_{\alpha ik} [\text{sgn}(y_i - y) - \text{sgn}(y_k - y)] \\
+ \sum_{jk} F^{(3)}_{\alpha jk} [\text{sgn}(y_j - y) - \text{sgn}(y_k - y)] \right\}
\] (2.39)

\[
P^{(3)T}_{\alpha y}(y) = \frac{1}{3A} \left\{ \sum_{ij} F^{(3)}_{\alpha ij} [\Theta(y_i - y)\Theta(y - y_j) - \Theta(y_j - y)\Theta(y - y_i)] \\
+ \sum_{ik} F^{(3)}_{\alpha ik} [\Theta(y_i - y)\Theta(y - y_k) - \Theta(y_k - y)\Theta(y - y_i)] \\
+ \sum_{jk} F^{(3)}_{\alpha jk} [\Theta(y_j - y)\Theta(y - y_k) - \Theta(y_k - y)\Theta(y - y_j)] \right\}
\] (2.40)

Equation (2.39) demonstrates that the potential contribution to the three-body pressure at a plane located at \( y \) occurs when components of the three-body force intersect that plane, in complete analogy with the two-body force contributions. For example, consider the situation shown in Fig. 2.1, in which a triangular configuration of three particles is shown. Only the force contributions along the vectors \( \mathbf{r}_{12} \) and \( \mathbf{r}_{13} \) intersect the plane at \( y = y_0 \) and contribute to the three-body pressure at this plane.

The kinetic part of the pressure tensor can be obtained by making an inverse Fourier transform of the kinetic component of the pressure tensor (2.31)

\[
P^{K}_{\alpha y}(y) = \frac{1}{2\pi A} \int_{-\infty}^{\infty} dk_y \sum_{\alpha i} \frac{mv_{\alpha i}}{ik_y} \frac{d}{dt} \exp[ik_y(y_i - y)] - \rho u_{\alpha} u_y.
\] (2.41)
2.3 Derivation of pressure tensor and heat flux for three-body forces

\[ y = y_0 \]

**Figure 2.1:** Triangular configuration of atoms and the plane located at \( y = y_0 \).

Contributions to the pressure tensor are included from atoms 1, 2, and 3 along the vectors \( r_{12} \) and \( r_{13} \).

If we interchange the order of integrating with respect to \( k_y \) and differentiating with respect to time we find

\[
P_{\text{eq}}^K (y) = \frac{1}{2\pi A} \sum_i m v_{\alpha i} \frac{d}{dt} \int_{-\infty}^{\infty} dk_y \exp[i k_y (y_i - y)] \frac{e^{i k_y y}}{i k_y} - \rho u_\alpha u_y
\]

\[
= \frac{1}{2A} \sum_i m v_{\alpha i} \frac{d}{dt} sgn(y_i - y) - \rho u_\alpha u_y
\]

(2.42)

In obtaining the second of these equalities, (2.33) has been used. Utilizing the fact that

\[
\frac{d[sgn(y_i - y)]}{dt} = \frac{d[sgn(y_i - y)]}{d(y_i - y)} \frac{d(y_i - y)}{dt} = \frac{d[sgn(y_i - y)]}{d(y_i - y)} \cdot v_{yi}, \text{ and}
\]

\[
\frac{d}{dy} sgn(y) = 2\delta(y)
\]

(2.43)
(2.42) can be written as

\[ P^K_{\alpha y}(y) = \frac{1}{A} \sum_i \frac{p_{\alpha i} p_{y i}}{m} \delta(y_i - y) , \]  \hspace{1cm} (2.44)

where \( p_{\alpha i} \) and \( p_{y i} \) are \( \alpha \) and \( y \) components of the peculiar momenta of particle \( i \). Here the peculiar momentum is defined as \( \mathbf{p}_i \equiv \mathbf{v}_i - \mathbf{c}(y) \), where \( \mathbf{v}_i \) is the laboratory momentum of particle \( i \) and \( \mathbf{c}(y) \) is the streaming velocity at \( y \). The time averaged kinetic component of the pressure tensor can be usefully expressed as

\[ \langle P^K_{\alpha y}(y) \rangle = \lim_{t \to \infty} \frac{1}{At} \sum_{0<\tau_i<j<t} \sum_i p_{\alpha i}(t_{i,j}) sgn[p_{y i}(t_{i,j})] . \]  \hspace{1cm} (2.45)

Here it is noted that particle \( i \) crosses the plane at \( y \) at a set of times \( (t_{i,j}; i = 1, \cdots, N; j = 1, 2, \cdots) \).

### 2.3.2 Heat flux vector

As with the pressure tensor derivation, we follow the method of planes formalism developed in [46] for the heat flux vector. That approach uses the microscopic definitions of the local energy density and the Fourier transform of the energy continuity equation to obtain a \( \mathbf{k} \)-space expression for the heat flux vector, which is again back transformed into \( \mathbf{r} \)-space.

We start the derivation at the Fourier transformed energy density continuity equation (2.15)

\[ \frac{\partial \rho(e(k, t))}{\partial t} = i\mathbf{k}.[\mathbf{J}_Q(k, t) + \mathcal{F}\{\rho(e \mathbf{u})\} + \mathcal{F}\{\mathbf{P} \cdot \mathbf{u}\}] . \]  \hspace{1cm} (2.46)

As we have shown in Eq. (2.25) the \( \mathbf{k} \)-space instantaneous energy density is

\[ \rho(e(k, t)) = \sum_i \left[ \frac{1}{2} m \mathbf{v}_i^2 + \frac{1}{2} \sum_j \phi^{(2)}_{ij} + \frac{1}{3} \sum_{jk} \phi^{(3)}_{ijk} \right] e^{i\mathbf{k} \cdot \mathbf{r}_i(t)} . \]

The total energy of atom \( i \) is

\[ e_i = \frac{1}{2} m \mathbf{v}_i^2 + \frac{1}{2} \sum_j \phi^{(2)}_{ij} + \frac{1}{3} \sum_{jk} \phi^{(3)}_{ijk} . \]
We first compute the time derivative of the energy density in k-space, Eq. (2.25)

\[
\frac{\partial [\rho e(k, t)]}{\partial t} = i \mathbf{k} \cdot \left( \sum_i v_i e_i e^{i \mathbf{k} \cdot \mathbf{r}_i} \right) + \sum_i m v_i \cdot \mathbf{v}_i e^{i \mathbf{k} \cdot \mathbf{r}_i} \\
+ \frac{1}{2} \sum_{ij} \left( \dot{\mathbf{r}}_i \cdot \frac{\partial \phi^{(2)}_{ij}}{\partial \mathbf{r}_i} + \dot{\mathbf{r}}_j \cdot \frac{\partial \phi^{(2)}_{ij}}{\partial \mathbf{r}_j} \right) e^{i \mathbf{k} \cdot \mathbf{r}_i} \\
+ \frac{1}{3} \sum_{ijk} \left( \dot{\mathbf{r}}_i \cdot \frac{\partial \phi^{(3)}_{ijk}}{\partial \mathbf{r}_i} + \dot{\mathbf{r}}_j \cdot \frac{\partial \phi^{(3)}_{ijk}}{\partial \mathbf{r}_j} + \dot{\mathbf{r}}_k \cdot \frac{\partial \phi^{(3)}_{ijk}}{\partial \mathbf{r}_k} \right) e^{i \mathbf{k} \cdot \mathbf{r}_i}.
\]

Consider now the two terms containing the three-body forces. The first term may be symmetrized as

\[
\sum_i v_i \cdot F^{(3)}_i e^{i \mathbf{k} \cdot \mathbf{r}_i} = \frac{1}{3} \left[ \sum_i v_i \cdot F^{(3)}_i e^{i \mathbf{k} \cdot \mathbf{r}_i} + \sum_j v_j \cdot F^{(3)}_j e^{i \mathbf{k} \cdot \mathbf{r}_j} + \sum_k v_k \cdot F^{(3)}_k e^{i \mathbf{k} \cdot \mathbf{r}_k} \right] \\
= \frac{1}{3} \left[ \sum_{ijk} v_i \cdot (F^{(3)}_{ij} + F^{(3)}_{ik}) e^{i \mathbf{k} \cdot \mathbf{r}_i} + \sum_{ijk} v_j \cdot (F^{(3)}_{ji} + F^{(3)}_{jk}) e^{i \mathbf{k} \cdot \mathbf{r}_j} + \sum_{ijk} v_k \cdot (F^{(3)}_{ki} + F^{(3)}_{kj}) e^{i \mathbf{k} \cdot \mathbf{r}_k} \right].
\]
The second term containing three-body forces in Eq. (2.47) may be similarly expanded:

\[
\frac{1}{3} \sum_{ijk} \left( \dot{r}_i \frac{\partial \phi_{ijk}^{(3)}}{\partial r_i} + \dot{r}_j \frac{\partial \phi_{ijk}^{(3)}}{\partial r_j} + \dot{r}_k \frac{\partial \phi_{ijk}^{(3)}}{\partial r_k} \right) e^{ikr_i} = \frac{1}{3} \sum_{ijk} \left[ v_i \cdot \left( \frac{\partial \phi_{ijk}^{(3)}}{\partial r_{ij}} + \frac{\partial \phi_{ijk}^{(3)}}{\partial r_{ik}} \right) + v_j \cdot \left( \frac{\partial \phi_{ijk}^{(3)}}{\partial r_{ji}} + \frac{\partial \phi_{ijk}^{(3)}}{\partial r_{jk}} \right) + v_k \cdot \left( \frac{\partial \phi_{ijk}^{(3)}}{\partial r_{ki}} + \frac{\partial \phi_{ijk}^{(3)}}{\partial r_{kj}} \right) \right] e^{ikr_i} \tag{2.49}
\]

Defining \( S^{(3)} \) as the sum of Eqs. (2.48) and (2.49) gives

\[
S^{(3)} = -\frac{1}{3} \sum_{ijk} \left[ v_j \cdot F_{ij}^{(3)} (e^{ikr_i} - e^{ikr_j}) 
- v_j \cdot F_{jk}^{(3)} (e^{ikr_i} - e^{ikr_j}) 
+ v_k \cdot F_{ik}^{(3)} (e^{ikr_i} - e^{ikr_k}) 
+ v_k \cdot F_{jk}^{(3)} (e^{ikr_i} - e^{ikr_k}) \right]. \tag{2.50}
\]

We now permute the triplet indices in Eq. (2.50) such that all velocities are in term of the index \( i \) to obtain

\[
S^{(3)} = \frac{1}{3} \sum_{ijk} v_i \cdot \left[ F_{ik}^{(3)} + F_{ij}^{(3)} \right] (e^{ikr_i} - e^{ikr_j}) 
+ \frac{1}{3} \sum_{ijk} v_i \cdot \left[ F_{ij}^{(3)} + F_{ik}^{(3)} \right] (e^{ikr_i} - e^{ikr_j}) 
= \frac{1}{3} \sum_{ik} v_i \cdot F_i^{(3)} (e^{ikr_i} - e^{ikr_k}) 
+ \frac{1}{3} \sum_{ij} v_i \cdot F_i^{(3)} (e^{ikr_i} - e^{ikr_j}) \tag{2.51}
\]
2.3 Derivation of pressure tensor and heat flux for three-body forces

Substituting Eq. (2.51) back into Eq. (2.47) gives

\[
\frac{\partial [\rho e(k, t)]}{\partial t} = i\mathbf{k} \cdot \left( \sum_i v_i e_i e^{ikr_i} \right) + \frac{1}{2} \sum_{ij} v_i \cdot F_{ij}^{(2)} (e^{ikr_i} - e^{ikr_j}) + \frac{1}{3} \sum_{ik} v_i \cdot F_{ik}^{(3)} (e^{ikr_i} - e^{ikr_k}) + \frac{1}{3} \sum_{ij} v_i \cdot F_{ij}^{(3)} (e^{ikr_i} - e^{ikr_j}).
\]  
\text{(2.52)}

Substitution of Eq. (2.52) into the Fourier transformed energy continuity equation (2.46) yields

\[
i\mathbf{k} \cdot \mathbf{J}_Q(k, t) = i\mathbf{k} \cdot \left( \sum_i [v_i - u_{ri}(t)] e_i e^{ikr_i} \right) + \frac{1}{2} \sum_{ij} v_i \cdot F_{ij}^{(2)} [e^{ikr_i} - e^{ikr_j}] + \frac{1}{3} \sum_{ik} v_i \cdot F_{ik}^{(3)} [e^{ikr_i} - e^{ikr_k}] + \frac{1}{3} \sum_{ij} v_i \cdot F_{ij}^{(3)} [e^{ikr_i} - e^{ikr_j}] - i\mathbf{k} \cdot \mathcal{F}\{\mathbf{p} \cdot \mathbf{u}\}.
\]  
\text{(2.53)}

Integrating over \( x \) and \( z \), dividing by \( ik_y \) and taking the inverse Fourier transform yields

\[
AJ_{Qy}(y, t) = \sum_i (v_{yi} - u_{y_i}) e_i \delta(y - y_i) - \frac{1}{4} \sum_{ij} v_i \cdot F_{ij}^{(2)} [\text{sgn}(y - y_i) - \text{sgn}(y - y_j)] - \frac{1}{6} \sum_{ij} v_i \cdot F_{ij}^{(3)} [\text{sgn}(y - y_i) - \text{sgn}(y - y_j)] - \frac{1}{6} \sum_{ik} v_i \cdot F_{ik}^{(3)} [\text{sgn}(y - y_i) - \text{sgn}(y - y_k)] - A\{\mathbf{p} \cdot \mathbf{u}\}_y.
\]  
\text{(2.54)}

We substitute the expression for the pressure tensor, Eq. (2.39), into Eq. (2.54) to give the kinetic and potential contributions to the heat flux vector

\[
J_{Qy}(y, t) = J_{Qy}^K(y, t) + J_{Qy}^U(y, t).
\]  
\text{(2.55)}
The kinetic contribution is, as with the pressure tensor, identical in form to the original derivation in [46]:

\[
J^K_{Qy}(y, t) = \frac{1}{A} \sum_i [v_{yi} - u(y)]U_i\delta(y - y_i) \tag{2.56}
\]

except that here \(U_i\) is the internal energy of a particle defined as

\[
U_i = \frac{1}{2}m[v_i - u(y_i)]^2 + \frac{1}{2} \sum_j \phi_{ij}^{(2)} + \frac{1}{3} \sum_{jk} \phi_{ijk}^{(3)}. \tag{2.57}
\]

The potential contribution to the heat flux vector is

\[
J^U_{Qy}(y, t) = -\frac{1}{2A} \sum_{ij} [v_i - u(y)] \cdot F^{(2)}_{ij} [\text{sgn}(y - y_i) - \text{sgn}(y - y_j)]
\]

\[-\frac{1}{6A} \sum_{ij} [v_i - u(y)] \cdot F^{(3)}_{ij} [\text{sgn}(y - y_i) - \text{sgn}(y - y_j)]
\]

\[-\frac{1}{6A} \sum_{ik} [v_i - u(y)] \cdot F^{(3)}_{ik} [\text{sgn}(y - y_i) - \text{sgn}(y - y_k)]
\]

\[-\frac{1}{6A} \sum_{ijk} v_i F^{(3)}_{ijk} [\text{sgn}(y - y_i) - \text{sgn}(y - y_k)]
\]

\[+ \frac{1}{6A} \sum_{jk} u(y) \cdot F^{(3)}_{jk} [\text{sgn}(y - y_j) - \text{sgn}(y - y_k)]. \tag{2.58}
\]

An alternative, more concise form of Eq. (2.58) is [44]

\[
J^U_{Qy}(y, t) = -\frac{1}{2A} \sum_i [v_i - u(y)] \cdot F^{(2)}_{i} \text{sgn}(y - y_i)
\]

\[-\frac{1}{2A} \sum_i [v_i - u(y)] \cdot F^{(3)}_{i} \text{sgn}(y - y_i)
\]

\[+ \frac{1}{6A} \sum_{ijk} v_i F^{(3)}_{ijk} [\text{sgn}(y - y_i) + \text{sgn}(y - y_j) + \text{sgn}(y - y_k)]. \tag{2.59}
\]

The last three terms in Eq. (2.58) and the last term in Eq. (2.59) are not direct analogies of the two-body heat flux, unlike the case in the pressure tensor three-body expressions, which are direct analogies. They are a result of particle velocities coupling to three-body forces, which does not occur in the pressure tensor calculation. However, it will be shown in Chapter 5 that these additional terms are negligible, if not zero. As was
the case for the kinetic term for the pressure tensor, the kinetic term for the heat flux
vector in (2.56) can be written in a more useful way for computer simulation as [46]
\[ J^K_{QY}(y) = \lim_{t \to \infty} \frac{1}{At} \sum_{0 < t_{ij} < t} \sum_i U_i \text{sgn}[c_{yi}(t_{ij})], \] (2.60)
where \( c_i \equiv v_i - u(y) \) is the plane peculiar velocity of atom \( i \).

Eqs. (2.36), (2.39), (2.59) and (2.60) will be used for computing the pressure tensor and
heat flux vector in Chapter 5.
3 Simulation Methodology

In this chapter the basic simulation methodologies used in this work are described. Specific descriptions for the equations of motion, force calculation, periodic boundary conditions, integration algorithm, Gaussian constraints and SLLOD algorithm are given.

3.1 Molecular Dynamics

If we have two masses interacting through gravitational forces we can exactly predict what is going to happen at any point in the future or calculate backwards to any point in history. But adding one extra mass makes it impossible to analytically solve the time evolution of the system. Although we know exactly how the masses interact with each other and what the equations of motion are there is, in general, no way of telling whether they will collide and if so at which instant in time analytically. Only by numerically solving the equations of motion for short successive intervals of time can this problem be tackled. Molecular dynamics (MD) simulation is a powerful tool to be used to solve the classical equations of motion for many body systems and has been successfully used to simulate systems with many millions of atoms interacting with each other.

3.1.1 Equations of motion

The microscopic coordinates and momenta of all the molecules should be determined, as they are used to obtain thermodynamic and transport properties via statistical mechanics. If the simulations are carried out for a sufficient length of time, the resulting
time averages are equivalent to ensemble averages due to ergodicity [47]. It is assumed that the motion of atoms is governed by Newton’s classical equations of motion given by

\[
\frac{p_i}{m_i} = \frac{dr_i}{dt} \quad \text{(3.1)}
\]

\[
F_i = \frac{dp_i}{dt} \quad \text{(3.2)}
\]

where \(F_i\) and \(p_i\) are the force and momentum of atom \(i\) respectively. The force is given by

\[
F_i = -\frac{\partial \phi}{\partial r_i} \quad \text{(3.3)}
\]

The potential energy \(\phi\) in the above equation is given by the sum of inter and intramolecular interactions. The equations are integrated using finite difference schemes, which typically involve a calculation of forces between molecules at each time step. Examples of such integration schemes are the velocity Verlet, the Gear-predictor, and the leap-frog algorithms [48, 49].

### 3.1.2 Force calculation

The force field or the potential energy used to describe interactions between atoms in the system is a function of the set of coordinates of each atom. The accuracy of results directly depends on how well the force field describes the various interactions between the atoms. It is therefore very important to select the appropriate model for each system.

In this work, the atomic fluids are simulated according to a model with truncated and shifted Lennard-Jones (LJ) interactions between all atoms. The LJ potential describing all interactions in the system is truncated and shifted so that the potential has no discontinuity and is zero beyond a cut-off distance \(r_c = 2^{1/6} \sigma\) (where \(\sigma\) is the distance at which the unshifted potential is zero). A LJ potential with this truncation point is often known as the WCA potential [50], and it results in purely repulsive interactions. This
potential is convenient for computational work because it is short-ranged, and therefore computationally undemanding, but still retains the essential physics, i.e. the repulsive (excluded volume) interaction. The truncated and shifted LJ potential is defined as

\[
\phi(r) = \begin{cases} 
4\varepsilon \left[ \left( \sigma/r \right)^{12} - \left( \sigma/r \right)^{6} \right] + \varepsilon & r \leq 2^{1/6}\sigma \\
0 & r \geq 2^{1/6}\sigma
\end{cases}
\] (3.4)

where \(\varepsilon\) is the LJ energy parameter and \(\sigma\) the LJ length parameter. When using the LJ potentials in simulation, it is customary to work in a system of units where \(\sigma = 1\) and \(\varepsilon = 1\). Comparison of WCA and LJ potentials is shown in figure 3.1.

**Figure 3.1:** Comparison of WCA and LJ potentials. Reduced units used (\(\sigma = \varepsilon = 1\)).

In order to validate our pressure tensor and heat flux vector derivation in section 2.3, we also perform some simulations on fluid and solid atoms that interact via the Barker-Fisher-Watts two-body potential [51] and Axilrod-Teller three-body potential [52]. The
total intermolecular potential $\phi$ is a contribution from two-body interactions $\phi^{(2)}$ and three-body dispersion interactions $\phi^{(3)}$

$$\phi(r) = \phi^{(2)}(r) + \phi^{(3)}(r) .$$ (3.5)

The two-body interaction of argon is well represented by the Barker-Fisher-Watts (BFW) potential [51]. The BFW potential is a linear combination of the Barker-Pompe [53] ($\phi_{BP}$) and Bobetic-Barker [54] ($\phi_{BB}$) potentials

$$\phi^{(2)}(r) = 0.75\phi_{BB}(r) + 0.25\phi_{BP}(r) ,$$ (3.6)

where the potentials of Barker-Pompe and Bobetic-Barker have the following form:

$$\phi^{(2)}(r) = \varepsilon \left[ \sum_{i=0}^{5} A_i (x - 1)^i \exp[\alpha(1 - x)] - \sum_{j=0}^{2} \frac{C_{2j+6}}{\delta + x^{2j+6}} \right] .$$ (3.7)

Here, $x = r/r_m$ where $r_m$ is the intermolecular separation at which the potential has a minimum value and the other parameters are listed in Table 3.1.

