A Practical Introduction to the Lattice Boltzmann Method

Alexander J. Wagner
Department of Physics
North Dakota State University
alexander.wagner@ndsu.edu
Fargo, March 2008

Contents

Preface

The lattice Boltzmann method is increasingly attracting researchers in many areas from turbulence to multi-phase flow in porous media. Several textbooks have been written to address the need of students to learn about this relatively new method.

The aim of this introduction is to provide a succinct description of the field and to provide students with sample codes so that they can immediatly apply their knowledge to practical applications. Source codes, which are provided under the GNU copyright, can be downloaded at http://www.physics.ndsu.nodak.edu/wagner/LB.html.

This manuscript is still in a rough state and is continuously beeing improoved. Your comments and corrections are always welcome.

A.J. Wagner, Fargo, March 2008.

Chapter 1

Introduction

When simulating traditional fluid dynamics one principally thinks of the continuity and Navier-Stokes equations. The is given by

$$\partial_t \rho + \nabla(\rho u) = 0 \tag{1.1}$$

and expresses that the density ρ is a locally conserved quantity and can only be changed if it is advected away by a flow with velocity $\mathbf{u}$. So the change of the density ρ is given by the divergence of the current $\rho\mathbf{u}$. So the only physical content of this equation is that the density in this system is locally conserved. We need to couple this with an equation for the local velocity $\mathbf{u}$ and this equation is the , given by

$$\partial_t(\rho\mathbf{u}) + \nabla(\rho\mathbf{u}\mathbf{u}) = -\nabla p + \nabla\sigma \tag{1.2}$$

where p is the local pressure and σ is a viscous stress tensor. For the stress is given by

$$\sigma = \eta[\nabla\mathbf{u} + (\nabla\mathbf{u})^T] + \nu\nabla.\mathbf{u}\mathbf{1}. \tag{1.3}$$

To numerically integrate these equations we need to discretize the derivatives requiring first and second order derivatives. There are many possible discretizations of these equations and finding approproate discrete operators is the subject of the field of . Many of these possible discretization will only conserve mass and momentum up to the order of discretization.

An alternative approach to these computational fluid dynamics simulations was invented in the late 1980s with the lattice gas methods. These methods allowed particles to move on a discrete lattice and local collisions conserved mass and momentum. Because the continuity and Navier Stokes equations are only continuous forms of the mass and momentum conservation statements and method that locally conserves mass and momentum will obey some kind of continuity and Navier Stokes equations and it was shown that the lattice gas methods could be used to simulate (rather noisy) hydrodynamics. However, the lattice gas methods had several drawbacks consisting mainly of their noisy nature and the apperance of some additional terms in the Navier Stokes level equations that limited their success. It was then discovered that instead of discrete particles a density distribution could be adevecting which eliminated the noisyness of the method and allowed for a more general collision operator. This is the lattice Boltzmann method which has been extraordinarily successful for many applications including turbulence, multi-component and multi-phase flows as well als additional applications, including simulations of the Schrödinger equation. We start by introducing the Boltzmann equation and describe how the hydrodynamic equations can be obtained from the Boltzmann equation. We then show how the Boltzmann equation can be simulated by a very simple numerical method leading to the same hydrodynamic equations.

Next we extend the lattice Boltzmann method to systems that are not typically described by a Boltzmann equation, namely non-ideal gases and phase-separating multi-component mixtures.

Chapter 2

The Boltzmann equation

We have already seen[1] that the dynamics of the Boltzmann equation always mimimizes the H-Functional given by

$$H(t) = \int d\mathbf{x}\, d\mathbf{v}\; f(\mathbf{x}, \mathbf{v}, t) \log\left(f(\mathbf{x}, \mathbf{v}, t)\right). \tag{2.1}$$