The triple-dipole Axilrod-Teller (AT) potential [52] is

$$\phi^{(3)}(r_i, r_j, r_k) = \frac{v_{DDD}(1 + 3\cos\theta_i\cos\theta_j\cos\theta_k)}{(r_{ij}r_{ik}r_{jk})^3} ,$$ (3.8)

where $v_{DDD}$ is the nonadditive coefficient, and the angles and intermolecular separations refer to a triangular configuration of atoms. The nonadditive coefficient for argon is 518.3 a.u. [55]. Recent work [56] has demonstrated that the Axilrod-Teller term can significantly improve the prediction of liquid phase properties.

3.1.3 Periodic boundary conditions

The problem of surface effects in small systems can be overcome through the use of periodic boundary conditions (PBC) [47] whereby the simulation box is replicated to form an infinite lattice by rigid translation in all the three principal directions, completely
<table>
<thead>
<tr>
<th></th>
<th>Argon</th>
<th>Barker-Pompe</th>
<th>Bobetic-Barker</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\varepsilon/\kappa (K)$</td>
<td>142.095</td>
<td>147.70</td>
<td>140.235</td>
</tr>
<tr>
<td>$\sigma (\text{Å})$</td>
<td>3.3605</td>
<td>3.341</td>
<td>3.3666</td>
</tr>
<tr>
<td>$r_m (\text{Å})$</td>
<td>3.7612</td>
<td>3.7560</td>
<td>3.7630</td>
</tr>
<tr>
<td>$A_0$</td>
<td>0.2349</td>
<td>0.29214</td>
<td></td>
</tr>
<tr>
<td>$A_1$</td>
<td>-4.7735</td>
<td>-4.41458</td>
<td></td>
</tr>
<tr>
<td>$A_2$</td>
<td>-10.2194</td>
<td>-7.70182</td>
<td></td>
</tr>
<tr>
<td>$A_3$</td>
<td>-5.2905</td>
<td>-31.9293</td>
<td></td>
</tr>
<tr>
<td>$A_4$</td>
<td>0.0</td>
<td>-136.026</td>
<td></td>
</tr>
<tr>
<td>$A_5$</td>
<td>0.0</td>
<td>-151.00</td>
<td></td>
</tr>
<tr>
<td>$C_6$</td>
<td>1.0698</td>
<td>1.11976</td>
<td></td>
</tr>
<tr>
<td>$C_8$</td>
<td>0.1642</td>
<td>0.171551</td>
<td></td>
</tr>
<tr>
<td>$C_{10}$</td>
<td>0.0132</td>
<td>0.013748</td>
<td></td>
</tr>
<tr>
<td>$\alpha$</td>
<td>12.5</td>
<td>12.5</td>
<td></td>
</tr>
<tr>
<td>$\delta$</td>
<td>0.01</td>
<td>0.01</td>
<td></td>
</tr>
</tbody>
</table>

Table 3.1: Parameters for the Barker-Fisher-Watts potential.
filling the space. In other words, if one of the particles is located at position \( \mathbf{r} \) in the box, we assume that this particle really represents an infinite set of particles located at

\[
\mathbf{r} + l\mathbf{a} + m\mathbf{b} + n\mathbf{c} \quad (l, m, n = -\infty, +\infty)
\]

where \( l, m, n \) are integer numbers, and \( \mathbf{a}, \mathbf{b}, \mathbf{c} \) are the vectors corresponding to the edges of the box. The key point is that now each particle \( i \) in the box should be thought as interacting not only with other particles \( j \) in the box, but also with their images in nearby boxes. Consequently, the position of the box boundaries has no effect (that is, there are no artificial walls in the systems) and surface effects are eliminated. Apparently, the number of interacting pairs increases enormously as an effect of the PBC, which turns out to be very computationally intensive when the number of molecules, \( N \), becomes large. In practice, this is not true, because most common intermolecular potential energy functions such as the LJ potential decay rapidly at large separation distances. For example, the LJ potential asymptotes rapidly to zero for distances greater than \( 2.5\sigma \).

Customarily, a potential cut-off radius is used whereby the potential is computed only for separations smaller than a distance \( r_c \). Interactions are disregarded for separations greater than \( r_c \). The cut-off distance relates to the periodic boundary conditions, in that it can be any distance up to half the length of the simulation box. The effects of truncating a full-ranged potential can be approximately estimated by treating the system as a uniform (constant density) continuum beyond \( r_c \). For a bulk periodic system, this usually amounts to a constant additive correction to quantities such as potential energy and pressure.

### 3.1.4 Time integration algorithm

The time evolution of a set of interacting particles is followed by integrating their equations of motion. Any changes in particle position will also affect the strength of interparticle forces, velocities, and accelerations. Consequently, a time integration algorithm
is required to integrate the equations of motion of the interacting particles and follow their trajectory. Time integration algorithms are based on finite difference methods, where time is discretized on a finite grid, the time step $\Delta t$ being the distance between consecutive points on the grid. Knowing the positions and some of their time derivatives at time $t$ (the exact details depend on the type of algorithm), the integration scheme gives these quantities at a later time $t + \Delta t$. By iterating the procedure, the time evolution of the system can be followed for long times. Of course, these schemes are approximate and there are errors associated with them. These errors are due to truncation, which is related to the accuracy of the finite difference method with respect to the true solution. Finite difference methods are usually based on a Taylor expansion truncated at some term. These errors do not depend on the implementation: they are intrinsic to the algorithm. Another source of errors is roundoff error, which is associated with a particular implementation of the algorithm, for instance, the finite number of digits used in the computer simulation. Both errors can be reduced by decreasing the time step $\Delta t$.

We use the Gear predictor-corrector algorithm [35, 47, 57] to integrate the equations of motion for its efficiency and accuracy. It consists of three steps:

- **Predictor.** From the positions and their time derivatives up to a certain order $n$ (we choose $n = 4$), all known at time $t$, the same quantities at time $t + \Delta t$ are predicted by means of a Taylor expansion. We use $r_i^{(n)}$ to represent the $n^{th}$ order derivative term in the Taylor expansion of the position vector of particle $i$, $r_i$, with respect to time $t$,

$$r_i^{(n)} = \frac{1}{n!} (\Delta t)^n \left( \frac{d^n r_i}{dt^n} \right).$$  \hspace{1cm} (3.9)

Similarly,

$$p_i^{(n)} = \frac{1}{n!} (\Delta t)^n \left( \frac{d^n p_i}{dt^n} \right).$$  \hspace{1cm} (3.10)
The predicted position vector, \( \mathbf{r}_i^p \), momentum, \( \mathbf{p}_i^p \), and their scaled time derivatives, \( \mathbf{r}_i^{(np)} \) and \( \mathbf{p}_i^{(np)} \) at time \( t + \Delta t \) are as follows,

\[
\begin{pmatrix}
\mathbf{r}_i^p(t + \Delta t) \\
\mathbf{r}_i^{(1)p}(t + \Delta t) \\
\mathbf{r}_i^{(2)p}(t + \Delta t) \\
\mathbf{r}_i^{(3)p}(t + \Delta t) \\
\mathbf{r}_i^{(4)p}(t + \Delta t)
\end{pmatrix}
= \begin{pmatrix}
1 & 1 & 1 & 1 & 1 \\
0 & 1 & 2 & 3 & 4 \\
0 & 0 & 1 & 3 & 6 \\
0 & 0 & 0 & 1 & 4 \\
0 & 0 & 0 & 0 & 1
\end{pmatrix}
\begin{pmatrix}
\mathbf{r}_i(t) \\
\mathbf{r}_i^{(1)}(t) \\
\mathbf{r}_i^{(2)}(t) \\
\mathbf{r}_i^{(3)}(t) \\
\mathbf{r}_i^{(4)}(t)
\end{pmatrix}
\tag{3.11}
\]

\[
\begin{pmatrix}
\mathbf{p}_i^p(t + \Delta t) \\
\mathbf{p}_i^{(1)p}(t + \Delta t) \\
\mathbf{p}_i^{(2)p}(t + \Delta t) \\
\mathbf{p}_i^{(3)p}(t + \Delta t) \\
\mathbf{p}_i^{(4)p}(t + \Delta t)
\end{pmatrix}
= \begin{pmatrix}
1 & 1 & 1 & 1 & 1 \\
0 & 1 & 2 & 3 & 4 \\
0 & 0 & 1 & 3 & 6 \\
0 & 0 & 0 & 1 & 4 \\
0 & 0 & 0 & 0 & 1
\end{pmatrix}
\begin{pmatrix}
\mathbf{p}_i(t) \\
\mathbf{p}_i^{(1)}(t) \\
\mathbf{p}_i^{(2)}(t) \\
\mathbf{p}_i^{(3)}(t) \\
\mathbf{p}_i^{(4)}(t)
\end{pmatrix}
\tag{3.12}
\]

- **Force evaluation.** The force is computed taking the gradient of the potential at the predicted positions. The resulting acceleration \( \mathbf{a}_i^c(t + \Delta t) \) will be in general different from the predicted acceleration \( \mathbf{a}_i^p(t + \Delta t) \). The difference between the two

\[
\Delta \mathbf{a}_i(t + \Delta t) = \mathbf{a}_i^c(t + \Delta t) - \mathbf{a}_i^p(t + \Delta t)
\tag{3.13}
\]

can be an error signal.
3. Simulation Methodology

- **Corrector.** This error signal is used to correct positions and their derivatives. All the corrections are proportional to the error signal. The corrector step is simply:

\[
\begin{pmatrix}
    \mathbf{r}_i^c(t + \Delta t) \\
    \mathbf{r}_i^{(1)c}(t + \Delta t) \\
    \mathbf{r}_i^{(2)c}(t + \Delta t) \\
    \mathbf{r}_i^{(3)c}(t + \Delta t) \\
    \mathbf{r}_i^{(4)c}(t + \Delta t)
\end{pmatrix}
= \begin{pmatrix}
    \mathbf{r}_i^p(t + \Delta t) \\
    \mathbf{r}_i^{(1)p}(t + \Delta t) \\
    \mathbf{r}_i^{(2)p}(t + \Delta t) \\
    \mathbf{r}_i^{(3)p}(t + \Delta t) \\
    \mathbf{r}_i^{(4)p}(t + \Delta t)
\end{pmatrix}
+ \begin{pmatrix}
    k_0 \\
    k_1 \\
    k_2 \\
    k_3 \\
    k_4
\end{pmatrix}
\Delta \mathbf{a}_i(t + \Delta t)
\] (3.14)

Here \( k_0, k_1, k_2, k_3, k_4 \), are the corrector coefficients which depend upon the order of the differential equation being solved [57]. In our case, \( k_0 = 251/720, k_1 = 1, k_2 = 11/12, k_3 = 1/3, k_4 = 1/24 \).

Other methods can be used to integrate equations of motion of the particles. The Commonly used Verlet algorithm has a truncation error proportional to \( \Delta t^4 \) for each integration time step. For more detailed information on time integration algorithm, the reader is referred to [47, 49]. A comparison between the Verlet and Gear schemes can be found in [58].

3.1.5 **Gaussian constraints**

In this work, we simulate Poiseuille flow. The simulation geometry is shown in Fig. 5.1. In order to keep the wall temperature constant in our simulation, we apply Gaussian constraints by introducing a perturbative force into the equations of motion, and minimising the least squares difference between the Newtonian and constrained trajectories. Generally speaking, there are two types of constraints: holonomic and nonholonomic.
holonomic constraint is one which can be expressed as equations connecting the coordinates of the particles (and possibly the time) having the form

\[ f(r_1, r_2, r_3, \ldots, t) = 0. \]  

(3.15)

Holonomic constraints may be integrated out of the equations of motion. Perhaps the simplest example of holonomic constraints is the rigid body, where the constraints are expressed by equations of the form

\[ (r_i - r_j)^2 = l^2, \]  

(3.16)

where \( l \) is the distance between particle \( i \) and particle \( j \). If we have \( M \) such equations then the original \( 3N \) coordinates are no longer free to vary independently. Connecting the \( M \) equations together establishes equations of constraint between the coordinates. For such a system having \( 3N \) coordinates and \( M \) constraint equations, there are \((3N - M)\) degrees of freedom.

Constraints which cannot be expressed as either algebraic or integrable differential equations are known as nonholonomic constraints. Nonholonomic constraints usually involve velocities, are not integrable, and usually do work on a system. Thermodynamic constraints are generally nonholonomic. In particular, there are two types of nonholonomic constraints in practical problems. The first type is the inequality constraint of the form

\[ f(r_1, r_2, r_3, \ldots) \geq 0. \]  

(3.17)

The second type of nonholonomic constraints is the nonintegrable differential equation,

\[ f(r_1, r_2, r_3, \ldots; v_1, v_2, v_3, \ldots; t) = 0. \]  

(3.18)

Consider the function \( f \), a function of the set of particle accelerations

\[ f(\ddot{r}_1, \ddot{r}_2, \ddot{r}_3, \ldots \ddot{r}_N) = \frac{1}{2} \sum_{i=1}^{N} \left[ \ddot{r}_i - \frac{F_i}{m_i} \right]^2. \]  

(3.19)
If we impose constant temperature constraints,

$$ G(\dot{r}_1, \dot{r}_2, \dot{r}_3, \ldots, \dot{r}_N; t) = \sum_{i=1}^{N} \frac{m_i \dot{r}_i^2}{2} - \frac{3NkT}{2} = 0 , \quad (3.20) $$

then we have the differential form of the constraint equation,

$$ g(\ddot{r}_1, \ddot{r}_2, \ddot{r}_3, \ldots, \ddot{r}_N; \dot{r}_1, \dot{r}_2, \dot{r}_3, \ldots, \dot{r}_N; t) = \frac{dG}{dt} = \sum_{i}^{N} m_i \dot{r}_i \cdot \ddot{r}_i = 0 . \quad (3.21) $$

Gauss’s principle states that the actual physical acceleration corresponds to the minimum value of $f$. If the system is not subject to a constraint then $f = 0$ and (3.19) reduces to the normal Newtonian equation of motion, otherwise the constrained equations of motion are obtained by finding the minimum of $f$ in (3.19), subject to the constraint, (3.21),

$$ \frac{\partial}{\partial \ddot{r}_i} (f - \zeta g) = 0 \quad (3.22) $$

$$ \frac{\partial}{\partial \ddot{r}_i} \left[ \frac{1}{2} \sum_{i=1}^{N} \left( \ddot{r}_i - \frac{F_i}{m_i} \right)^2 - \zeta \sum_{i}^{N} m_i \dot{r}_i \cdot \ddot{r}_i \right] = 0 \quad (3.23) $$

Differentiating this equation and performing the necessary algebraic manipulations, we obtain

$$ m_i \ddot{r}_i = F_i - \zeta m_i \dot{r}_i \quad i = 1, 2, 3, \ldots, N \quad (3.24) $$

where $\zeta$ is the Gaussian multiplier. Eq. (3.24) is a 2nd order differential equation, equivalent to the two first order differential equations:

$$ \dot{\mathbf{r}}_i = \frac{\mathbf{p}_i}{m_i} \quad (3.25) $$

$$ \dot{\mathbf{p}}_i = \mathbf{F}_i - \zeta m_i \dot{\mathbf{r}}_i \quad (3.26) $$
Substituting Eqs. (3.25) and (3.26) into the differential form of the constraint equation (3.21) yields the result

\[
\zeta = \frac{\sum_{i=1}^{N} \dot{r}_i \cdot F_i}{\sum_{i=1}^{N} m_i \dot{r}_i^2}
\]  

(3.27)

### 3.2 SLLOD algorithm

Gosling *et al.* [59] pioneered the use of nonequilibrium molecular dynamics (NEMD) by imposing a spatially periodic external force on the particles to generate an oscillatory velocity profile while maintaining conventional periodic boundary conditions. The amplitude of this velocity profile at steady state is inversely related to the shear viscosity, and hence the viscosity can be calculated. The shear viscosity determined in this manner is wavelength dependent. However, the Newtonian shear viscosity is obtained only in the long wavelength limit i.e. in the limit \( k \to 0 \), where \( k \) is the wave vector of the oscillatory perturbation. This means that a very large simulation box is required to get reasonably accurate values of shear viscosity, which limits the usefulness of this technique. The more successful NEMD techniques are the zero wave vector techniques which involve imposing a planar Couette flow velocity profile shown in Fig. 3.2. The most efficient direct NEMD algorithm for shear viscosity is the SLLOD algorithm [1].

For an atomic fluid the SLLOD equations of motion [1] are

\[
\dot{r}_i = \frac{p_i}{m_i} + r_i \cdot \nabla u
\]

(3.28)

\[
\dot{p}_i = F_i - p_i \cdot \nabla u - \zeta p_i
\]

(3.29)

where \( \zeta \) is the thermostating multiplier, \( u = (u_x, 0, 0) \), and \( u_x = \dot{\gamma} y \), where \( \dot{\gamma} \) is the strain rate; \( y \) is the direction normal to the direction of flow in the \( x \) direction. The SLLOD algorithm is combined with the Lees-Edwards “sliding brick” periodic boundary
Figure 3.2: Planar Couette flow velocity profile. \( \mathbf{u} = i \dot{\gamma} y \).

conditions [60], where the periodic images above and below the central simulation box (in the \( y \) direction) are moved in opposite directions. The strain rate dependent shear viscosity is obtained from the constitutive equation

\[
\eta(\dot{\gamma}) = -\frac{\langle P_{xy} \rangle}{\dot{\gamma}}
\]  

(3.30)

The Newtonian shear viscosity is estimated by extrapolating the shear viscosities to zero shear rate.
In this chapter we describe the parallel programming scheme used to implement our code with message passing interface (MPI). In the first section we provide a brief overview of the architectures that support programs running in parallel. In the second section we give a brief introduction to MPI. The third section demonstrates the frequently used MPI subroutines by analysing a sample of the parallelized program for Poiseuille flow which we have developed and implemented.

4.1 Parallel architectures

The main components in a typical computer system are the processor, memory, input/output devices, and the communication channels that connect them. The processor is the workhorse of the system; it is the component that executes a program by performing arithmetic and logical operations on data. It is the only component that creates new information by combining or modifying current information. In a typical system there will be only one processor, known as the central processing unit, or CPU. Modern high performance systems, for example vector processors and parallel processors, often have more than one processor. Systems with only one processor are serial processors, or, especially among computational scientists, scalar processors.

Memory is a passive component that simply stores information until it is requested by another part of the system. During normal operations it feeds instructions and data
to the processor, and at other times it is the source or destination of data transferred by I/O devices. Information in a memory is accessed by its *address*.

Input/output (I/O) devices transfer information without altering it between the external world and one or more internal components. I/O devices can be secondary memories, for example disks and tapes, or devices used to communicate directly with users, such as video displays, keyboards, and mice.

The communication channels that tie the system together can either be simple links that connect two devices or more complex *switches* that interconnect several components and allow any two of them to communicate at a given point in time. When a switch is configured to allow two devices to exchange information, all other devices that rely on the switch are *blocked*, i.e. they must wait until the switch can be reconfigured.

The architecture of parallel computers can be categorized in terms of two aspects: whether the memory is physically centralized or distributed, and whether or not the address space is shared. A shared memory architecture uses shared system resources such as memory and I/O subsystem that can be accessed equally from all the processors shown in Fig. 4.1. The processors communicate with one another by one processor writing data into a location in memory and another processor reading the data. The advantage of this type of architecture is that it is easy to program as there are no explicit communications between processors as communications are handled via the global memory store. However, a bottleneck happens when a number of processors attempt to access the global memory store at the same time.

The distributed memory architecture illustrated in Fig. 4.2 gives each processor its own memory. A processor can only access the memory which is attached directly to it. If a processor needs data which is contained in the memory of a remote processor, then it must send a message to the remote processor asking it to send the data. In spite of the drawback of explicit communications in the distributed memory architecture, it is
much faster to access the local memory and far more scalable than the shared memory architecture.

![Shared memory architecture](image1.png)

**Figure 4.1:** Shared memory architecture.

![Distributed memory architecture](image2.png)

**Figure 4.2:** Distributed memory architecture.
4.2 Introduction to MPI

MPI is a message passing interface used for parallel processing in distributed memory systems [61]. MPI is a library of routines that can be called from C, C++, FORTRAN77 and FORTRAN90 programs. A single user program is prepared, but is run on multiple processes. Each instance of the program is assigned a unique process identifier, so that it is labelled and known which process it is. This allows the same program to be executed, but different tasks are performed in each process. By convention, the user sets up one process as a master, and the others as workers, but this is not necessary. Each process has its own set of data, and can communicate directly with other processes by passing data around. Because the data is distributed, it is likely that a computation on one process will require that a data value be copied from another process. For example, if process $A$ needs the value of data item $X$ that is stored in the memory of process $B$, then the program must include the following lines.

```plaintext
if ( I am processor A ) then
  call MPI_Send ( X )
else if ( I am processor B ) then
  call MPI_Recv ( X )
end
```

We keep the lines as simple as possible to illustrate a number of important features of message passing interface.

4.2.1 Frequently used MPI subroutines

Frequently used environmental management and collective communication subroutines are introduced here.

4.2.1.1 Environmental management subroutines

Four environmental management subroutines, MPI_INIT, MPI_COMM_SIZE, MPI_COMM_RANK and MPI_FINALIZE are described.
CALL MPI_INIT(ierr)

Purpose
Initializes MPI.

Parameters
INTEGER ierr is the Fortran return code.

Description
This routine initializes MPI. All MPI programs must call this routine once and only once before any other MPI subroutine. In Fortran, the return code of every MPI subroutine is given in the last argument of its subroutine call. If an MPI subroutine call is done successfully, the return code is 0; otherwise, a non zero value is returned.
| CALL | MPI_COMM_SIZE(MPI_COMM_WORLD, nprocs, ierr) |

**Purpose**
Returns the number of processes belonging to the communicator specified in the first argument. A communicator is an identifier associated with a group of processes.

**Parameters**
- INTEGER *MPI_COMM_WORLD* is the communicator
- INTEGER *nprocs* is an integer used to specify the number of processes in the group.
- INTEGER *ierror* is the Fortran return code.

**Description**
This routine returns the size of the group associated with a communicator, *MPI_COMM_WORLD*, which is a MPI related parameter and defined in the header file *mpif.h*. All Fortran procedures that use MPI subroutines have to include this file.
### CALL

```
CALL MPI_COMM_RANK(MPI_COMM_WORLD, rank, ierror)
```

### Purpose

Returns the rank of the local process in the group associated with a communicator.

### Parameters

- **INTEGER** `MPI_COMM_WORLD` is the communicator.
  - INTEGER `rank` is an integer used to specify the calling processes in the group.
  - INTEGER `ierror` is the FORTRAN return code.

### Description

This routine returns the rank of the local process in the group associated with a communicator. Each process in a communicator has its unique rank, which is in the range from 0 to `(SIZE - 1)`, where `SIZE` is the number of processes in that communicator and it is the return value of `MPI_COMM_SIZE`. 
CALL MPI_FINALIZE(ierr)

Purpose Terminates all MPI processing.

Parameters INTEGER *ierr* is the Fortran return code.

Description This routine is the last MPI call. Any MPI calls made after MPI_FINALIZE raise an error. All pending communications involving a process have to be completed and all files opened by the process have to be closed before the process calls MPI_FINALIZE. Although MPI_FINALIZE terminates MPI processing, it does not terminate the process. It is possible to continue with non-MPI processing after calling MPI_FINALIZE, but no other MPI calls (including MPI_INIT) can be made.

4.2.1.2 Collective communication subroutines

A group of processes can exchange data by collective communication. The communicator argument in the collective communication subroutine specifies which processes are involved in the communication. In other words, all the processes belonging to that communicator must call the same collective communication subroutine with matching arguments. The commonly used patterns of collective communication are illustrated in Fig. 4.3 and discussed in the pages that follow.
Figure 4.3: Patterns of collective communication. (a) Message broadcast (see MPI_BCAST); (b) and (c) reduction operations (see MPI_REDUCE); (d) distribution of individual messages (see MPI_SCATTER).
**CALL**  
`MPI_BCAST(buffer, count, datatype, root, comm, ierror)`

<table>
<thead>
<tr>
<th>Purpose</th>
<th>Broadcasts a message from root to all processes in communicator, <code>comm</code>.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Parameters</td>
<td></td>
</tr>
</tbody>
</table>
  - `buffer`, the starting address of the buffer.  
  - `count`, the number of elements in the buffer.  
  - `datatype`, the data type of the buffer elements.  
  - `root`, the rank of the root process.  
  - `comm`, the communicator.  
  - `ierror`, the Fortran return code. |
| Description | This routine broadcasts a message from root to all processes in `comm`. The contents of root buffer is copied to all processes. The amount of data sent must be equal to the amount of data received. All processes in `comm` need to call this routine. |
CALL MPI_SCATTER(sendbuf, sendcount, sendtype, recvbuf, recvcount, recvtype, root, comm, ierror)

### Purpose
Distributes individual messages from root to each process in *comm.*

### Parameters
- `sendbuf`, the starting address of the send buffer.
- `sendcount`, the number of elements to be sent to each process, not the number of total elements to be sent from root.
- `sendtype`, the data type of the send buffer elements.
- `recvbuf`, the starting address of the receive buffer.
- `recvcount`, the number of elements in the receive buffer.
- `recvtype`, the data type of the receive buffer elements.
- `root`, the rank of the sending process.
- `comm`, the communicator.
- `ierror`, the Fortran return code.

### Description
This routine distributes individual messages from root to each process in *comm.* The number of elements sent to each process is the same (*sendcount*). The first *sendcount* elements are sent to process 0, the next *sendcount* elements are sent to process 1, and so on. This routine is the inverse operation to MPI_GATHER. The amount of data sent must be equal to the amount of data received. All processes in *comm* need to call this routine.
CALL MPI_REDUCE(sendbuf, recvbuf, count, datatype, op, root, comm, ierror)

**Purpose**
Applies a reduction operation to the vector `sendbuf` over the set of processes specified by `comm` and places the result in `recvbuf` on root.

**Parameters**
- `sendbuf`, the starting address of the send buffer.
- `recvbuf`, the starting address of the receive buffer.
- `INTEGER count`, the number of elements in the send buffer.
- `INTEGER datatype` The data type of elements of the send buffer.
- `INTEGER op`, the reduction operation.
- `INTEGER root`, the rank of the root process.
- `INTEGER comm`, the communicator.
- `INTEGER ierror`, the Fortran return code.

**Description**
This routine applies a reduction operation to the vector `sendbuf` over the set of processes specified by `comm` and places the result in `recvbuf` on root. Both the input and output buffers have the same number of elements with the same type. The arguments `sendbuf`, `count`, and `datatype` define the send or input buffer and `recvbuf`, `count` and `datatype` define the output buffer. MPI_REDUCE is called by all group members using the same arguments for `count`, `datatype`, `op`, and `root`. All processes in `comm` need to call this routine.
4.3 Sample of parallelized program for Poiseuille flow

In this section, we explain how we parallelize the program for Poiseuille flow. The details we consider are kept as simple as possible to focus on the important features of MPI. In order to use MPI to explicitly pass messages between parallel processes, we must add message passing constructs to our program. To enable our programs to use MPI, we must include the MPI header file, `mpif.h`, in our source code `MPI.f`.

1. PROGRAM MPI
2. INCLUDE ‘poisflow.inc’
3. INCLUDE ‘/usr/include/mpif.h’
4. INTEGER ierror, NP, ID
5. CALL MPI_INIT(ierr)
6. CALL MPI_COMM_SIZE(MPI_COMM_WORLD, NP, ierr)
7. CALL MPI_COMM_RANK(MPI_COMM_WORLD, ID, ierr)
8. CALL poisflow(ID, NP)
9. CALL datacollect(ID, NP)
10. CALL MPI_FINALIZE(ierr)
11. END
Note that the program is executed in the SPMD (Single Program Multiple Data) model. All the nodes that run the program, therefore, need to see the same executable file with the same path name. The input file is located on a shared file system and each process reads data from the same file. The input file on a shared file system is shown in Fig. 4.4.

**Figure 4.4:** The input file on a shared file system.

- **Line 1:** MPI program.
- **Line 2:** Includes poisflow.inc, which defines parameters and global variables for the subroutine poisflow called in Line 8.
- **Line 3:** Includes mpif.h, which defines MPI-related parameters such as MPI_COMM_WORLD. All Fortran procedures that use MPI subroutines have to include this file.
Line 4: Declares INTEGER variables. NP is the number of processes, ID is the rank of the process and ierror is the Fortran return code.

Line 5: Calls the environmental subroutine, MPI_INIT, to initialize an MPI environment. MPI_INIT must be called once and only once before calling any other MPI subroutines.

Line 6: Calls the environmental subroutine, MPI_COMM_SIZE, to return the number of processes belonging to the communicator, MPI_COMM_WORLD specified in the first argument and defined in mpif.h. The communicator represents the group consisting of all the processes participating in the parallel job.

Line 7: Calls the environmental subroutine, MPI_COMM_RANK, to return the rank of the process within the communicator given as the first argument.

Line 8: Calls the subroutine poisflow performing the NEMD simulations for Poiseuille flow. The number of processes and the identifier of the current process are passed to the subroutine poisflow by two parameters. The program runs independently on different processes. In the subroutine poisflow, the initial velocities of all
the particles are generated with the random number which relates to the ID of the process. This means the different processes run the same program with different initial random conditions for the momenta and avoids getting exactly the same results.

Line 9: Calls the subroutine datacollect to collect the data from all the processes. This subroutine will be analysed next.

Line 10: Calls MPI_FINALIZE to terminate MPI processing and no other MPI call can be made after this subroutine. Ordinary Fortran code can follow MPI_FINALIZE.
Subroutine *datacollect* is called in *MPI.f* to collect the data from all the processes.

```fortran
1. SUBROUTINE datacollect (ID,NP)
2. INCLUDE ‘poisflow.inc’
3. INCLUDE ‘/usr/include/mpif.h’
4. DOUBLE PRECISION temp(1:N)
5. INTEGER ID, NP, ierror, i
6. CALL MPI_REDUCE (data(1), temp(1), N, MPI_DOUBLE_PRECISION,
                   & MPI_SUM, 0, MPI_COMM_WORLD, ierror)
7. do 60 i=1, N
8.   data(i)=temp(i)/NP
9. 60 CONTINUE
10. if (ID.eq.0) then   CALL output
11. RETURN
12. END
```

**Line 1:** Subroutine *datacollect* which is called in MPI program.
Line 2: Includes `poisflow.inc`, which defines parameters and global variables for the subroutine `poisflow` called in MPI program.

Line 3: Includes `mpif.h`, which defines MPI-related parameters such as `MPI_COMM_WORLD`. All Fortran procedures that use MPI subroutines have to include this file.

Line 4: Declares DOUBLE PRECISION one dimensional array `temp` with the size, $N$.

Line 5: Declares INTEGER variables. `NP` is the number of processes, `ID` is the rank of the process and `ierror` is the Fortran return code.

Line 6: Calls the collective communication subroutine, `MPI_REDUCE` to apply a reduction operation, `MPI_SUM`, to the array `data` which is declared in `poisflow.inc` over the set of processes specified by the communicator, `MPI_COMM_WORLD`, and places the result in the array `temp` on root 0. Both the input and output buffers have the same number of elements, $N$, with the same data type specified as `MPI_DOUBLE_PRECISION`, which is one of the MPI data types. MPI provides several common operators by default, where `MPI_SUM` is one of them defined in
4.4 Running a parallelized program

mpif.h. MPI_REDUCE is called by all group members using the same arguments. All processes in the communicator need to call this routine.

Line 7: Do loop over all the elements in the array temp.

Line 8: Calculates averages.

Line 9: Continues the loop.

Line 10: Checks if the process is root. If the process is root, write the data to output file, as the results are put in the array on root 0.

When an MPI program is written in the C language, there are some points which one should be aware of:

- Include mpi.h instead of mpif.h.
- C is case-sensitive. All of the MPI functions have the form MPI_FUNCTION, where MPI and the function name are in upper-case as in MPI_INIT. Constants defined in mpi.h are all in upper-case such as MPI_INT, MPI_SUM, MPI_COMM_WORLD.
- Those arguments of an MPI function call that specify the address of a buffer have to be given as pointers.
- Predefined MPI data types in C are different from Fortran bindings.

4.4 Running a parallelized program

In this section we explain how to compile, debug and run an MPI program.

Compiling and linking:

For Fortran, compile with a command like:
% f90 mpi.f -o mpi.exe -lmpi -lelan

For C, compile with a command similar to:

% cc mpi.c -o mpi.exe -lmpi -lelan

When compiling the MPI program, we must link to the MPI libraries, by using -lmpi. We must also link to the ELAN library, a communications library used for internode communication, by including -elan in our command line on the AlphaServer SC, the machine based at the APAC facilities at the Australia National University in Canberra, where most of our simulations were performed.

Running MPI jobs:

It is necessary to use the prun command to start an MPI executable, both when running within a batch job and when running small interactive test jobs. To run a small test with 4 processes (or tasks) where the MPI executable is called mpi.exe, enter the command:

% prun -n 4 mpi.exe

The argument to -n is the number of mpi.exe processes required to run. For larger jobs and production use, submit a job to the PBS batch system which allows access to all the processors on the machine. The batch file, com-batch, looks like:

```bash
#!/bin/csh
# PBS -P f15
# PBS -q normal
# PBS -l walltime=15:51:23, vmem=450MB, ncpus=4
# PBS -r n
# PBS -wd
prun -n 4 mpi.exe
```
Most jobs will require greater resources than are available to interactive processes. Larger jobs must be scheduled by the batch job system. We submit jobs to the portable batch system (PBS) specifying the project name, number of CPUs, the amount of memory, and the length of time needed. PBS runs the job when the resources are available, subject to constraints on maximum resource usage.

**Options of note:**

-P f15

The project, f15, which you want to charge the jobs resource usage to.

-q normal

The systems have a simple queue structure with three levels of priority, normal, express and bonus. The queue names reflect their priority.

-l walltime = 20:00:00

The wall clock time limit for the job. Time is expressed in seconds as an integer, or in the form:

hours : minutes : seconds.

-l vmem = 450MB

The total (virtual) memory limit for the job. It can be specified with units of “MB” or “GB” but only integer values can be given. A job will only run if there is sufficient free memory so making a sensible memory request will allow jobs to run sooner. A little trial and error may be required to find how much memory the jobs are using. nqstat lists jobs actual usage.

-l ncpus = 4

The number of CPUs required for the job to run on. The default is 1.
-r n

Specifies the job is not restartable, and if the job is executing on a node when it crashes, the job will not be requeued.

-wd

Start the job in the directory from which it is submitted. Normally jobs are started in the users home directory.

prun command is used to start an MPI executable, mpi.exe. By not specifying the -n option with the batch job, prun will start as many MPI processes as there have been CPUs requested with qsub. It is possible to specify the number of processes on the batch job prun command, as for the interactive case. To submit jobs to the queues, the simplest use of the qsub command is as follows

% qsub com-batch

Debugging a program that executes in parallel can be quite difficult. If the program crashes, for instance, each process will create a separate core file, in its own directory, with a name like coredir.1. dbx debugger can be used to query each core file by itself. If we can assume that the executable program is called mpi.exe, and that we want to look at the core file associated with process 4, and we just want to get a traceback of “where” process 4 was when it crashed, it would be enough to type:

```
  dbx mpi.exe coredir.4/core
  where
  quit
```

Command where is used to get a traceback of the point where the process crashed.
In this chapter we explain the general simulation considerations and present our simulation results for atomic fluids. In Chapter 2 we derived the pressure tensor and heat flux vector for inhomogeneous fluids under the influence of three-body forces. In order to validate our derivation, we perform nonequilibrium molecular dynamics simulations of a confined fluid acted upon by a two-body Barker-Fisher-Watts (BFW) force coupled with the Axilrod-Teller (AT) three-body force. Our method of planes calculations agree perfectly with the equivalent mesoscopic route of integrating the momentum and energy continuity equations directly from the simulation data. Our calculations reveal that three-body forces on simple liquids like argon have an important consequence for the isotropic pressure, but have negligible influence on the shear stress and strain rate (hence viscosity) and heat flux vector (hence thermal conductivity). This chapter is based on work published in [44].

5.1 General simulation considerations

The geometry of our simulation cell is shown in Fig. 5.1 and Fig. 5.2. An atomic fluid is confined between atomistic walls as depicted. Our geometry is such that $y = 0$ defines the centre of the fluid channel. A constant field drives the fluid and is directed in the $x$ direction. Each wall is three atomic layers thick, and the second wall is just the periodic image of the first. The entire cell is thus periodic in $x$, $y$, and $z$. The total number
of atoms is $N = 324$, which includes 270 liquid atoms and 54 wall atoms ($N_w = 18$ atoms per layer). In what follows all quantities are expressed in terms of Lennard-Jones reduced units defined in terms of $\sigma$, $\varepsilon$ and $m$: reduced distance $r_{ij}^* = r_{ij}/\sigma$, the density $\rho^* = \rho\sigma^3$, the temperature $T^* = k_B T/\varepsilon$, the energy $U^* = U/\varepsilon$, the time $t^* = t/(\sigma\sqrt{m/\varepsilon})$, the pressure $P^* = P\sigma^3/\varepsilon$, the strain rate $\dot{\gamma}^* = \sqrt{m\sigma^2/\varepsilon}\dot{\gamma}$, and the viscosity $\eta^* = \sqrt{\sigma^4/(m\varepsilon)}\eta$. For simplicity of notation, hereafter the asterisk will be omitted. We set $\sigma = \varepsilon = m = 1$. The density of the fluid is 0.44 and the wall density is 0.84. The cell dimensions are $L_x = 5.0565$, $L_y = 27.5143$, $L_z = 5.0565$. The thickness of the walls is $\Delta y_w = 2.5143$. The pore width, $L$, is defined as the distance between the centres-of-mass of the two innermost opposing wall layers. A cutoff potential radius of $L_x/2 = 2.5282$ is used for the two-body force calculation, whereas a value of $L_x/4 = 1.2641$ is used for the three-body force. These are optimal values, based upon the work performed in [56]. In our simulations we use a truncated and shifted version of the BFW and AT potentials, so that long-range corrections need not be considered. In this way the potential is zero at and beyond the cutoff value. We justify this as our goal is to verify the MOP expressions derived in Chapter 2 for the pressure tensor and heat flux vector, rather than to accurately reproduce experimental results.

Our simulations are performed on fluid and solid atoms that interact via the Barker-Fisher-Watts two-body potential [51] and Axilrod-Teller three-body potential [52] introduced in Chapter 3. The equations of motion used to simulate wall and fluid atoms were developed in [45] and quoted here as follows, modified for three-body forces. We note that wall atoms interact via the two plus three body forces in addition to a harmonic spring force that tethers them together. For the wall particles,

$$\dot{r}_i = \frac{p_i}{m} \quad (5.1)$$

$$\dot{p}_i = -\kappa (r_i - q_i) + F_i^{(2)} + F_i^{(3)} - \alpha p_i - j\lambda L_n, \quad i \in L_n \quad (5.2)$$
Figure 5.1: Geometry of the simulation environment. (a) Overall system including fluid and wall atoms (z-direction normal to page); (b) wall atoms in the $xz$ plane, defining separation of layers ($y$-direction normal to page).
where \( \kappa \) is the spring force constant and is set to 57.15 in the simulations. \( \mathbf{q}_i \) is the equilibrium “frozen” position of atom \( i \). \( \mathbf{r}_i \) is the laboratory position of atom \( i \) and \( \mathbf{p}_i \) here refers to the laboratory momentum of atom \( i \). As the walls are not under the influence of a driving field the peculiar (thermal) and laboratory momenta are equivalent. \( \mathbf{j} \) is the unit vector in the \( y \)-direction and the layer multiplier \( \lambda_{Ln} \) ensures that the centre of mass of each wall layer stays fixed, where the index \( n = 1, 2, 3 \), refers to the three wall layers. This is important otherwise the walls separate as the fluid heats up under flow. \( \alpha \) is a thermostat multiplier used to keep the temperature of the walls fixed (in our simulations the wall temperature is fixed at 0.722). The layer multiplier and thermostat are computed as

\[
\lambda_{Ln} = \frac{\mathbf{j}}{N_w} \sum_{i \in L_n} N_w \left[ -\kappa (\mathbf{r}_i - \mathbf{q}_i) + F^{(2)}_i + F^{(3)}_i \right] 
\]

(5.3)

where

\[
\sum_{L_n=1}^{3} \sum_{i \in L_n} N_w = 3N_w
\]

and

\[
\alpha = \frac{\sum_{i \in L}^{3N_w} \left[ -\kappa (\mathbf{r}_i - \mathbf{q}_i) + F^{(2)}_i + F^{(3)}_i - \mathbf{j}\lambda_{Ln} \right] \cdot \mathbf{p}_i}{\sum_{i \in L}^{3N_w} \mathbf{p}_i^2} 
\]

(5.4)
Here $L = \{L_1, L_2, L_3\}$ representing the wall layers, and $N_w$ is the number of wall atoms.

The fluid atoms obey Newton’s equations of motion

\[ \dot{r}_i = \frac{p_i}{m} \]  
\[ \dot{p}_i = F^{(2)}_i + F^{(3)}_i + iF_e \]

(5.5) \hspace{1cm} (5.6)

where $F_e$ is the external driving field and $i$ is the unit vector in the $x$-direction and we again note that $r_i$ and $p_i$ refer to the laboratory position and momentum of atom $i$ respectively. In our simulations the field strength used is $F_e = 0.2$.

The equations of motion are solved with a fifth order Gear predictor-corrector scheme with an integration time step of $\tau = 0.001$. Our simulations are first run for a total of $10^6$ time steps to reach a nonequilibrium steady-state. Once steady-state is achieved, production runs of a total of $10^6$ time steps are run with averages accumulated in blocks of 50000 time steps. For clarity of visualisation, we do not put error bars on most of the simulation data throughout this thesis as they are the size of the plotting symbol.

In our simulations we do not assume any functional form for the streaming velocity. Rather, we first run a steady-state simulation of $10^6$ time steps and compute a time-averaged velocity profile at planes, using the procedure developed in [62]. These plane velocity values are then used as the streaming velocity $u_x(y)$ in MOP calculations of the pressure tensor and heat flux vector in all subsequent production runs.

Finally, we note that a total of 200 planes are used in the MOP calculations, though not all planes data are plotted on the figures presented in this work for clarity of visualisation. The further details of the simulation methodology can be referred from [45, 46, 62].

5.2 Simulation results

5.2.1 Number density and streaming velocity

Strongly oscillatory surface forces measured with the Surface Forces Apparatus (SFA) were first reported for small molecules confined between atomically smooth mica plates [63].
The existence of such oscillating surface forces suggested that confined fluids are inhomogeneous near the surface. In our simulation, we observe density fluctuations in the vicinity of the solid surfaces, in the form of layers, shown in Fig. 5.3. The layered structure near the solid walls has the same physical origin as the oscillations in the radial distribution function of simple liquids [64, 65]. It can be understood easily for the reference system of a hard sphere fluid next to a hard wall. For this system the inter particle interactions are characterized by the complete absence of any attraction. The actual hard wall - hard sphere potential is infinite when a hard sphere penetrates the wall and zero otherwise. The effective hard wall - hard sphere interaction in the presence of the other hard spheres is different [64]. When a hard sphere is adjacent to the wall then the collisions with other fluid particles from the opposite direction of the wall are much more frequent than from the side of the wall. As a result, the rest of the fluid particles give rise to a net force pushing this particle towards the wall. Therefore the particles touching the surface are attracted to it due to the asymmetry of their collisions with the other fluid particles. This idea was exploited to explain the oscillations of the radial pair correlation function [64]. In Fig. 5.4 we plot the streaming velocity profile. It is seen to be well represented by a symmetric quadratic function in $y$, when the channel width is more than 10 atomic diameters, in conformity with hydrodynamics [45]. It is well known that hydrodynamics breaks down at smaller channel widths [9].

5.2.2 Pressure tensor

Before we present our simulation results, the methods exploited to compute the pressure tensor and strain rate are briefly outlined.

Elements of the pressure tensor $P_{yy}$ and $P_{xy}$ are computed by the method of planes (MOP). $P_{yy}$ and $P_{xy}$ for the two-body potential are computed by Eqs. (2.36) and (2.45) and those for three-body potential are computed by Eqs. (2.40) and (2.45). All these
equations were derived in Chapter 2. We compare our method of planes calculations of $P_{xy}$ with the equivalent mesoscopic route of integrating the momentum continuity equation (IMC) [45]. The IMC method is derived as follows. We first write down the momentum continuity equation in the form

$$\rho(r, t) \frac{du(r, t)}{dt} = -\nabla \cdot \Pi - \nabla p + \rho(r, t)F_e \tag{5.7}$$

In this equation $\rho(r, t)$ is the mass density at position $r$ and time $t$, $u(r, t)$ the fluid streaming velocity at $r$ and $t$, and $\Pi$ the viscous pressure tensor (i.e. $\Pi = P - pI$, where $P$ is the pressure tensor and $p$ the hydrostatic pressure). It is the external force $F_e$ which drives the flow.

**Figure 5.3:** Number density profile for the fluid system.
Figure 5.4: Streaming velocity profile (circle data points) for the fluid. Superposed (solid curve) is a symmetric quadratic fit, in conformity with hydrodynamic prediction. Error bars are the size of the plotting symbol.

For our geometry (in the absence of a microscopic pressure gradient), shown in Fig. 5.1, Eq. (5.7) reduces to

\[
\frac{\partial \Pi_{xy}}{\partial y} = \rho(y) F_e \tag{5.8}
\]

The solution to Eq. (5.8) is

\[
\Pi_{xy} = F_e \int_0^y dy' \rho(y') \tag{5.9}
\]

In Fig. 5.5 we plot MOP calculations of \( P_{yy} \), the \( y \)-component of the pressure in the direction normal to the wall surface, for both the BFW fluid and the BFW fluid with the inclusion of the AT three body forces. For mechanical stability \( P_{yy} \) must be constant.
5.2 Simulation results

Figure 5.5: $P_{yy}$ as a function of $y$ for the BFW (squares) and BFW+AT (circles) fluids. The pressure is computed by the method of planes. Also shown is the pressure at the walls (crosses for BFW+AT and diamonds for BFW).

Throughout the channel, and this is indeed seen to be the case. Also shown is the pressure calculated on the system walls. This is computed from the total $y$ component of the force per unit area exerted on the wall atoms by fluid atoms on one side of the wall. We see perfect agreement between wall and fluid $P_{yy}$ values, as expected. Clearly the addition of the three-body force significantly affects the $P_{yy}$. Neglecting to include three-body forces overestimates the value by almost 11%.

Of greater interest to us is the shear stress ($-P_{xy}$). In Fig. 5.6 we plot $P_{xy}$ as a function of $y$ for the BFW and BFW+AT fluids. We show the results of our MOP calculations (circle data points for BFW+AT fluids and square for BFW fluids) and compare them with direct integration of the momentum continuity equation, the IMC method given by Eq. (5.9) (solid line for BFW+AT fluids and dash line for BFW fluids). Error bars are of the order of the size of the plotting symbols. We find excellent
agreement between both methods, demonstrating that the MOP calculations are correct. The value of $P_{xy}$ calculated at the walls (cross data points) is also included and seen to be consistent with both the MOP and IMC values. Note that the stress deviates from the linear hydrodynamics prediction close to the walls, as is to be expected for such a confined system which is inhomogeneous in the vicinity of the walls. From Fig. 5.6 it is clear that three-body forces have negligible effect on the shear stress. This is seen more clearly in Fig. 5.7 in which the region between $5.7 \leq y \leq 6.1$ is magnified. This is consistent with the observations reported in [66] that showed that three-body forces only affected the shear viscosity by approximately 3% for a bulk system under planar Couette flow. The precise degree of influence is likely to depend on temperature and density, and
5.2 Simulation results

Figure 5.7: As with Fig. 5.6 but magnified in the range between $5.7 \leq y \leq 6.1$.

Error bars are of the order of plotting symbol sizes.

to a lesser degree the number of atoms, so we do not say anything further at this stage. We also note that our potential is truncated and shifted for numerical expediency and this also will affect the longer range attractive contributions to the shear stress.

5.2.3 Heat flux

In Fig. 5.8 we plot the heat flux vector as a function of $y$ for the BFW and BFW+AT fluids. Classical hydrodynamics predicts a cubic heat flux profile. We again show the results of our MOP calculations of $J_{Qy}$ computed by Eqs. (2.59) and (2.60) derived in Chapter 2 and compare them with direct integration of the energy continuity equation,
the IEC method of [46] given as
\[
J_{Q_y}(y) = - \int_0^y dy' P_{xy}(y') \dot{\gamma}(y'),
\] (5.10)
where \( \dot{\gamma}(y) \) is the strain rate,
\[
\dot{\gamma}(y) = \frac{\partial u_x(y)}{\partial y}.
\] (5.11)

Once again, excellent agreement is found between both methods, confirming the validity of the MOP expressions. For a channel this size it is clear that the classical cubic heat flux profile is obeyed. The value of the heat flux at the walls is also computed by noting that the Gaussian thermostat acting on the walls removes heat at a rate of
\[
\dot{Q}(t) = \alpha(t) \sum_{i=1}^N \frac{p_i^2}{m}.
\] (5.12)
The heat flux at the walls is therefore
\[
\langle J_{Q_y}(y = y_{\text{wall}}) \rangle = \frac{1}{A} \langle K_w \alpha \rangle,
\] (5.13)
where \( K_w \) is the kinetic energy of the wall atoms and the angle brackets indicate a time average [46]. The heat flux at the walls is in excellent agreement with the MOP and IEC values at the wall-fluid interface.

We again observe that the presence of three-body forces has very little influence in the transportation of energy across the channel. In Fig. 5.9 the region between \( y \leq 7.0 \) is magnified. Three body forces contribute a very small but noticeable effect on the heat flux, slightly increasing its magnitude. Again, this effect is likely to be temperature, density and potential dependent and we refrain from specific conclusions here. What is important for our purposes is the excellent agreement between the MOP and IEC methods.

Finally, in Fig. 5.10 we plot several contributions to the three-body component of the heat flux vector. Term 1 (circles) represents the second and third terms in Eq. (2.58). These terms are the direct three-body analogy of the two-body contribution to the heat...
Figure 5.8: Heat flux vector $J_{Qy}(y)$ as a function of $y$ for the BFW and BFW+AT fluids. $J_{Qy}(y)$ computed by the MOP and IEC methods are shown, as well as $J_{Qy}(y)$ computed at the walls.

It is clear that these last three terms are negligible, if not zero.
Figure 5.9: As with Fig. 5.8 but magnified in the range between $4.0 \leq y \leq 7.0$.

Error bars are of the order of plotting symbol sizes.
Figure 5.10: Individual terms of the three-body contribution to the heat flux vector.

Term 1 represents the second and third terms of Eq. (2.58) and is the direct analogy of the two-body contribution to the heat flux vector. Term 2 represents the last three terms in Eq. (2.58).
Non-local Linear Hydrodynamic Constitutive Model

In this chapter, we use the non-local linear hydrodynamic constitutive model, proposed by Evans and Morriss [1], for computing a viscosity kernel, a function of compact support, for inhomogeneous nonequilibrium fluids. This chapter is set out as follows. In the first section we revise the Navier-Stokes theory for our simulation geometry. In the second section we explain the model for computing a non-local viscosity kernel. In the third section we describe the numerical details of the extraction of the viscosity kernel in Eq. (6.8). In the fourth section we validate our Discrete Fourier Transforms (DFTs) which are used in this work. In section five we give specific details of the simulations. In section six we present our simulation and DFT results and discuss them at length. In section seven we explain the effect of domain restriction on inverse convolution. Finally, in the eighth section we show the problems associated with the analytic Fourier Transform. The work in this chapter is based partly on work published in [67].

6.1 Navier-Stokes theory for fluids in slit pores

The momentum continuity equation describing the flow of a single component fluid under the influence of an external field per unit mass, $F_e$, is given by

$$\rho \frac{du_x(y)}{dt} = -\frac{dP_{yx}(y)}{dy} + \rho F_e$$

(6.1)
where $\rho$ is the mass density, $P_{yx}$ is the $yx$ component of the viscous pressure tensor and the confinement is in the $y$-direction while the external field acts in the $x$-direction. The corresponding Navier-Stokes equation is then obtained by substituting Newton’s law of viscosity

$$P_{yx}(y) = -\eta \dot{\gamma}(y)$$

into the fluid equation of motion, and noting that at low Reynolds number and in the steady-state $du_x(y)/dt = 0$, i.e.,

$$\eta \frac{d\dot{\gamma}(y)}{dy} + \rho F_e = 0$$

The Navier-Stokes equation is trivial to solve for a one-component fluid subject to gravity flow in a planar channel. Two key assumptions must be made however. The first is that the density is uniform throughout the fluid. The second assumption refers to the boundary conditions. A common boundary condition is the stick boundary condition (no slip) which states that the $x$-component of velocity must vanish at the walls, i.e., $u_x(y = \pm h) = 0$, where $h$ is the distance from the centre of the channel to the walls. With these assumptions, the solution for the streaming velocity is

$$u_x(y) = -\frac{h^2 \rho F_e}{2\eta} (\bar{y}^2 - 1)$$

(6.4)

where $\bar{y} = y/h$.

The form of the streaming velocity can alternatively be determined from NEMD simulations of simple fluids undergoing Poiseuille flow. Here, the local velocity is obtained from its definition

$$J = \rho u$$

(6.5)
where $J$ is the momentum density, and no assumptions are made in this case. The momentum and mass densities are obtained using their microscopic definitions

$$J_x(y) = \frac{1}{A} \sum_i m_i v_{xi} \delta(y - y_i) \quad (6.6)$$

$$\rho(y) = \frac{1}{A} \sum_i m_i \delta(y - y_i) \quad (6.7)$$

with the Dirac delta functions replaced by functions which are finite in some small region of $y$, i.e., the data is simply collected in bins of thickness $\Delta y$ and area $A$ in the $xz$-plane. Here $m_i$ and $v_{xi}$ are the mass and $x$ component of the velocity of particle $i$ respectively. Comparison of the simulated streaming velocity profiles with that predicted by Eq. (6.4) gives an indication of the accuracy of Navier-Stokes theory for high degrees of confinement. Work performed in [9, 45, 68] have made these comparisons and determined that for channels of the order of 10 atomic diameters in width or more, the Navier-Stokes predictions are largely correct if one allows for some slip at the walls. Below these pore widths, the velocity profiles develop significant fine structure.

The fluid viscosity can be determined within an NEMD simulation if one assumes a form for the constitutive relation. No other assumptions are made. The shear stress can be obtained by direct integration of the density profile followed by multiplication by the external force field magnitude (see Eq. (5.9)), while the strain rate profile can be obtained by numerical differentiation of the streaming velocity data (or, alternatively, by analytic differentiation of a functional fit to the velocity data). Departure from the quadratic behaviour depicted by Eq. (6.4) indicates the limitations of assuming Newton’s law of viscosity, while a viscosity which is not uniformly finite implies the breakdown of assuming a local version of Newton’s law.
6.2 Non-local constitutive equation

A central problem in the study of confined fluids is the computation of meaningful transport coefficients. For confined fluids undergoing flow at low Reynolds number, classical Navier-Stokes hydrodynamic theory adequately describes the rheological properties of the fluid, provided the length scale of the confinement is much greater than the dimension of the fluid molecules. However, once the confinement approaches molecular dimensions, Navier-Stokes theory must be generalised to allow for local, position dependent transport coefficients. Computer simulations of simple fluids undergoing planar Poiseuille flow have demonstrated the validity of the local generalisation of Navier-Stokes hydrodynamics [45]. However, very careful simulations at quite high confinement of the order of 5 atomic diameters or less have shown that even the local generalisation becomes invalid at this length scale [9, 68]. A local shear viscosity, for instance, relates the local strain rate at a point in the fluid to the local stress at the same point. In the simulations (which use NEMD) it is possible to obtain both stress and strain rate profiles with high spatial resolution without any assumptions being made regarding the transport coefficients. The ratio of these two quantities was then used to calculate a shear viscosity. At very high degrees of confinement, the strain rate profile contained zeros resulting from stationary points in the fluid streaming velocity profile. The stress profile, on the other hand, contained only one zero, located at the mid point of the channel. Clearly, a viscosity obtained from the ratio of these quantities will involve singularities. This unsatisfactory result suggests further generalisation of Navier-Stokes hydrodynamics is necessary. The need for an improved hydrodynamic theory is paramount given the increasing drive towards nanoscale technologies and applications such as nanofluidics, and the desire to have a unified theory of fluids, able to describe phenomena occurring at all length and timescales. Experimental results involving the atomic force microscope
Non-local Linear Hydrodynamic Constitutive Model

(AFM) [69, 70, 71] indicate that very thin fluid films exhibit unusually high viscosities under confinement. An explanation for this phenomenon remains elusive.

Equilibrium and nonequilibrium molecular simulations of nanofluids have continued to escalate since the pioneering work done in the 1980s. However in most of these simulations the viscosity was only treated heuristically, often computing effective viscosities based on local measures rather than a genuine non-local viscosity [3, 72].

The first statistical mechanics theory that rigorously attempted to solve this problem for nonequilibrium fluids was the Enskog-like theory of Davis [4, 5]. While successful, it was limited to the case of uni-directional inhomogeneity and several other approximations, such as the substitution of the pair correlation function for the inhomogeneous fluid by a corresponding function for the homogeneous fluid at a smoothed density. This work was later extended by Pozhar and Gubbins [40, 41, 73] who developed a theory of transport, again based on a modified Enskog theory for inhomogeneous fluids [74]. The major limitation to this theory is that it relates local values of transport coefficients to integrals over equilibrium inhomogeneous singlet and pair correlation functions. It is unlikely that the theory will hold when systems are far from equilibrium. A comparison of the Pozhar-Gubbins (PG) theory was made with NEMD simulations of a fluid undergoing planar Poiseuille flow [42]. However, the computation of local NEMD viscosities, defined as the ratio of the shear stress to the strain rate, was already known to be a severe approximation, leading to poor statistical accuracy as the strain rate approaches zero.

Some time ago it was suggested that the linear constitutive relations could be further generalised by invoking non-local transport coefficients [1]. The non-local generalisation of Newton’s law of viscosity is then

\[
P_{yx}(\mathbf{r}, t) = - \int_0^t ds \int_{-\infty}^{\infty} d\mathbf{r}' \eta(\mathbf{r} - \mathbf{r}', t - s) \dot{\gamma}(\mathbf{r}', s),
\]

(6.8)
where $\eta$ and $\dot{\gamma}$ are the viscosity kernel and strain rate respectively, and we have ignored any explicit dependencies of these quantities on the local, intensive, thermodynamic state variables. Though not stated by Evans and Morriss [1], this form of the expression is exactly valid only for a system which is homogeneous in space because there is no explicit density dependence in the viscosity kernel. However, for systems that are strongly inhomogeneous the viscosity kernel should be written as $\eta(r, r - r', t - s)$. As yet there is no clear way to extract this viscosity kernel for an inhomogeneous system such as fluids in nanopores, and so in what follows we drop the explicit $r$ dependence. In this case the viscosity kernel in Eq. (6.8) is an effective viscosity kernel.

The simple proportionality between stress and strain rate implicit in Newton’s law and its local generalisation has been replaced by a double convolution in space and time so that the stress at position $r$ and time $t$ is not simply proportional to the strain rate at position $r$ and time $t$, but is proportional to the entire strain rate history, and strain rate distribution across the fluid. What this means in practical terms is that all fluids have a memory effect and therefore behave viscoelastically.

### 6.3 The method employed for solving the model

At first sight, Eq. (6.8) appears somewhat unhelpful. However, in $k\omega$-space, the double convolution becomes algebraic:

$$\tilde{P}_{yx}(k, \omega) = -\tilde{\eta}(k, \omega)\tilde{\dot{\gamma}}(k, \omega)$$

where the tildes denote the Fourier-Laplace Transform of a function defined for an arbitrary function $A(r, t)$ by

$$\tilde{A}(k, \omega) = \int_0^\infty dt \int_{-\infty}^{\infty} dr \exp[-i(k \cdot r + \omega t)]A(r, t)$$
If the strain rate does not vary appreciably in time compared with the time scale for molecular motions, the zero frequency limit of Eq. (6.9) may be taken, giving

\[ \tilde{P}_{yx}(k) = -\tilde{\eta}(k)\tilde{\gamma}(k) \]  

(6.11)

where the missing frequency arguments imply \( \omega = 0 \). Eq. (6.11), which defines the wave vector dependent viscosity, has been used previously to extract this quantity for bulk inhomogeneous fluids via the sinusoidal transverse force (STF) method [75]. In the STF method the inhomogeneities are introduced artificially via a sinusoidally varying external field which gives rise to an oscillatory density profile.

In the case of a homogeneous fluid we can represent the viscosity kernel as \( \eta(r - r') \equiv \eta_0 \delta(r - r') \), where \( \eta_0 \) is a constant. The spatial localisation is introduced into Eq. (6.8) by means of the delta function. It is clear that for a spatially homogeneous system Eq. (6.8) simply reduces to the Navier-Stokes shear viscosity, as expected, i.e., \( P_{yx}(r) = -\eta_0 \dot{\gamma}(r) \).

Previous work [9, 42] suggested that Eq. (6.8) might be used to extract the viscosity from simulations of liquid flow through porous solids in which the degree of confinement meant that the local constitutive relation for viscosity was invalid. It was suggested that if the strain rate and stress profiles could be obtained with sufficient spatial resolution and with high signal to noise, a wave vector dependent viscosity could be calculated from the Fourier transformed data via Eq. (6.11). In this work we demonstrate that there are limitations in the model when it is applied to our confined fluids. In what follows, we will discuss it in details.

We apply numerical Fourier Transforms and inverse Fourier Transforms to all our data from nonequilibrium molecular dynamics simulations, which will be detailed in the next section. The Discrete Fourier Transform is defined as [76]

\[ \tilde{F}(k_y) = T_s \sum_{n=-M}^{M} f(y_n)e^{-2\pi n k i/N} \]  

(6.12)
where \( k_y \equiv 2k\pi/NT_s \), \( y_n \equiv nT_s \), and \( k, n \) are integers in the range \(-M\) to \( M\), where \( M = (N - 1)/2 \) and \( N \) is the number of sampling points, which is odd for all the systems we studied. \( T_s \) is the sampling period. Here \( f(\ldots) \) denotes a function in \( \mathbf{r} \)-space and \( \tilde{F}(\ldots) \) is its Fourier Transform. The inverse Discrete Fourier Transform is given as

\[
f(y_n) = \frac{1}{NT_s} \sum_{k=-M}^{M} \tilde{F}(k_y) e^{2\pi n k i / N} \tag{6.13}
\]

As our integrals in Eq. (6.8) are over all space, the DFT is defined as two-sided, ranging from \(-M\) to \( M\), although we note that the standard DFT definition is one-sided requiring that \( k, n = 0, 1, 2, \ldots, N - 1 \). The one-sided Fourier Transform, denoted by \( \tilde{G} \), and the two-sided transform, \( \tilde{F} \), are however related by a simple symmetry transformation [76]. If \( 0 \leq k_y \leq \pi/T_s \),

\[
\tilde{F}(k_y) = \tilde{G}(k_y) \tag{6.14}
\]

If \(-\pi/T_s \leq k_y \leq 0\),

\[
\tilde{F}(k_y) = \tilde{G} \left( k_y + \frac{2\pi}{T_s} \right) . \tag{6.15}
\]

### 6.4 Discrete Fourier Transforms

First of all, we demonstrate that we perform DFTs correctly by a simple test example. Let \( f(y) = e^{-|y|} \). We use this simple function as it is likely that the viscosity kernel will have a similar form, ie, it has a maximum at \( y = 0 \) and decays to zero at large \( y \). The
Fourier Transform of $f(y)$ is as follows,

$$\tilde{f}(k_y) = \int_{-\infty}^{\infty} e^{-|y|} e^{-ik_y y} dy$$

$$= \int_{-\infty}^{0} e^{-|y|} e^{-ik_y y} dy + \int_{0}^{\infty} e^{-|y|} e^{-ik_y y} dy$$

$$= \int_{-\infty}^{0} e^{y} e^{-ik_y y} dy + \int_{0}^{\infty} e^{-y} e^{-ik_y y} dy$$

$$= \int_{0}^{+\infty} e^{-y} e^{ik_y y} dy + \int_{0}^{+\infty} e^{-y} e^{-ik_y y} dy$$

$$= \int_{0}^{+\infty} e^{-y} [e^{i k_y y} + e^{-i k_y y}] dy$$

$$= \int_{0}^{+\infty} e^{-y} [2 \cos(k_y y)] dy = \frac{2}{1 + k_y^2}$$

(6.16)

In Fig. 6.1 we plot $f(y) = e^{-|y|}$ as a function of $y$.

We take samples of $N$ points ($N = 33, 101$ and $2001$) from $f(y)$ and perform DFTs. In Fig. 6.2 we plot the dependence of the DFTs on the number of sampling points $N$ and show the comparison of the DFTs with the analytic transform given by Eq (6.16). It is clear that the best numerical result can be achieved by a sufficiently large number of sampling points. For such a fast decaying function, our DFTs are in very good agreement with the analytic transform if the number of sampling points $N \gtrsim 101$. As $P_{yx}(y)$ and $\dot{\gamma}(y)$ are actually computed over a limited domain, ie, $y \in (-C, +C)$, in Fig. 6.3 we show the dependence of the DFTs on the truncation $C$, which is defined as the boundary of the domain (otherwise known as support) of $P_{yx}(y)$ and $\dot{\gamma}(y)$, and comparison of the DFTs with the analytic transform. It indicates that the numerical results depend on the truncation size. Our DFTs are in very good agreement with the analytic transform if we choose the truncations approximately where the function decays to zero. In the
6.5 Simulation details

We perform NEMD simulations for a Weeks-Chandler-Andersen (WCA) atomic fluid [50] undergoing planar Poiseuille flow. The algorithm for Poiseuille flow has been

![Figure 6.1: $f(y) = e^{-|y|}$ as a function of $y$.](image)

In the case of our confined fluid systems, we have no control over the truncation size, because the maximum value we can have is $l_y/2 = (L - 1)/2$, i.e., half of the effective width of the pore. As we will see in section 6.7.1, this truncation has important consequences for the extraction of our viscosity kernels. In Table 6.1 we give details on the number of sampling points, sampling period and truncation size used for all the systems that we studied.
Figure 6.2: Dependence of the DFTs on the number of sampling points $N$ and comparison of the DFTs with the analytic transform.

<table>
<thead>
<tr>
<th>Pore width</th>
<th>Sampling points ($N$)</th>
<th>Sampling period ($T_s$)</th>
<th>Truncation size</th>
</tr>
</thead>
<tbody>
<tr>
<td>$L = 5.1$</td>
<td>101</td>
<td>0.041</td>
<td>2.05</td>
</tr>
<tr>
<td>$L = 10.2$</td>
<td>101</td>
<td>0.092</td>
<td>4.60</td>
</tr>
<tr>
<td>$L = 15.3$</td>
<td>101</td>
<td>0.143</td>
<td>7.15</td>
</tr>
</tbody>
</table>

Table 6.1: The number of sampling points, sampling period and truncation size used for all the systems.
The slit-pore model consists of a single wall comprising 3 layers of atoms arranged in an FCC lattice. The atoms in each layer are confined to their equilibrium sites through harmonic restraining forces; the layers themselves are subject to holonomic constraints which act to prevent their centres-of-mass from moving in the transverse direction. This model for pore walls was suggested by Powles et al [77]. Periodic boundary conditions are applied in all 3 Cartesian directions so that only one wall is needed per simulation cell to represent a slit-pore. We perform simulations at three different reduced pore widths: $L/\sigma = 5.1, 10.2$ and $15.3$, where the pore width, $L$, was defined in section 5.1.

Our coordinate system is defined such that the walls are parallel to the $xz$ plane and the fluid is therefore confined in the $y$-direction (shown in Fig. 5.1). To generate Poiseuille flow, we apply a constant force (pressure head) in the positive $x$-direction to all fluid atoms which has the same desired effect as applying a pressure gradient.
without the undesirable complication of density gradients in the flow direction which would result from it. The magnitude of the pressure head used in this work is \( F_e = 0.02 \) for all systems studied. This value is chosen to maximise the signal to noise ratio while maintaining a linear response. Typically, lower values of \( F_e \) may be used for wider pore widths but we found it convenient to retain a single value for all simulations.

In all our simulations, the number of wall atoms is fixed at 216 (72 atoms per layer). The number of fluid atoms differ, depending on the value of the pore width: 216, 504 and 864 for \( L = 5.1, 10.2 \) and 15.3 respectively. For each channel width we simulate fluid densities equal to 0.442 and 0.650, with a constant wall density for all systems of 0.850. We note that there is no unambiguous definition of volume in a pore system [45]. In this work we define the volume to be \( V = (L - 1)L_xL_z \). Parameters for the construction of the simulation cell, including number of atoms, wall and fluid densities, surface density of walls, cell lengths and separation of wall atoms in the \( x \), \( y \) and \( z \) directions are contained in Table 6.2 and 6.3. The geometry of our simulation cell is shown in Fig. 5.1 and Fig. 5.2.

The equations of motion for the wall and fluid atoms are integrated using a 5\(^{th}\) order Gear algorithm with a time step of 0.001. The magnitude of the spring constant for the wall atoms is chosen to be 57.1. This value is chosen to maximise the thermal coupling between the wall and fluid atoms, but we note that this relatively low value permits some penetration of the walls by fluid atoms. A Gaussian thermostat is applied to the wall atoms to maintain a constant kinetic wall temperature of 0.722 throughout the simulations. No thermostats are applied to the fluid; the viscous heat generated is allowed to dissipate via conduction through the walls.

Our simulations are conducted in the following manner. First, a pore of the appropriate dimensions is constructed and then filled with fluid atoms (the wall layers are simply replicated a certain number of times in the pore space). Second, the external field
6.6 Simulation and DFT results

<table>
<thead>
<tr>
<th>Separation of wall atoms in FCC lattice</th>
<th>Wall thickness</th>
<th>Wall density</th>
<th>Number of wall atoms</th>
<th>Surface density of wall layer</th>
</tr>
</thead>
<tbody>
<tr>
<td>∆x</td>
<td>∆z</td>
<td>∆y_w</td>
<td>ρ_w</td>
<td>N_w</td>
</tr>
<tr>
<td>1.818</td>
<td>1.286</td>
<td>2.13</td>
<td>0.85</td>
<td>216</td>
</tr>
<tr>
<td>1.500</td>
<td>1.061</td>
<td>3.13</td>
<td>0.85</td>
<td>216</td>
</tr>
<tr>
<td>1.855</td>
<td>1.312</td>
<td>2.05</td>
<td>0.85</td>
<td>216</td>
</tr>
<tr>
<td>1.530</td>
<td>1.082</td>
<td>3.01</td>
<td>0.85</td>
<td>216</td>
</tr>
<tr>
<td>1.948</td>
<td>1.378</td>
<td>1.86</td>
<td>0.85</td>
<td>216</td>
</tr>
<tr>
<td>1.599</td>
<td>1.131</td>
<td>2.73</td>
<td>0.85</td>
<td>216</td>
</tr>
</tbody>
</table>

Table 6.2: Parameters for setting up the wall atoms in the simulation cell.

is applied and the system allowed to reach a non-equilibrium steady state (typically 1 million time steps). Production runs are then conducted for a duration of 4 million time steps. These extremely long runs are necessary in order to obtain the very high signal to noise values on the binned quantities. The shear stress is computed by integrating the momentum continuity equation, as in Eq. (5.9), and the streaming velocity and density are collected in bins of small width.

6.6 Simulation and DFT results

In Fig. 6.4 the number density $\rho(y)$ is plotted as a function of position $y$ for the three different channel widths. We note that $y = 0$ represents the centre of the fluid channel. As is to be expected the fluid is highly inhomogeneous for the smallest channel
width \((L = 5.1)\) and experiences large density oscillations particularly in the layers immediately adjacent to both walls. These density oscillations are more pronounced for the higher average fluid density system \((\rho = 0.650)\) compared to the lower average density system \((\rho = 0.442)\). They become less significant towards the centre of the channel as the channel size increases. For the \(L = 15.3\) system the only significant inhomogeneity appears in the immediate vicinity of the walls.

The streaming velocity profiles are displayed in Fig. 6.5. For a classical Navier-Stokes fluid undergoing planar Poiseuille flow the streaming velocity profile should be quadratic in \(y\) as demonstrated in Eq. (6.4). This is clearly not the case for the \(L = 5.1\) system, in which pronounced oscillations in the profile exist throughout the fluid, particularly for the higher density system. As the channel width increases these oscillations diminish near the centre of the channel and are only significant close to the interface between

<table>
<thead>
<tr>
<th>Simulation box size</th>
<th>Pore width</th>
<th>Fluid density</th>
<th>Number of fluid atoms</th>
</tr>
</thead>
<tbody>
<tr>
<td>(L_y) (L_x, L_z)</td>
<td>(L)</td>
<td>(\rho_f)</td>
<td>(N_f)</td>
</tr>
<tr>
<td>7.23</td>
<td>10.91</td>
<td>5.1</td>
<td>0.442</td>
</tr>
<tr>
<td>8.23</td>
<td>9.00</td>
<td>5.1</td>
<td>0.650</td>
</tr>
<tr>
<td>12.25</td>
<td>11.13</td>
<td>10.2</td>
<td>0.442</td>
</tr>
<tr>
<td>13.21</td>
<td>9.18</td>
<td>10.2</td>
<td>0.650</td>
</tr>
<tr>
<td>17.16</td>
<td>11.69</td>
<td>15.3</td>
<td>0.442</td>
</tr>
<tr>
<td>18.03</td>
<td>9.60</td>
<td>15.3</td>
<td>0.650</td>
</tr>
</tbody>
</table>

*Table 6.3:* Parameters for setting up the fluid atoms in the simulation cell.
6.6 Simulation and DFT results

Figure 6.4: Number density $\rho(y)$ as a function of $y$ for average fluid densities of $\rho = 0.442$ (dashed line) and 0.650 (solid line). (A) $L = 5.1$, (B) $L = 10.2$, (C) $L = 15.3$.

wall and fluid atoms. The maximum streaming velocity decreases for fixed field strength as the channel width decreases due to increased frictional resistance between wall and
Figure 6.5: Streaming velocity $u_x(y)$ as a function of $y$ for average fluid densities of $ho = 0.442$ (dashed line) and 0.650 (solid line). (A) $L = 5.1$, (B) $L = 10.2$, (C) $L = 15.3$.
Figure 6.6: Strain rate $\dot{\gamma}(y)$ as a function of $y$ for average fluid densities of $\rho = 0.442$ (dashed line) and 0.650 (solid line). (A) $L = 5.1$, (B) $L = 10.2$, (C) $L = 15.3$. 
Figure 6.7: Element of the pressure tensor $P_{yx}(y)$ as a function of $y$ for average fluid densities of $\rho = 0.442$ (dashed line) and 0.650 (solid line). (A) $L = 5.1$, (B) $L = 10.2$, (C) $L = 15.3$. 
Figure 6.8: Imaginary part of k-space transform of shear stress for the system

\[ \rho = 0.442 \text{ and } L = 5.1. \] Filled dots are actual data points while crosses represent approximate zeros.

fluid atoms. Fig. 6.5 does show evidence of slip behaviour, which have been reported by Sokhan, Nicholson and Quirke [78, 79].

The strain rate profiles for the same systems are shown in Fig. 6.6, where deviations from linearity are indicative of departures from Navier-Stokes predictions. As expected, these deviations are particularly pronounced for the \( L = 5.1 \) system, again increasing in extent as the overall system density is increased from 0.442 to 0.650. Linearity is only apparent for the \( L = 10.2 \) and 15.3 systems sufficiently far removed from the walls.

In Fig. 6.7 we plot \( P_{yx}(y) \) as a function of \( y \) for all systems and again observe similar trends to those observed in Figs. 6.4-6.6, noting that deviations from the hydrodynamic linearity expected for Navier-Stokes fluids undergoing Poiseuille flow [45] are manifest as strong oscillations in the vicinity of the walls, and that these oscillations increase in
magnitude with increasing overall system density. In Figs. 6.8, 6.9 we plot the imaginary part of $\tilde{P}_{yx}(k_y)$ and $\tilde{\gamma}(k_y)$ for a pore width of $L = 5.1$. Imaginary parts of both $\tilde{P}_{yx}(k_y)$ and $\tilde{\gamma}(k_y)$ go to zero at some points (represented by crosses), and we note that if all zero points coincide for both, then a limiting behavior will be observed, i.e., at zero points, $k_{yi}, i = 1, 2, \ldots,$

$$\tilde{\eta}(k_{yi}) = -\lim_{k_y \to k_{yi}} \frac{\tilde{P}_{yx}(k_y)}{\tilde{\gamma}(k_y)}.$$ 

In our DFTs, we sample data (filled dots shown in Figs. 6.8, 6.9) such that these zeros are avoided. This results in a smooth distribution of our data. In Figs. 6.10, 6.11 we show that the real parts of $\tilde{P}_{yx}(k_y)$ and $\tilde{\gamma}(k_y)$ are zero. As is to be expected, DFTs of
both $\tilde{P}_{yx}(k_y)$ and $\tilde{\gamma}(k_y)$ must be imaginary and antisymmetric, as the stress and strain rate are antisymmetric.

In Figs. 6.12 and 6.13 we plot Newton’s local viscosity, $\eta_L$, defined by Eq. (6.2), and the Navier-Stokes viscosity, $\eta_{NS}$, for both densities studied. The Navier-Stokes viscosity is computed in the following way. We fit the streaming velocity simulation data to a parabolic function, $u_x(y) = c_2 y^2 + c_0$. Comparison of this function with the Navier-Stokes predicted velocity profile $u_x(y) = -\frac{\rho F_e}{2\eta} (y^2 - h^2)$ leads to the value $c_2 = -\rho F_e / 2\eta_{NS}$, from which we compute the Navier-Stokes viscosity as $\eta_{NS} = -\rho F_e / 2c_2$. It is clear that the local viscosity suffers from singularity problems at $y = 0$ [9, 68, 80]. This is dramatically observed for the high density $L = 5.1$ system, where the local viscosity oscillates significantly throughout the channel and diverges to singular or even negative.
Figure 6.11: Real part of \( k \)-space transform of strain rate as a function of \( y \) for the system \( \rho = 0.442 \) and \( L = 5.1 \).

values near the stationary points in the streaming velocity profile where the strain rate goes to zero (which is, strictly speaking, undefined in the local model). Also shown are the Navier-Stokes viscosities which, by definition, are constant and thus incapable of predicting spatial variations in the viscosity for such highly inhomogeneous fluid systems.

In Fig. 6.14 we plot \( \tilde{\eta}(k_y) \), the Discrete Fourier Transform of \( \eta(y) \) for average fluid densities of \( \rho = 0.442 \) and 0.650. (A) \( L = 5.1 \), (B) \( L = 10.2 \), (C) \( L = 15.3 \). At \( k_y = 0 \), \( \tilde{\eta}(0) = \int_{-L_y}^{L_y} \eta(y) dy \), which is an integral over \( \eta(y) \) with respect to the separation \( y \), and represents the effective viscosity for the system. We note that \( \tilde{\eta}(0) \) is computed as \( -\tilde{P}_{yx}(0)/\tilde{\gamma}(0) \), which involves the ratio of two quantities that are identically zero. Remarkably however we find that the finite precision numerics gives us physically sensible values of this ratio, which can be interpreted as the infinite wavelength viscosity, or an
Figure 6.12: Shear viscosities as a function of $y$ for an average fluid density of $\rho = 0.442$. (A) $L = 5.1$, (B) $L = 10.2$, (C) $L = 15.3$. $\eta_L(y)$ is the local viscosity computed as $\langle -P_{yx}(y)/\dot{\gamma}(y) \rangle$, where $\langle \cdots \rangle$ denotes a time average; $\eta_{NS}$ is the Navier-Stokes shear viscosity.
Figure 6.13: Shear viscosities as a function of \( y \) for an average fluid density of \( \rho = 0.650 \). (A) \( L = 5.1 \), (B) \( L = 10.2 \), (C) \( L = 15.3 \). Viscosities are as defined in Fig. 6.12.

effective pore viscosity. Even though this is fortuitous we find that this value agrees well with the Navier-Stokes viscosity which is obtained by an independent method. The results of this comparison are shown in Table 6.4. As already mentioned, apart from the
Figure 6.14: \( \tilde{\eta}(k_y) \), DFTs of \( \eta(y) \) for average fluid densities of \( \rho = 0.442 \) and 0.650.

(A) \( L = 5.1 \), (B) \( L = 10.2 \), (C) \( L = 15.3 \).
Figure 6.15: $\eta(y)$, a function of compact support, computed via Eq. (6.8) for average fluid densities of $\rho = 0.442$ and $0.650$. (A) $L = 5.1$, (B) $L = 10.2$, (C) $L = 15.3$. 


6.6 Simulation and DFT results

Figure 6.16: $P_{yx}(y)$ computed directly by the IMC method [45] (solid line) compared with the reconstructed value computed by the constitutive model defined by Eq. (6.8) (dashed line) for the system $L = 5.1$, $\rho = 0.442$.

$k_y = 0$ value of $\tilde{\eta}(k_y)$, we avoid problems of division by zero by judiciously sampling non-zero values of the strain rate transform data. Fig. 6.14(B), (C) shows that the effective viscosity ($\tilde{\eta}(k_y = 0)$) increases as the average fluid density goes up. In Fig. 6.14(A), $\tilde{\eta}(k_y)$ is not shown for $\rho = 0.650$ because the resulting kernel is unphysical indicating that the model used for extracting the viscosity kernel breaks down for high density ($\rho = 0.650$) and small pore width ($L = 5.1$). We notice that $\tilde{\eta}(k_y)$ does not decay to zero as would be expected [75], but sits on a constant background as $k_y$ approaches infinity. However, we find $\tilde{\eta}(k_y)$ does approach zero, as $k_y \to \infty$ if the missing part of shear stress with support $|y| > l_y/2$ is known, shown in Fig. 6.22(a). We will comment on this in more
detail in section 6.7. Note also that the magnitude of this constant background decreases as the density of the fluid decreases.

From the inverse transform we extract the nonlocal viscosity kernel, $\eta(y)$, which is called a function of compact support. It is a function of a spatial coordinate. That spatial coordinate only has physical meaning inside the convolution. In this case $y$ must be interpreted as a separation distance. It is the separation between where we want to know the stress and where the strain rate is that makes the contribution to the convolution. Thus $y = 0$ means zero separation between these two points. In Fig. 6.15 we plot $\eta(y)$ as a function of relative coordinate $y$, computed via Eq. (6.8) for average fluid densities of $\rho = 0.442$ and 0.650. (A) $L = 5.1$, (B) $L = 10.2$, (C) $L = 15.3$.

Fig. 6.15 indicates that $\eta(y)$ has a peak at $y = 0$, and get smaller and decays to zero as $y$ increases. Physically, it means that the strain rate at the location where we want to know the stress contributes most to the stress, and the contribution of the strain rate becomes less significant as the relative distance $y$ increases.

We note that the value at $y = 0$ appears to be an unphysical spike as one can not smoothly extrapolate this value from the other $\eta(y)$ values. It is easy to see why this spike arises. Note that our $\tilde{\eta}(k_y)$ data all sit on a constant background as $k_y \to \infty$. If we write this as

$$\tilde{\eta}(k_y) \approx \tilde{\eta}_1(k_y) + \tilde{\eta}_0 \quad (6.17)$$

where $\tilde{\eta}_0$ is a constant, then the inverse transform is clearly

$$\eta(y) \approx \eta_1(y) + \tilde{\eta}_0 \delta(y) \quad (6.18)$$

Thus a constant background in $k$-space results in a delta function, which appears as an unphysical spike for $\eta(y = 0)$ when we compute the inverse transform by the DFT method. As it is a mathemtical requirement that $\tilde{\eta}(k_y = 0) = \int_{-\infty}^{\infty} \eta(y)dy$, if a spike exists at $y = 0$ the inverse DFT will necessarily suppress all subsequent values of $\eta(y)$ such
that the overall integral of $\eta(y)$ is equal to $\tilde{\eta}(k_y = 0)$. This explains why, apart from the spike at $\eta(y = 0)$, the values of $\eta(y)$ for the denser fluids displayed in Fig. 6.15(B), (C) are actually lower than those for the less dense fluid. This is physically unsound (the viscosity of a dense fluid is larger than that of a less dense fluid of the same substance at the same thermodynamic state point) and is a consequence only of mathematical necessity.

Table 6.4 shows the Navier-Stokes viscosity, $\eta_{NS}$, and the effective pore viscosity, $\tilde{\eta}(k_y = 0)$. For the systems of $L = 10.2$ and 15.3, the Navier-Stokes viscosity and the effective pore viscosity agree well, but for the system of $L = 5.1$ and $\rho = 0.65$, the model seems to break down completely and the kernel behaves unphysically, so we do not present the viscosity kernel for this system.

In order to confirm that the viscosity kernel defined by Eq. (6.8) is consistent with that definition, we compute the stress from Eq. (6.8), using as input the viscosity kernel computed from Eqs. (6.11), (6.12), (6.13) and the strain rate data. Excellent agreement is found between both stress values, shown in Fig. 6.16, confirming the validity of our numerical procedure.

<table>
<thead>
<tr>
<th></th>
<th>$L = 5.1$</th>
<th>$L = 10.5$</th>
<th>$L = 15.3$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\eta_{NS}$, $\tilde{\eta}(0)$</td>
<td>$\rho = 0.442$</td>
<td>$\rho = 0.650$</td>
<td>$\rho = 0.442$</td>
</tr>
<tr>
<td>$\eta_{NS}$</td>
<td>0.410</td>
<td>1.240</td>
<td>0.380</td>
</tr>
<tr>
<td>$\tilde{\eta}(0)$</td>
<td>0.406</td>
<td>x</td>
<td>0.364</td>
</tr>
</tbody>
</table>

Table 6.4: Comparison of Navier-Stokes viscosity, $\eta_{NS}$, and $\tilde{\eta}(k_y = 0) = \int_{-l_y}^{l_y} \eta(y)dy$. 
6.7 Effect of domain restriction on inverse convolution

Our DFTs have a problem in that \( \tilde{\eta}(k_y) \) does not drop to zero as \( k_y \) approaches infinity as one might expect \[75\]. In what follows we perform an analysis of the simple homogeneous constitutive equation used in this work and illustrate the limitations of the model by using simple test functions. Our analysis is made using numerical DFTs and exact analytic FTs.

6.7.1 The physical system

In Eq. (6.19), both \( P_{yx}(y) \) and \( \dot{\gamma}(y) \) are actually computed over a limited domain \( y \in (-a, a) \) from our simulations, even though the integral extends over all space.

\[
P_{yx}(y) = -\int_{-\infty}^{\infty} dy' \eta(y - y') \dot{\gamma}(y') \quad (6.19)
\]

It is obvious that Eq. (6.19) could only strictly apply to a shift invariant system, but for our system under investigation this is not the case as we have explained in section 6.2. We assume that \( \eta(y) \) has compact support, in other words, \( \eta(y) \) is non-zero only on a finite interval, for example, \( \eta(y) = 0, \text{ if } |y| > \xi \). If \( \xi \ll a \), we might be able to derive an effective convolution kernel \( \eta(y) \) from Eq. (6.19).

6.7.2 Test functions:

In Eq. (6.19), we will model \( P_{yx}(y) \) and \( \dot{\gamma}(y) \) by linear ramp-like test functions,

\[
f(y) = A\Pi^{(a)}(y) y \quad (6.20)
\]

where \( A \) is a constant and the top-hat function \( \Pi^{(a)}(y) \) is defined by

\[
\Pi^{(a)}(y) = \begin{cases} 
1 & |y| \leq a \\
0 & |y| > a 
\end{cases} \quad (6.21)
\]

If the kernel \( \eta(y) \) has an effective length scale, \( \xi \) (i.e. \( \eta(y) = 0, \text{ if } |y| > \xi \)), which is
6.7 Effect of domain restriction on inverse convolution

Figure 6.17: A moving window of $\eta(y)$. If the support of $\eta(y)$ is $\xi = 0.2$, at $y = 0.4$, $P_{yx}(y)$ is dependent on $\dot{\gamma}(y)$ only in the domain of $y \in (0.2, 0.6)$, as $\eta(y) = 0$ if $|y - 0.4| > 0.2$.

considerably less than the length of the support (ie, the effective domain) of $\dot{\gamma}(y)$, then outside of the range $(y - \xi, y + \xi)$, $P_{yx}(y)$ is more or less independent of the strain rate. For example, if we choose the support of $\eta(y)$ is $\xi = 0.2$, then at $y = 0.4$, $P_{yx}(y)$ is dependent on $\dot{\gamma}(y)$ only in the domain of $y \in (0.2, 0.6)$, as $\eta(y) = 0$ if $|y - 0.4| > 0.2$ shown in Fig. 6.17.

6.7.3 Restriction of the domain of $P_{yx}(y)$:

The problem arises because the function $P_{yx}(y)$ has the same domain as that of $\dot{\gamma}(y)$. This is not the natural support of a function described in Eq. (6.19). We assume that $\eta(y-y')$ in Eq. (6.19) has an effective length scale, $\xi$, ie, $\eta(y-y') \neq 0$, if $-\xi < y-y' < \xi$, 
and that \( \dot{\gamma}(y') \) in Eq. (6.19) has the domain of \(-a < y' < a\). It is obvious that \( P_{yx}(y) \) in Eq. (6.19) has the domain of \(- (\xi + a) < y < (\xi + a)\). \( P_{yx}(y) \) has a larger domain than \( \dot{\gamma}(y) \). In other words, the support of \( P_{yx}(y) \), which has a length equal to the sum of the lengths of the supports of \( \dot{\gamma}(y) \) and \( \eta(y) \), extends beyond the interval \((-a, a)\) to \((-\xi - a, \xi + a)\). Note that in our NEMD simulations we do not have this information, i.e., we do not know what the natural stress response of the fluid would be in this extended region. In Fig. 6.18 we show that the information in the section \( a < |y| < a + \xi \) is lost in the current approach and consequently the estimated kernel is inaccurate. This is relatively easy to believe since essentially half of the relevant region affected by the convolution has been lost and a simple inversion based on the assumption that all the data is available will be inaccurate. More details will be discussed in section 6.7.5.
6.7 Effect of domain restriction on inverse convolution

6.7.4 Type of error expected:

Assuming that the kernel \( \eta(y) \) is non-zero only on the interval \( |y| \leq \xi \), then the true stress function \( P_{yx}(y) \) can be written as the sum of the available estimate, \( P_{yx}^{(a)}(y) \) with support \( |y| \leq a \), and the missing part \( P_{yx}^{(e)}(y) \) with support \( a < |y| \leq a + \xi \).

\[
P_{yx}(y) = P_{yx}^{(a)}(y) + P_{yx}^{(e)}(y) \Rightarrow P_{yx}^{(a)}(y) = P_{yx}(y) - P_{yx}^{(e)}(y) \tag{6.22}
\]

If \( \dot{\gamma}(y) \) is odd in \( y \), and \( \eta(y) \) is even in \( y \), then \( P_{yx}^{(e)}(y) \) is of the form

\[
P_{yx}^{(e)}(y) = b(y - a) - b(-a - y) \tag{6.23}
\]

where \( b(y) = 0 \) for \( y < 0 \). In other words,

\[
P_{yx}^{(e)}(y) = \begin{cases} 
  b(y - a) & y > a \\
  0 & -a \leq y \leq a \\
  -b(-a - y) & y < -a 
\end{cases} \tag{6.24}
\]

Given that \( b(y) \) is real, the Fourier Transform of the error term can be written as

\[
\tilde{P}_{yx}^{(e)}(k_y) = -2i|\tilde{B}(k_y)|\sin(k_ya + \phi(k_y)) \tag{6.25}
\]

where \( \tilde{B}(k_y) \) is the Fourier Transform of \( b(y) \) and \( \phi(k_y) \) is the argument of the complex number \( \tilde{B}(k_y) \). The estimate of \( \tilde{\eta}(k_y) \) is therefore as follows

\[
\tilde{\eta}^{(a)}(k_y) = -\frac{\tilde{P}_{yx}^{(a)}(k_y)}{\tilde{\dot{\gamma}}(k_y)} = -\left[ \frac{\tilde{P}_{yx}(k_y) - \tilde{P}_{yx}^{(e)}(k_y)}{\tilde{\dot{\gamma}}(k_y)} \right] = \tilde{\eta}(k_y) - \frac{2i|\tilde{B}(k_y)|\sin(k_ya + \phi(k_y))}{\tilde{\dot{\gamma}}(k_y)} \tag{6.26}
\]

where we note that \( \tilde{\eta}(k_y) \) is the true kernel in \( k \)-space. The last term on the right in Eq. (6.26) is the error due to the restriction of the domain of \( P_{yx}(y) \).
denominator has zeros which are not exactly cancelled by zeros in the numerator, the error term diverges. For example, if

\[ \dot{\gamma}(y) = -\alpha \Pi^{(a)}(y)y, \]  

(6.27)

its Fourier Transform is

\[ \hat{\dot{\gamma}}(k_y) = -2i\alpha \left( \frac{k_y a \cos(k_y a) - \sin(k_y a)}{k_y^2} \right). \]  

(6.28)

\[ \dot{\gamma}(k_y) \] has zeros at the points where \( \tan(k_y a) = k_y a. \) If these zeros are not cancelled by corresponding zeros in the numerator of the error term, divergences in \( \tilde{\eta}^{(a)}(k_y) \) will take place at these zero points.

6.7.5 Explicit example:

In what follows we use examples to demonstrate what we have derived above. Suppose that \( \dot{\gamma}(y) \) is strictly a ramp given by Eq. (6.27), and a kernel \( \eta(y) \) takes the form of

\[ \eta(y) = \frac{\eta_0}{2\xi} e^{-\frac{|y|}{\xi}}. \]  

(6.29)

The Fourier Transform of \( \eta(y) \) is

\[ \hat{\eta}(k_y) = \frac{\eta_0}{1 + (k_y \xi)^2}. \]  

(6.30)

From Eqs. (6.28) and (6.30), we obtain

\[ \hat{P}_{yx}(k_y) = -\hat{\eta}(k_y)\dot{\gamma}(k_y) = -i \frac{2\alpha \eta_0}{1 + (k_y \xi)^2} \left( \frac{k_y a \cos(k_y a) - \sin(k_y a)}{k_y^2} \right). \]  

(6.31)

\( P_{yx}(y) \) is given as follows by performing the inverse Fourier Transform on \( \hat{P}_{yx}(k_y) \), or alternatively by taking the convolution of \( \eta(y) \) with \( \dot{\gamma}(y) \) via Eq. (6.19) over the full
support of $P_{yx}(y)$,

$$
P_{yx}(y) = \frac{\alpha \eta_0 \xi}{2} \left\{ \begin{array}{ll} 
\left[ \left( 1 - \frac{a}{\xi} \right) e^{\frac{(y+a)}{\xi}} - \left( 1 + \frac{a}{\xi} \right) e^{\frac{(y-a)}{\xi}} \right] ; & y < -a \\
2 \frac{y}{\xi} + \left( 1 + \frac{a}{\xi} \right) \left( e^{\frac{-\alpha}{\xi}} \right) - e^{\frac{(y-a)}{\xi}} ; & -a \leq y \leq a \\
- \left[ \left( 1 - \frac{a}{\xi} \right) e^{\frac{-\alpha}{\xi}} - \left( 1 + \frac{a}{\xi} \right) e^{\frac{(y+a)}{\xi}} \right] ; & y > a
\end{array} \right. \hspace{1cm}(6.32)$$

Eq. (6.32) shows that the support of $P_{yx}(y)$ given by Eq. (6.19) extends beyond the support of $\gamma(y)$. Fig. 6.19 shows that restriction of the domain of $P_{yx}(y)$ to $(-a, a)$ will lead to a serious loss of information and so the estimate of $\eta(y)$ will be inaccurate for highly confined fluids. This is because we have no direct knowledge of the stress response function $P_{yx}(y)$ in the range $|y| > a$, which is required to extract the true kernel via Eq. (6.19). The information about the kernel’s shape is contained predominantly in the behaviour of $P_{yx}(y)$ in the vicinity of the boundaries. The linear form of $P_{yx}(y)$ with a slope that depends on the integral of the kernel function but is independent of the exact shape is apparent far removed from the boundaries.

If $P_{yx}(y)$ is truncated at $y = \pm a$, then regardless of the value of $\xi$, the truncation has a significant effect on the estimate of $\eta(y)$. Actually the difference between the kernel found using the full $P_{yx}(y)$ function and that obtained using the truncated function could be dramatic, as we will now show.

In this example, we are able to analytically calculate the Fourier Transforms of all the relevant quantities. For the truncated $P_{yx}$, the function $b(y)$ can be derived from Eqs. (6.23) and (6.32). When $y > a$,

$$
b(y-a) = P_{yx}^{(c)}(y) = P_{yx}(y) - P_{yx}^{(a)}(y) = -\frac{\alpha \eta_0 \xi}{2} \left[ \left( 1 - \frac{a}{\xi} \right) e^{\frac{-\alpha}{\xi}} - \left( 1 + \frac{a}{\xi} \right) e^{\frac{(y+a)}{\xi}} \right]
$$
Figure 6.19: Strain rate given by Eq. (6.27) and shear stress computed using Eq. (6.32). We choose $a = 0.5$ and $\xi = 0.2$ for illustrative purposes.

Let $t = y - a$, we obtain

$$b(t) = \frac{\alpha \eta_0 \xi}{2} \left[ \left( \frac{a}{\xi} - 1 \right) + \left( 1 + \frac{a}{\xi} \right) e^{-\frac{2a}{\xi}} \right] e^{-\frac{t}{\xi}} \quad t > 0$$

(6.33)

The Fourier Transform of Eq. (6.33) is

$$\tilde{B}(k_y) = \frac{\alpha \eta_0 \xi}{2} \left[ \left( \frac{a}{\xi} - 1 \right) + \left( 1 + \frac{a}{\xi} \right) e^{-\frac{2a}{\xi}} \right] \frac{\xi}{1 + ik_y \xi}$$

(6.34)

Substituting Eqs. (6.28), (6.30) and (6.34) into Eq. (6.26), we obtain the Fourier Transform of the estimated kernel, $\tilde{\eta}^{(a)}(k_y)$

$$\tilde{\eta}^{(a)}(k_y) = \frac{\eta_0}{1 + (k_y \xi)^2} \left[ 1 + \left| \left( \frac{a}{\xi} - 1 \right) + \left( \frac{a}{\xi} + 1 \right) e^{-\frac{2a}{\xi}} \right| \frac{\xi^2 k_y^2 (\sin(k_y a - \arctan(k_y \xi)))}{2(k_y \cos(k_y a) - \sin(k_y a))} \right]$$

(6.35)
Equation (6.35) shows that the Fourier Transform of the estimated kernel is significantly different from the transform of the true kernel (Eq. (6.30)). For any typical $\xi > 0$ the second term diverges at a sequence of values of $k_ya$. The lowest value for which this occurs is at about $k_ya \approx 4.5$. Fig. 6.20 shows the comparison of the transform of the true kernel $\tilde{\eta}(k_y)$ (Eq. (6.30)) with the transform of the estimated kernel $\tilde{\eta}^{(a)}(k_y)$ (Eq. (6.35)) based on restricted domain data which clearly shows this divergence. As for Fig. 6.20, Fig. 6.21 is with smaller effective length scale of the kernel, $\xi = 0.01$.

### 6.7.6 The problem with the DFT method:

In Fig. 6.22(a) and (b) we show the comparison of the true kernel transform with the
Figure 6.21: As for Fig. 6.20 but with smaller effective length scale of the kernel. $a = 0.5$ and $\xi = 0.01$ (Note change in scale).

estimated kernel transform and the comparison of the true kernel with the estimated kernel by the DFT method. $\dot{\gamma}(y)$ is padded to $\sqrt{5}$ times the length of its support, ie, the domain of $\dot{\gamma}(y)$ is extended beyond the interval $(-a, a)$ to $(-\sqrt{5}a, \sqrt{5}a)$, and we let $\dot{\gamma}(y) = 0$ in locations beyond its true support, $a < |y| < \sqrt{5}a$. Exact values of $P_{yx}(y)$ (defined in Eq. (6.32) and shown in Fig. 6.19) are used with the support of $(-\sqrt{5}a, \sqrt{5}a)$. To compute $\tilde{\eta}^{(a)}(k_y)$ we take the DFT of $P_{yx}(y)$ and the DFT of $\dot{\gamma}(y)$ defined by Eq. (6.27) and take their negative ratio (ie, $\tilde{\eta}^{(a)}(k_y) = -\tilde{P}_{yx}(k_y)/\tilde{\dot{\gamma}}(k_y)$). $\eta^{(a)}(y)$ is then computed by taking the inverse DFT of $\tilde{\eta}^{(a)}(k_y)$. We choose $\eta_0 = 1$, $a = 0.5$ and $\xi = 0.1$. It is
6.7 Effect of domain restriction on inverse convolution

Figure 6.22: (a) Comparison of the true kernel transform \( \tilde{\eta}(k_y) \) (dashed line, via Eq. (6.30)) with the estimated kernel transform \( \tilde{\eta}^a(k_y) = -\tilde{P}_{yx}(k_y)/\tilde{\gamma}(k_y) \) by the DFT method (filled dots). Exact values and the true support of \( P_{yx}(y) \) with padding to \( \sqrt{5} \) times the length of the support of \( \tilde{\gamma}(y) \) are used. (b) Comparison of the true kernel \( \eta(y) \) (dashed line, via Eq. (6.29)) with the estimated kernel \( \eta^{(a)}(y) \) (filled dots) by inverse DFT of \( \tilde{\eta}^a(k_y) \) shown in (a). \( \eta_0 = 1, \ a = 0.5 \) and \( \xi = 0.1 \).
Figure 6.23: As for Fig. 6.22, but the data were sampled only over the support of \( \dot{\gamma}(y) \) and processed without padding. Note the elevated floor in the estimated kernel transform. The value of \( \tilde{\eta}(k_y = 0) \) is taken as the average of the two adjacent points. We do not plot the spike at \( \eta^{(a)}(y = 0) \) for display purposes.
shown that if we take the true value and full support of $P_{yx}$ into account and pad the values of $\dot{\gamma}(y)$ then the estimated kernel and its transform are in good agreement with our analytic Fourier Transform results. We note that the estimated kernel transform drops to zero as $k_y$ increases as is required. On the other hand, if we now restrict the support of $P_{yx}(y)$ to $(-a, a)$ (ie, restricted to the support of $\dot{\gamma}(y)$), then we find that our estimated kernel is inaccurate due to the loss of information in $P_{yx}(y)$ at the boundaries. This is shown in Fig. 6.23. We notice that the estimated kernel transform shown in Fig. 6.23(a) does not drop to zero as $k_y$ approaches infinity. This is precisely what we have found in our real simulation systems in Fig. 6.14. We note that the elevated floor in the estimated kernel transform results in a distorted shape of $\eta^{(a)}(y)$ and a delta function spike at $\eta^{(a)}(y = 0)$ which significantly underestimates the viscosity kernel at all other locations. The precise degree of influence is likely to depend on the elevated floor in the estimated kernel transform due to the missing part of the shear stress profile as well as by the circular (ie, periodic) convolution which is inherent in the DFT approach (this will be addressed in a separate paper but does not form part of this thesis). This indicates the limitation of the model when it is applied to confined systems because of the lack of complete information of the true stress profile due to the fact that our FTs are all truncated because of the nature of the physical systems studied, and also to the circular convolution inherent in the DFT approach.

6.8 Analytic Fourier Transform

In this section we compute the estimated kernel $\tilde{\eta}^{(a)}(k_y)$ for our real system ($\rho = 0.650, L = 10.2$) analytically. We do this by fitting the shear stress and strain rate data with a simple series and then take the Fourier Transforms using Mathematica. Our
Figure 6.24: The comparison of a series fit to our simulation data for $P_{yx}(y)$ for the system $\rho = 0.650$ and $L = 10.2$.

Simulation data for $P_{yx}(y)$ and $\dot{\gamma}(y)$ can be fit to a simple series as follows [9],

$$P_{yx}(y) = a_0 y + \frac{l_y}{2\pi} \sum_{n=1}^{14} a_n \sin \left( \frac{2\pi n}{l_y} y \right)$$  \hspace{1cm} (6.36)

$$\dot{\gamma}(y) = b_0 y - \frac{2\pi}{l_y} \sum_{n=1}^{14} b_n \sin \left( \frac{2\pi n}{l_y} y \right)$$  \hspace{1cm} (6.37)

where $l_y$ is the effective pore width ($l_y = L - 1.0$), and $a_n$ and $b_n$ are given in Table 6.5 for the system ($\rho = 0.650, L = 10.2$). The fits to our simulation data for $P_{yx}(y)$ and $\dot{\gamma}(y)$ are shown in Fig. 6.24 and Fig. 6.25 respectively.

In Fig. 6.26, 6.27 we show that the imaginary parts of the analytic $k$-space transforms of $\tilde{P}_{yx}(k_y)$ and $\tilde{\dot{\gamma}}(k_y)$ are antisymmetric as expected. The real parts of the both are zero, which is again what we expect as the stress and strain rate are antisymmetric.
### Table 6.5: Coefficients of the series fits for $P_{yx}(y)$ and $\dot{\gamma}(y)$.

<table>
<thead>
<tr>
<th>n</th>
<th>$a_n$</th>
<th>$b_n$</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0.01282</td>
<td>-0.02536</td>
</tr>
<tr>
<td>1</td>
<td>0.00014</td>
<td>-0.03354</td>
</tr>
<tr>
<td>2</td>
<td>-0.00008</td>
<td>0.01525</td>
</tr>
<tr>
<td>3</td>
<td>0.00007</td>
<td>-0.01001</td>
</tr>
<tr>
<td>4</td>
<td>-0.00007</td>
<td>0.00647</td>
</tr>
<tr>
<td>5</td>
<td>0.00008</td>
<td>0.00468</td>
</tr>
<tr>
<td>6</td>
<td>-0.00010</td>
<td>0.00339</td>
</tr>
<tr>
<td>7</td>
<td>0.00015</td>
<td>-0.00047</td>
</tr>
<tr>
<td>8</td>
<td>-0.00024</td>
<td>-0.00137</td>
</tr>
<tr>
<td>9</td>
<td>0.00031</td>
<td>0.00455</td>
</tr>
<tr>
<td>10</td>
<td>-0.00023</td>
<td>-0.00517</td>
</tr>
<tr>
<td>11</td>
<td>0.00013</td>
<td>0.00153</td>
</tr>
<tr>
<td>12</td>
<td>-0.00007</td>
<td>0.00031</td>
</tr>
<tr>
<td>13</td>
<td>0.00004</td>
<td>0.00051</td>
</tr>
<tr>
<td>14</td>
<td>-0.00002</td>
<td>0.00091</td>
</tr>
</tbody>
</table>
Fig. 6.25: The comparison of a series fit to our simulation data for $\dot{\gamma}(y)$ for the

system $\rho = 0.650$ and $L = 10.2$.

Fig. 6.28 shows $\tilde{\eta}^{(a)}(k_y)$ computed as $\tilde{\eta}^{(a)}(k_y) \equiv -\tilde{P}_{yx}(k_y)/\tilde{\gamma}(k_y)$, for the system $\rho = 0.650$ and $L = 10.2$. Our analytic Fourier Transform displays singularities in $\tilde{\eta}^{(a)}(k_y)$, when $k_y \approx \pm 0.98, \pm 1.68, \pm 2.37, \pm 3.06, \pm 3.74, \pm 4.43$, etc, shown in Fig. 6.28. The occurrence of these singularities at regular intervals of $\sim 0.69$ has been predicted and analysed for a simple linear strain rate profile in Eq. (6.35) in section 6.7.5. The use of the DFT method allows us to sample points judiciously so as to avoid these singularities as explained in section 6.6. Clearly we can not obtain sensible $\eta(y)$ values analytically and so the DFT procedure is probably better suited for confined fluids despite its own inherent limitations described in section 6.7.
Figure 6.26: Imaginary part of $\tilde{P}_{yx}(k_y)$ for the system $\rho = 0.650$ and $L = 10.2$. 
Figure 6.27: Imaginary part of $\tilde{\gamma}(k_y)$ for the system $\rho = 0.650$ and $L = 10.2$. 
Figure 6.28: \( \tilde{\eta}(k_y) \equiv -\tilde{P}_{yz}(k_y)/\tilde{\gamma}(k_y) \) for the system \( \rho = 0.650 \) and \( L = 10.2 \).
In this chapter, the rheological and structural properties of polymeric fluids confined in slit pores undergoing Poiseuille flow are investigated. In the first section, the molecular model for the simulations is explained. The second section presents the simulation results.

### 7.1 Computational methodology

The systems studied in this chapter are confined polymeric fluids in slit pores (shown in Fig. 7.1 and Fig. 7.2). The coordinate system is chosen so that the wall surfaces lie in the $x-z$ plane, and the normal of the surface is in the $y$ direction. Our geometry is such that $y = 0$ defines the center of the fluid channel. The simulation cell is as described in section 5.1. The walls are kept at a constant temperature of 1.0 and density 0.85 and are thermostatted as in section 5.1.

The chain molecules between the walls are described by a bead-spring model. Previous studies have shown that this model yields realistic dynamics for polymer melts [81]. Work has also been done to show how to map between it and detailed chemical models of polymers [82, 83]. Each molecule contains $n$ spherical atoms of mass $m$. We consider chains with $N_S = 2, 4, 10$ and 20 “beads” per chain. All atoms interact through the WCA potential [50]:

$$U_{ij}^L = 4\varepsilon\left[(\sigma/r_{ij})^{12} - (\sigma/r_{ij})^6\right] + \varepsilon \quad \text{for} \quad r < r_c$$  \hspace{1cm} (7.1)
Figure 7.1: Geometry of the overall simulation system including molecular fluid and wall atoms (z-direction normal to page).

Figure 7.2: Snapshot of the simulation. Black particles represent the walls and the others confined between the walls are fluid molecules. $z$ axis is normal to the page.
where \( r_{ij} \) is the separation between the sites represented by atoms \( i \) and \( j \), \( r_c \) is the truncation distance, \( 2^{1/6} \sigma \), \( \varepsilon \) is the potential well depth and \( \sigma \) is the effective diameter of the atoms. The potential is zero for \( r > r_c \). In our simulations the wall atoms and fluid atoms have the same Lenard-Jones parameters and the same mass. We give all quantities in terms of Lennard-Jones reduced units defined in section 5.1.

Adjacent atoms along a chain are coupled by an additional potential that prevents chains from crossing or breaking :

\[
U^{FENE}_{ij} = -\left(1/2\right) k R_0^2 \ln[1 - (r_{ij}/R_0)^2] \quad \text{for} \quad r_{ij} \leq R_0 ,
\]

where \( U^{FENE}_{ij} \) is an attractive finitely extensible nonlinear elastic (FENE) potential [81], \( R_0 \) is a finite extensibility, and \( k \) is a spring constant. We set \( R_0 = 1.5\sigma \) and \( k = 30\varepsilon/\sigma^2 \). The choice of parameters for the FENE potential has been demonstrated to prevent bond crossing at temperatures even higher than the one used in our simulation [81]. In Fig. 7.3 we show the comparison of WCA, FENE and WCA+FENE potentials. In our simulations we set \( \sigma = \varepsilon = 1 \).

The equations of motion for the wall atoms are given as Eqs. (5.1) and (5.2). The fluid atoms obey Newton’s equations of motion

\[
\dot{\mathbf{r}}_{i\alpha} = \frac{\mathbf{p}_{i\alpha}}{m} ,
\]

\[
\dot{\mathbf{p}}_{i\alpha} = \mathbf{F}^{LJ}_{i\alpha} + \mathbf{F}^{FENE}_{i\alpha} + iF_e ,
\]

where \( \mathbf{F}^{LJ}_{i\alpha} \) is the Lennard-Jones force exerted on atom \( \alpha \) on molecule \( i \) due to all the other fluid atoms and wall atoms, \( \mathbf{F}^{FENE}_{i\alpha} \) refers the FENE force exerted on atom \( \alpha \) on molecule \( i \) due to intramolecular interactions governed by the additional FENE potential. \( F_e \) is the external driving field and \( i \) is the unit vector in the \( x \)-direction and we again note that \( \mathbf{r}_{i\alpha} \) and \( \mathbf{p}_{i\alpha} \) refer to the laboratory position and momentum of atom \( \alpha \) on
molecule $i$ respectively. In our simulations the field strengths used are $F_e = 0.012$ and 0.5.

The equations of motion are solved with a fifth order Gear predictor-corrector scheme with an integration time step of $\tau = 0.001$. Our simulations are first run for a total of $10^6$ time steps to reach a nonequilibrium steady-state. Once steady-state is achieved, typical production runs of a total of $10^6$ time steps are run with averages accumulated in blocks of 10000 time steps.

7.2 Simulation results

In order to validate our simulation results, we firstly simulate a fluid composed of diatomic molecules undergoing planar Poiseuille flow between two atomistic walls. The simulation conditions are similar to those used by Travis and Evans [84]. Our results qualitatively agree with their work.
Figure 7.4: Fluid molecular number density $\rho_m(y)$ multiplied by the number of atoms $N_S$ per chain for an average fluid density of $\rho = 0.544$, force field $F_e = 0.25$, molecular length of $N_S = 2$ and pore width $L = 10.2$.

The atomic number density $\rho_a(\mathbf{r}, t)$ and molecular number density $\rho_m(\mathbf{r}, t)$ at position $\mathbf{r}$ and time $t$ are defined as

$$
\sum_i m \delta(\mathbf{r} - \mathbf{r}_i(t)) \equiv m \rho_a(\mathbf{r}, t), \quad (7.5)
$$

$$
\sum_j M \delta(\mathbf{r} - \mathbf{r}_{jc}(t)) \equiv M \rho_m(\mathbf{r}, t). \quad (7.6)
$$

In Eq. (7.5), the index $i$ labels the individual fluid atoms and $\mathbf{r}_i(t)$ is the position vector of fluid atom $i$ at time $t$. Fluid atoms are assumed here to have identical masses, namely, $m$. In Eq. (7.6), $M$ is the mass of a fluid molecule ($M = N_S \cdot m$, $N_S$ is the number of atoms on each molecule), the index $j$ labels the fluid molecules and $\mathbf{r}_{jc}(t)$ is the position vector of the center of mass of molecule $j$ at time $t$. 
7.2 Simulation results

Similarly, the atomic streaming velocity $u_a$ and molecular streaming velocity $u_m$ are defined as follows,

$$
    u_a(r, t) = \frac{\sum_i m v_i(t) \delta(r - r_i(t))}{\sum_i m \delta(r - r_i(t))},
$$

(7.7)

$$
    u_m(r, t) = \frac{\sum_j M v_{jc}(t) \delta(r - r_{jc}(t))}{\sum_j M \delta(r - r_{jc}(t))}.
$$

(7.8)

In Eq. (7.7), $v_i(t)$ is the velocity of fluid atom $i$ at time $t$. In Eq. (7.8), $v_{jc}(t)$ is the velocity of fluid molecule $j$ at time $t$. For a system such as ours we divide the simulation cell into a number of slabs of thickness $\Delta_y$ and compute the number density and streaming velocity as an average evaluated at the midpoint of each slab. This is the simple histogram method. In order to be compared with the atomic number density profile, the molecular number density is multiplied by the number of atoms $N_S$ per

Figure 7.5: Streaming velocity for an average fluid density of $\rho = 0.544$, force field $F_e = 0.25$, molecular length of $N_S = 2$ and pore width $L = 10.2$. 
chain. In Fig. 7.4 we plot the molecular number density $\rho_m(y)$ multiplied by the number of atoms $N_S$ per chain for an average fluid density of $\rho = 0.544$, force field $F_e = 0.25$, molecular length of $N_S = 2$ and pore width $L = 10.2$. As is to be expected the fluid experiences large density oscillations particularly in the layers immediately adjacent to both walls. These oscillations become less significant towards the centre of the channel. Significant inhomogeneity appears only in the immediate vicinity of the walls. The molecular streaming velocity profile is displayed in Fig. 7.5. For a classical Navier-Stokes fluid undergoing planar Poiseuille flow the streaming velocity profile should be quadratic in $y$. We can see near the centre of the channel the streaming velocity profile is close to the quadratic profile, but deviations from the classical Navier-Stokes prediction are significant close to the interface between wall and fluid molecules. Abrupt changes in the streaming velocity close to the walls indicate slip boundary conditions.

If $S$ is the average intrinsic angular momentum per unit mass, $\omega$ is the streaming angular velocity and $\omega_i$ the angular velocity of molecule $i$, then

$$ S = \langle \Theta_i \cdot \omega_i \rangle = \Theta \omega $$

(7.9)

where

$$ \Theta_i = \sum_{\alpha \in i} m(r_{i\alpha}^2 I - r_{i\alpha} r_{i\alpha}')/\sum_{\alpha \in i} m $$

(7.10)

is the moment of inertia tensor per unit mass of molecule $i$, $\Theta = tr \langle \Theta_i \rangle /3$ and $I$ is the unit tensor. The index $\alpha$ labels the atoms in molecule $i$ and $r_{i\alpha}'$ is the vector from the center of mass of molecule $i$ to site $\alpha$ of the same molecule. We compute the streaming angular velocity as an average evaluated at the midpoint of each slab as described in number density and streaming velocity calculations.

In Fig. 7.6 we show the $x$, $y$ and $z$ components of the streaming angular velocity for an average fluid density of $\rho = 0.544$, force field $F_e = 0.25$, molecular length of $N_S = 2$ and pore width $L = 10.2$. $\omega_x$ and $\omega_y$ are zero and $\omega_z$ is a linear function of $y$ away from
Figure 7.6: $x$, $y$ and $z$ components of the streaming angular velocity for an average fluid density of $\rho = 0.544$, force field $F_c = 0.25$, molecular length of $N_S = 2$ and pore width $L = 10.2$. (a) $x$ component, $\omega_x$; (b) $y$ component, $\omega_y$; (c) $z$ component, $\omega_z$ (filled circles), and one half the vorticity, $1/2 \nabla \times u$ (solid line).

The walls. In Fig. 7.6(c) we also plot $1/2 \nabla \times u$, one half the vorticity. We note that Sarman and Evans [85] have proved that for homogeneous systems, in the absence of
external applied torques, \( \omega = \frac{1}{2} \nabla \times u \). Away from the walls this relation holds for our system. The streaming velocity shown in Fig. 7.5 and \( \omega_z \) in Fig. 7.6(c) qualitatively agree with the results obtained by Travis and Evans [84].

### 7.2.1 Structural properties

Probably the most interesting static properties in confined systems concern the shape and size of the chains in the vicinity of the surfaces and how they deviate from the respective bulk shape and sizes. The most appropriate quantities to describe these are the radius of gyration of the chain and the atomic and molecular density profiles across the pore.

**7.2.1.1 Atomic and molecular number density profiles**

In Fig. 7.7, we plot the atomic and molecular number densities for the indicated chain lengths for an average fluid density of \( \rho = 0.65 \), force field \( F_e = 0.5 \), and pore width of \( L = 7 \). The influence of the chain length on the atomic number density is minimal shown in Fig. 7.7(a). The molecular number density shown in Fig. 7.7(b) depends much more strongly on the chain length. For \( N_S = 2 \), the high density next to the wall is followed by a density depletion and a second density peak. Alternating density minima and maxima are developed across the fluid channel. The layered structure is more obvious if the chain length is small. By constraining the atoms in chains, an extra entropic cost is introduced and this entropy loss scales as \( N_S^{1/2} \) [86]. As the decrease of the layering further away from the walls is of an entropic nature, it would be expected that with increasing molecular length this entropic cost would increase. As a result, for the larger molecular lengths of \( N_S = 10 \) and 20 compared with \( N_S = 4 \) we observe the decrease of the first layer in the molecular number density profile as well as a fast decay towards the centre of the channel. As is physically plausible, the location of the first peak of
Figure 7.7: (a) Atomic number density $\rho_a(y)$ and (b) molecular number density $\rho_m(y)$ multiplied by the number of atoms $N_S$ per chain for an average fluid density of $\rho = 0.650$, force field $F_e = 0.5$, and pore width of $L = 7$. 
Figure 7.8: Comparison of atomic number density $\rho_a(y)$ and molecular number density $\rho_m(y)$ multiplied by the number of atoms $N_S$ per chain for average fluid densities of $\rho = 0.442$ and 0.650, force field $F_e = 0.012$, and molecular length of $N_S = 10$. (a) $L = 7, \rho = 0.442$; (b) $L = 14, \rho = 0.442$; (c) $L = 7, \rho = 0.650$; (d) $L = 14, \rho = 0.650$. 
Figure 7.9: Comparison of atomic number density $\rho_a(y)$ and molecular number density $\rho_m(y)$ multiplied by the number of atoms $N_S$ per chain for average fluid densities of $\rho = 0.442$ and $0.650$, force field $F_e = 0.012$, and molecular length of $N_S = 20$. (a) $L = 7, \rho = 0.442$; (b) $L = 14, \rho = 0.442$; (c) $L = 7, \rho = 0.650$; (d) $L = 14, \rho = 0.650$. 
Figure 7.10: Comparison of molecular number density, $\rho_m(y)$, multiplied by the number of atoms $N_S$ per chain for average fluid densities of $\rho = 0.442$ and 0.650, force field $F_c = 0.012$, and molecular lengths of $N_S = 10$ and $N_S = 20$. (a) $L = 7, \rho = 0.442$; (b) $L = 14, \rho = 0.442$; (c) $L = 7, \rho = 0.650$; (d) $L = 14, \rho = 0.650$. 
the molecular density profile shifts towards the centre of the channel as the chain length increases.

In Fig. 7.8 we show the comparison of atomic and molecular number densities for molecular length $N_S = 10$ and average fluid densities 0.442 and 0.650. In order to be compared with the atomic number density profile, the molecular density is multiplied by the number of atoms $N_S$ per chain. It is shown that the location of the first molecular layer is shifted toward the center of the pore around one atomic diameter from the location of the atomic layer. The molecular number density does not follow the atomic layering, since atoms in the first atomic layer have their center of mass of the molecular chain in the region of the depletion zone between the first and the second atomic layers. Similarly, we make the comparison of atomic and molecular number densities for the molecular length $N_S = 20$ in Fig. 7.9 and observe the same phenomenon. In Fig. 7.10 we present the comparison of molecular number densities for two different molecular lengths of $N_S = 10$ and 20, average fluid densities of $\rho = 0.442$ and $\rho = 0.650$, force field $F_e = 0.012$ and pore widths of $L = 7$ and $L = 14$. It can be seen that the layered structure is more obvious for the smaller chain length, which is the direct result of its entropic nature. As the pore width is increased to $L = 14$, the molecular number density is roughly a constant for both molecular lengths in the middle part of the pore shown in Fig. 7.10(b) and (d).

7.2.1.2 Extension of molecules

A polymer molecule exists in many different configurations due to many internal degrees of freedom. Because of this high degree of flexibility, we can depict a polymer chain with a very long piece of string. In order to study the conformation of such a polymer chain, we start with the simple model of Fig. 7.11, where we suppose that the chain follows a regular lattice. The portions of the polymer occupying the lattice points are called
Figure 7.11: The random walk model of the polymer. The filled circles are the beads and the thick lines are the bonds.

‘beads’, and the rods connecting the beads are called ‘bonds’. Let $b$ be the length of each bond.

If we assume that there is no correlation between the directions that different bonds take and that all directions have the same probability, the configuration of the polymer will be the same as a random walk on the lattice, and so the calculation we are about to perform can also be applied to the statistical properties of random walks.

The extent of spreading out or size of the polymer can be measured in terms of the average ‘end-to-end’ distance $R_e$, which is the absolute value of the vector $\mathbf{R}_e$ joining one end of the polymer to the other. If the polymer is made up of $N$ bonds, with $r_n$ the
vector of the \( n \)th bond, we have [87]

\[ \mathbf{R}_e = \sum_{n=1}^{N} \mathbf{r}_n . \]  

(7.11)

Clearly, the average value \( < \mathbf{R}_e > \) of \( \mathbf{R}_e \) is zero, since the probability of the end-to-end vector being \( \mathbf{R}_e \) is the same as it being \(-\mathbf{R}_e\) so that the two contributions cancel out. Therefore we will calculate \( < \mathbf{R}_e^2 > \), the average of the square of \( \mathbf{R}_e \), and express the size of the polymer by taking the square root of this quantity. From Eq. (7.11),

\[ R_e^2 \equiv < \mathbf{R}_e^2 > = \sum_{n=1}^{N} \sum_{m=1}^{N} < \mathbf{r}_n \cdot \mathbf{r}_m > . \]  

(7.12)

Since there is no correlation between the directions of different bond vectors, if \( n \neq m \) then \( < \mathbf{r}_n \cdot \mathbf{r}_m > = < \mathbf{r}_n > \cdot < \mathbf{r}_m > = 0 \). Therefore we find

\[ R_e^2 = \sum_{n=1}^{N} < \mathbf{r}_n^2 > = Nb^2 . \]  

(7.13)

Thus the size of the polymer is proportional to \( N^{1/2} \). What this means is that the polymer chain occupies a roughly spherical region in space with a diameter of order \( R_e = N^{1/2}b \). Up to now we have only been concerned with the overall size of the polymer. We are also interested in the spatial distribution of beads in the polymer chain. To do this, we introduce a length called the radius of gyration \( R_g \), which is defined as follows,

\[ R_g^2 = \frac{1}{2N^2} \sum_{n=1}^{N} \sum_{m=1}^{N} < (\mathbf{R}_n - \mathbf{R}_m)^2 > , \]  

(7.14)

where \( \mathbf{R}_n (n = 1, 2, \ldots, N) \) represents the position vector of a bead of the chain. It happens that the radius of gyration \( R_g \) is a more convenient way of expressing the size of a polymer than the average of the square of the end-to-end vector \( < \mathbf{R}_e^2 > \). The radius of gyration can be defined not only for linear chain polymers but also for polymers with branched structure.
Notice that $R_g^2$ also equals the square of the average distance between the beads and the centre of mass of the polymer. The position of the centre of mass is defined by

$$R_G = \frac{1}{N} \sum_{n=1}^{N} R_n .$$

(7.15)

Using this, it is easy to show that $R_g$ can be rewritten as

$$R_g^2 \equiv \frac{1}{N} \sum_{n=1}^{N} < (R_n - R_G)^2 >.$$  

(7.16)

Let us calculate the radius of gyration for an ideal chain. When $|n - m|$ is large, $R_n - R_m$ of an ideal chain has a Gaussian distribution with variance $|n - m|b^2$, which means that

$$< (R_n - R_m)^2 > = |n - m|b^2 .$$

(7.17)

Therefore

$$R_g^2 = \frac{1}{2N^2} \sum_{n=1}^{N} \sum_{m=1}^{N} |n - m|b^2 .$$

(7.18)

For large $N$ the summation can be replaced by an integration:

$$R_g^2 = \frac{b^2}{2N^2} \int_{0}^{N} dn \int_{0}^{N} dm |n - m| = \frac{1}{6} Nb^2 .$$  

(7.19)

Therefore, the ratio of $R_e^2$ to $R_g^2$ is the constant 6 for an ideal chain.

In order to quantify the extension of the confined molecules, we divide the channel between the two walls into slices and for each slice the squared average radius of gyration and the squared average end-to-end distance are computed as an average evaluated at the midpoint of each slice. In Fig. 7.12 and Fig. 7.13 we show the squared average radius of gyration, $R_g^2$, and the squared average end-to-end distance, $R_e^2$, as a function of $y$ for average fluid densities of $\rho = 0.442$ and 0.650, force field $F_e = 0.5$, pore width $L = 7$ and molecular lengths of $N_S = 2, 4, 10$ and 20. Our results show that $R_g^2$ and $R_e^2$ for $N_S = 2$ are roughly constants while there are substantial fluctuations for $N_S = 4, 10$ and 20. The
Figure 7.12: Squared average radius of gyration as a function of $y$ for average fluid densities of $\rho = 0.442$ and 0.650, force field $F_e = 0.5$, and pore width of $L = 7$. (a) $N_S = 2$; (b) $N_S = 4$; (c) $N_S = 10$; (d) $N_S = 20$. 

1.5. Simulation results

- $\rho = 0.442$
- $\rho = 0.650$

$R_g^2(y)$

$y$

$y$

$R_g^2(y)$

$y$

$y$
Figure 7.13: Squared average end-to-end distance as a function of $y$ for average fluid densities of $\rho = 0.442$ and $0.650$, force field $F_e = 0.5$, and pore width of $L = 7$. (a) $N_S = 2$; (b) $N_S = 4$; (c) $N_S = 10$; (d) $N_S = 20$. 
reason is that for 2-site molecules the centre of mass of each molecule locates on the line between the two sites and the distance between these two sites does not change much. However, for molecules of $N_S = 4, 10$ and $20$, the configurations of the molecules are changed, and the centres of mass of the molecules locate outside of the molecular chains, as depicted in Fig 7.14. As is expected, both the radius of gyration and the end-to-end distance are lower for the higher average density systems ($\rho = 0.650$) compared to the lower average density systems ($\rho = 0.442$). The higher density restricts the extension of molecules due to more frequent and stronger inter- and intra-molecular interactions, thus reducing the effective space that one molecule can occupy.

The confinement induces fluctuations of the squared average radius of gyration and the squared average end-to-end distance across the pore. These fluctuations are strongly affected by the fluctuating molecular number density. In Fig. 7.15 we show the squared average radius of gyration with superposed and magnified molecular number density vs $y$ for an average fluid density of $\rho = 0.650$, force field $F_e = 0.012$, molecular length of $N_S = 10$, and pore width $L = 7$. A decrease in the squared average radius of gyration reflects the shrinkage of molecules resulting from the increase in the molecular number density. On the other hand, an increase in the squared average radius of gyration indicates the expansion of molecules due to the decrease in the molecular number density. For the given values of the pore size, external force and chain length, decreases in $R_g^2$ and $R_e^2$ with the increasing average fluid density indicate that the stretching of the molecules decreases when average fluid density increases as the compactness of space prevents molecules from expansion. As is to be expected, for $N_S = 4, 10, 20$ and the pore width of $L = 7$ both $R_g^2$ and $R_e^2$ experience strong oscillations which decay in strength towards the centre of the channel. If the pore width is doubled, they are almost flat away from the walls shown in Figs. 7.16(b), (d) and 7.17(b), (d).
Figure 7.14: Schematic representation of the configurations of the 2-site and 4-site chain molecules. \( c \) is the centre of mass of each molecule.

Figure 7.15: Squared average radius of gyration with superposed and magnified molecular number density \( N \rho_m \) vs \( y \) for an average fluid density of \( \rho = 0.650 \), force field \( F_e = 0.012 \), molecular length of \( N_S = 10 \), and pore width \( L = 7 \).
Figure 7.16: Squared average radius of gyration vs $y$ for average fluid densities of $\rho = 0.442$ and 0.650, force field $F_e = 0.012$. (a) $L = 7$, $N_S = 10$; (b) $L = 14$, $N_S = 10$; (c) $L = 7$, $N_S = 20$; (d) $L = 14$, $N_S = 20$. 

\[ R_g^2(y) \]
Figure 7.17: Squared average end-to-end distance as a function of $y$ for average fluid densities of $\rho = 0.442$ and $0.650$, force field $F_c = 0.012$. (a) $L = 7$, $N_S = 10$; (b) $L = 14$, $N_S = 10$; (c) $L = 7$, $N_S = 20$; (d) $L = 14$, $N_S = 20$. 
Figure 7.18: Ratio of the squared average end-to-end distance to the squared average radius of gyration as a function of $y$ for average fluid densities of $\rho = 0.442$ and 0.650, force field $F_e = 0.5$, and pore width of $L = 7$. (a) $N_S = 2$; (b) $N_S = 4$; (c) $N_S = 10$; (d) $N_S = 20$. 
In Fig. 7.18(a), (b), (c) and (d) we plot the ratio of the squared average end-to-end distance to the squared average radius of gyration as a function of $y$ for $N_S = 2, 4, 10$ and 20 respectively for average fluid densities of $\rho = 0.442$ and 0.650, force field $F_e = 0.5$, and pore width of $L = 7$. It can be seen that for $N_S = 2$ and 4, the ratios are less than six, a theoretical value of the ratio for ideal chains as discussed earlier [87, 88]. For $N_S = 10$ and 20, the ratios fluctuate around six. As is physically plausible, the ratios deviate from the theoretical prediction for the ideal chains as a result of the confinement induced change of the molecular size which is strongly correlated with the molecular number density variation shown in Fig. 7.15. As the pore width increases to $L = 14$, at the central region of the pore the ratios are very close to six which is in quantitative agreement with theory shown in Fig. 7.19(b) and (d). It should be stressed that the molecular theories on chain statistics are valid only for homogeneous systems. Our systems only approach homogeneity near the centre of the channel, which is why agreement with theory is best here.

The ideal chain model only takes into account the short range interactions between beads which are located close to each other along the chain. Thus this model permits a chain to loop back onto itself so that beads which are widely separated along the chain will occupy the same region in space. Of course this is a physical impossibility since each bead possesses its own finite volume. In the lattice model of a polymer, this effect can be accounted for by imposing the condition that two beads cannot occupy the same lattice site. In general, this type of condition is called the ‘excluded volume effect’. If we model the polymer as a connected path on a lattice, the excluded volume effect will correspond to the condition that the path cannot pass through any sites that have been traversed previously. This is called a ‘self avoiding walk’, and the polymer thus represented is called an ‘excluded volume chain’. An ‘ideal chain’ polymer corresponds to a random walk without the excluded volume effect.
Figure 7.19: Ratio of the squared average end-to-end distance to the squared average radius of gyration as a function of $y$ for average fluid densities of $\rho = 0.442$ and $0.650$, force field $F_e = 0.012$. (a) $L = 7$, $N_S = 10$; (b) $L = 14$, $N_S = 10$; (c) $L = 7$, $N_S = 20$; (d) $L = 14$, $N_S = 20$. 
The beads of the Lennard-Jones chains used in this work can not overlap each other due to the $LJ$ potential which is strongly repulsive at short distances. Therefore there exist regions of excluded volume in these chains. The beads of an ideal chain are not restricted in position and are therefore free to overlap. The average size of an excluded volume chain is larger than that of an ideal chain, which can be easily seen as follows. For an ideal chain, there is a greater possibility of beads overlapping in a compressed polymer coil than in one of large size. Thus when we add the restriction that no overlapping is permitted, we would expect the size distribution to be shifted to larger values, and so the excluded volume chain is larger than the ideal chain of the same length. The effect of the excluded volume is that the size of excluded volume chains is proportional to $N^{3/5}$, and not $N^{1/2}$ [87, 89]. The statistical properties of excluded volume chains have been extensively investigated, and for large $N$ it has been found that the size obeys the following formula [87, 89]:

$$R_g = AN^\nu,$$

(7.20)

where $N = N_S - 1$ is the number of bonds in the chain. The prefactor $A$ is model dependent, whereas the exponent $\nu$ is very close to the value 0.6.

In Fig. 7.20 we plot average radius of gyration, $R_g$, at the center of the fluid channel vs the number of bonds per chain, $N_S - 1$, for an average fluid density of $\rho = 0.442$, force field $F_e = 0.5$, and pore width $L = 7$. As is to be expected, it is not a linear function since we really do not expect very short chains ($N_S = 2, 4$) to be valid for the power law described above. In order to quantitatively compare our data with the random walk ideal chain model and the model which takes into account the excluded volume effect, in Fig. 7.21 we show the logarithm of average radius of gyration, $\ln(R_g)$, vs the logarithm of the number of bonds, $\ln(N_S - 1)$, only for $N_S = 10$ and 20. We fit our data to a linear function and determine the value of exponent $\nu$ for our system. Our simulation results
7.2 Simulation results

Figure 7.20: Average radius of gyration, $R_g$, at the centre of fluid channel vs the number of bonds $N_S - 1$ for an average fluid density of $\rho = 0.442$, force field $F_e = 0.5$, and pore width $L = 7$.

show that for $N_S = 10$ and 20 the radius of gyration follows the power law $R_g \propto N^\nu$, where $\nu = 0.60 \pm 0.04$. The larger exponent than the equilibrium value of 0.5 for a melt is a consequence of stretching of the molecules under shear [90].

7.2.2 Dynamical properties

7.2.2.1 Streaming velocity and strain rate

The molecular streaming velocity profiles are displayed in Fig. 7.22 for average fluid densities of $\rho = 0.442$ and 0.650, force field $F_e = 0.012$, pore widths of $L = 7$ and 14 and molecular lengths of $N_S = 10$ and 20. The maximum streaming velocity decreases for fixed field strength and chain length as the channel width decreases due to increased frictional resistance between wall and fluid. For fixed pore width, force strength, and
Figure 7.21: Logarithm of average radius of gyration at the centre of fluid channel, \( \ln(R_g) \), vs the logarithm of the number of bonds, \( \ln(N_S - 1) \), for \( N_S = 10 \) and 20 (filled squares) with linear fit (solid line) for an average fluid density of \( \rho = 0.442 \), force field \( F_e = 0.5 \), and pore width \( L = 7 \). The linear fit gives \( R_g = 0.4N^\nu \), where \( \nu = 0.60 \pm 0.04 \).

As the molecular length increases, the strain rate in the middle of the pore is close to zero if shear stress is less than a certain value and the velocity gradients are concentrated near the walls shown in Fig. 7.23 for average fluid densities of \( \rho = 0.442 \) and 0.650, force field \( F_e = 0.012 \), pore widths of...
Figure 7.22: Molecular streaming velocity as a function of $y$ for average fluid densities of $\rho = 0.442$ and 0.650, force field $F_e = 0.012$. (a) $L = 7$, $N_S = 10$; (b) $L = 14$, $N_S = 10$; (c) $L = 7$, $N_S = 20$; (d) $L = 14$, $N_S = 20$.

$L = 7$ and 14, and molecular lengths of $N_S = 10$ and 20. Plug flow occurs when the Reynolds number (in our case defined as $\rho u_x L/\eta$) is small, i.e., the ratio of inertial to viscous forces is low, and when there exists some degree of slip. As the viscosity of the
Figure 7.23: Strain rate as a function of $y$ for average fluid densities of $\rho = 0.442$ and 0.650, force field $F_e = 0.012$. (a) $L = 7, N_S = 10$; (b) $L = 14, N_S = 10$; (c) $L = 7, N_S = 20$; (d) $L = 14, N_S = 20$. 
fluid increases for a particular pore size (either due to higher average density or larger chain length) we might expect plug flow to occur. This is the case for the $L = 7$, $N_S = 10$ system (Fig. 7.22(a)) where the low density fluid displays a typical quadratic Poiseuille flow profile away from the walls, but the higher density fluid shows a flatter velocity profile. Similarly for the $L = 7$, $N_S = 20$ system (Fig. 7.22(c)), both densities display plug flow behavior, because now the viscosity of the $N_S = 20$ polymer fluid is larger than the viscosity of the $N_S = 10$ system. It is well known that for low molecular weight polymers the viscosity scales as $\eta \propto N_S$ for a homogeneous fluid [92]. This has been confirmed by a number of independent NEMD simulation studies [93, 94, 95]. In all cases the degree of slip is not large but the velocity profiles are qualitatively similar to those obtained in actual experimental measurements of high density polyethylene in a slit die [96].
Figure 7.25: \( z \) component of the streaming angular velocity (filled circles) and one half the vorticity, \( \frac{1}{2} \nabla \times \mathbf{u} \) (solid line) for an average fluid density of \( \rho = 0.650 \), force field \( F_e = 0.5 \), and pore width of \( L = 7 \). (a) \( N_S = 2 \); (b) \( N_S = 4 \); (c) \( N_S = 10 \); (d) \( N_S = 20 \).
7.2.2.2 *Molecular rotation*

The streaming angular velocity of polymers is determined by Eq. (7.9). Fig. 7.24 shows a plot of the $z$ component of the streaming angular velocity as a function of $y$ for the indicated chain lengths. The absolute value of $\omega_z$ decreases as the chain length increases for the given values of the overall system density and external force field. In Fig. 7.25 we plot the $z$ component of the streaming angular velocity (filled circles) and one half the vorticity (solid line) for $N_S = 2, 4, 10$ and $20$. We note that away from the boundaries, in the absence of external applied torques, the relation, $\omega = 1/2 \nabla \times u$, holds for our systems. Away from the boundaries our simulation results of the streaming angular velocity are in good agreement with the theory proved by Sarman and Evans [85]. Deviations from the theory can only be observed in the vicinity of the walls, where there are abrupt changes in the streaming velocity profiles shown in Fig. 7.22, and consequently, there are significant changes in $1/2 \nabla \times u$. In Fig. 7.26 we show the squared average radius of gyration with superposed absolute value of $\omega_z$ vs $y$ for an average fluid density of $\rho = 0.650$, force field $F_e = 0.012$, molecular length of $N_S = 10$, and pore width $L = 7$. We observe that the maxima of the $|\omega_z|$ locate at the points where the squared average radius of gyration goes to minima. This is physically plausible. The squared average radius of gyration affects the moment of inertia directly. The lower the radius of gyration, the lower the moment of inertia. Thus, if the angular momentum is conserved, a lower moment of inertia corresponds to a higher streaming angular velocity. This is a similar effect to that experienced by an ice skater who, if she wishes to spin faster, simply needs to contract her body. The conservation of her angular momentum dictates that by doing this she must by necessity rotate at a faster rate.

7.2.2.3 *Shear stress*

We plot $P_{yx}$ as a function of $y$ for the same systems mentioned above in Fig. 7.27.
Figure 7.26: Squared average radius of gyration with superposed absolute value of $\omega_z$ vs $y$ for an average fluid density of $\rho = 0.650$, force field $F_e = 0.012$, molecular length of $N_S = 10$, and pore width $L = 7$.

$P_{yx}$ is integrated as the molecular stress, i.e., $P_{yx}(y) = N_S F_e \int_0^y \rho_m(y')dy'$. Deviations from linearity are only significant in the vicinity of the walls. The absolute value of $P_{yx}$ increases with the chain length in accord with the pressure.
7.2 Simulation results

Figure 7.27: Element of the molecular pressure tensor $P_{yx}(y)$ as a function of $y$ for an average fluid density of $\rho = 0.650$, force field $F_e = 0.5$, pore width of $L = 7$, and molecular lengths of $N_S = 2, 4, 10$ and $20$. 
In this thesis, the method of Non-Equilibrium Molecular Dynamics (NEMD) computer simulations was employed to investigate the structural and dynamic properties of atomic and polymeric fluids undergoing planar Poiseuille flow. The fluids are confined between channels of several atomic diameters width and subjected to a uniform external force field. For the atomic fluids, the dependences of the number density, streaming velocity, shear stress, strain rate, and viscosity kernel as functions of channel width, average fluid density and field strength were investigated. As to the polymeric fluids, in addition to all the quantities studied for atomic fluids, the extension and rotation of the confined polymers were explored.

In Chapter 1 the background of the theoretical and experimental studies of atomic and molecular fluids was sketched, including the computational study of nano-scale flows. In Chapter 2 the basic theoretical framework of this thesis was introduced, in particular the microscopic expressions for the continuity equations in hydrodynamics. We then proceeded to derive the method of planes (MOP) expressions for the pressure tensor and heat flux vector for an atomic fluid under the influence of three-body forces.

In Chapter 3 the basic simulation methodologies used in this work were described. We gave a brief introduction to the equations of motion, force calculations, periodic boundary conditions, time integration algorithms, Gaussian constraints and the SLLOD algorithm. In Chapter 4, a concise description of parallel programming with a message
passing interface (MPI) was provided. The emphasis was placed on the demonstration of the frequently used MPI subroutines by analysing the sample of parallelized program for Poiseuille flow which we have developed and implemented in this work.

In Chapter 5 the validity of the theoretical derivations obtained in Chapter 2 was checked by performing non-equilibrium molecular dynamics computer simulations of a confined field driven fluid acted upon by a two-body Barker-Fisher-Watts (BFW) force coupled with the Axilrod-Teller (AT) three-body force. The MOP calculations were found to be in excellent agreement with independent calculations based upon direct integration of the hydrodynamic momentum and energy continuity equations. Our simulation results showed that the isotropic pressure is sensitive to the presence of three-body forces, whereas the shear stress and heat flux vector seem to be largely independent of them. Further work is required to study the potential, temperature and density dependence on the relative magnitudes of the two to three-body force contributions. While such effects are clearly small for noble gas fluids such as argon, they will most likely play a significant and important role for heavier atomic and molecular fluids and liquid metals. It is hoped that our MOP expressions will be useful for the study of such liquids in the future.

In Chapter 6 we used the non-local linear hydrodynamic constitutive model for computing a viscosity kernel, a function of compact support, for inhomogeneous nonequilibrium fluids. We applied the model to a simple atomic fluid undergoing planar Poiseuille flow in a channel of several atomic diameters width. We validate our Discrete Fourier Transforms applied in this work and analyse the behaviour of our system by using suitable test functions to study the effects of truncations in our system. Our results show that the viscosity kernel, \( \eta(y) \), has a peak at \( y = 0 \), and decays to zero as \( y \) increases, where \( y \) represents the separation between where we want to know the stress and where the strain rate is. Physically, it means that the strain rate at the location where we want
to know the stress contributes most to the stress, and the contribution of the strain rate becomes less significant as the relative distance \( y \) increases. The integral over the viscosity kernel, which is the value of \( \tilde{\eta}(k_y) \) when \( k_y = 0 \), compares reasonably well with the Navier-Stokes viscosity. We demonstrate that there are limitations in the model when it is applied to our confined fluids due to the effect of domain restriction on inverse convolution.

Finally, in Chapter 7 we studied the rheology of highly confined polymeric fluids undergoing Poiseuille flow. The structural and dynamic properties as simulated via NEMD were discussed in terms of the density profiles, the squared average radius of gyration, the squared average end-to-end distance, streaming velocity, strain rate, shear stress, and streaming angular velocity. Our simulation results show that sufficiently far from the walls, the radius of gyration for molecules under shear in the middle of the channel follows the power law \( R_g \propto N^\nu \), where \( N \) is the number of bonds and the exponent has a value \( \nu = 0.60 \pm 0.04 \), which is larger than the melt value of 0.5 for a homogeneous equilibrium fluid. Under the conditions simulated, we find that viscous forces dominate the flow, resulting in the onset of plug-like flow velocity profiles with some wall slippage. An examination of the streaming angular velocity displays a strong correlation with the radius of gyration, being maximum in those regions where \( R_g \) is minimum and vice-versa. The angular velocity is shown to be proportional to half the strain rate sufficiently far from the walls, consistent with the behaviour for homogeneous fluids in the linear regime.

Further work in this field should focus on the influence of wall affinities on the spatially dependent structural and dynamic properties and boundary conditions for inhomogeneous nonequilibrium fluids using nonequilibrium molecular dynamics simulations. Slip boundary conditions were suggested in several molecular dynamics simulations [78, 79, 97, 98]. Recently rate-dependent slip [99] and shear-dependent slip [100] of
Newtonian fluids were detected. We demonstrated in this work, through direct measurement of the streaming velocity, that slip is also detectable in a highly confined atomic fluid bounded by solid surfaces. Further investigation on the slip length and dynamic friction is needed as nano-scale flow is dominated by surface properties [101, 102, 103].

It is clear that the simple homogeneous linear constitutive equation used to model confined fluids is limited to low densities and relatively large pore widths. Progress in this work will require theoretical methodologies to compute the inhomogeneous kernel, $\eta(r, r')$. This is a significant theoretical challenge, but an important one in the science of nanofluidics.

Finally, it might also be desirable to simulate rheological properties for much longer chains and polymer solutions to provide a better understanding of the behaviour of highly confined polymeric fluids at the molecular level. To be able to apply the simulation results to industrially relevant polymeric systems, techniques to enable accurate mapping between simulated and real molecules need to be established.
BIBLIOGRAPHY


[31] V. P. Sokhan and N. Quirke, Molecular Simulation 30, 217 (2004).