So we can conclude that the equilibrium distribution function f^0 in a volume V for a given density n, mean momentum $n\mathbf{u}$ and energy $n\epsilon = 1/2\, n\mathbf{u}^2 + 3/2\, n\theta$ will minimize the H-functional. We can use Lagrangian multipliers to minimize this functional. With Lagrangian multipliers the H-functional reads

$$\begin{aligned} H(t) \quad = \quad & \int d\mathbf{x}\, d\mathbf{v}\; f(\mathbf{x}, \mathbf{v}, t) \log f(\mathbf{x}, \mathbf{v}, t) \\ & -\lambda_1 (nV - \int d\mathbf{x}\, d\mathbf{v}\; f(\mathbf{x}, \mathbf{v}, t)) \\ & -\lambda_{2\alpha} (nu_\alpha V - \int d\mathbf{x}\, d\mathbf{v}\; f(\mathbf{x}, \mathbf{v}, t) v_\alpha) \\ & -\lambda_3 (n\epsilon V - \int d\mathbf{x}\, d\mathbf{v}\; f(\mathbf{x}, \mathbf{v}, t) \frac{\mathbf{v}^2}{2} \end{aligned} \tag{2.2}$$

Now we calculate the functional derivative of the H-functional with respect to the distribution function. Since the equilibrium distribution minimizes the H-functional this derivative has to vanish when the distribution is the equilibrium distribution $f = f^0$.

$$0 \overset{!}{=} \left. \frac{\delta H(t)}{\delta f} \right|_{f=f^0} = 1 + \log(f^0) + \lambda_1 + \lambda_{2\alpha} v_\alpha + \lambda_3 \frac{\mathbf{v}^2}{2} \tag{2.3}$$

We can solve this for the equilibrium distribution:

$$f^0 = \exp\left[-1 - \lambda_1 + \frac{\lambda_2^2}{2\lambda_3} - \frac{\lambda_3}{2} \left(\frac{\lambda_2}{\lambda_3} + \mathbf{v} \right)^2 \right] \tag{2.4}$$

Introducing a new set of Lagrangian multipliers a, b_α and c we can also write this as

$$f^0 = a \exp\left[\frac{(\mathbf{b} - \mathbf{v})^2}{c} \right] \tag{2.5}$$

Note that this expression does not depend on spatial variables. So we know that the solution will be homogeneous. We now find the Lagrange multipliers by invoking the conservation laws

$$nV = \int d\mathbf{x}\, d\mathbf{v}\; f^0 \quad nu_\alpha V = \int d\mathbf{x}\, d\mathbf{v}\; f^0 v_\alpha \quad \frac{n\mathbf{u}^2}{2} V + \frac{3}{2} n\theta V = \int d\mathbf{x}\, d\mathbf{v}\; f^0 \frac{\mathbf{v}^2}{2} \tag{2.6}$$

---

[1] Kerson Huang, Statistical Mechanics, 2nd ed., chapter 4.

Solving these equations for a, b_α and c we find that the equilibrium distribution is given by

$$f^0 = \frac{n}{(2\pi\theta)^{3/2}} \exp\left[-\frac{(\mathbf{v}-\mathbf{u})^2}{2\theta}\right] \tag{2.7}$$

which is known as the Maxwell-Boltzmann distribution.

Problems

2.1: Which function $\phi(x)$ minimizes the functional $\Psi[\phi] = \int_a^b -\phi^2(x) + \phi^4(x)dx$?

2.2: Which function $\phi(x)$ minimizes the functional $\Psi[\phi] = \int_a^b -\phi^2(x) + \phi^4(x)dx$ if the integral $\int_a^b \phi(x)dx = c(b-a)$ is a constant? Do you obtain different kinds of solutions for certain ranges of the constant c?

2.3: Derive (2.7) [*i.e.* determine a, $\mathbf{b}$, and c for equation (2.5)].

Chapter 3

Derivation of the hydrodynamic equations from the Boltzmann equation

3.1 Introduction

In this lecture we will examine the hydrodynamic limit of the Boltzmann equation and derive the transport equations for the macroscopic quantities from first principles. We will see that the macroscopic equations of motion are simply the conservation equations for continuous fields. Because of the general concepts involved the transport equations we derive are not only applicable for dilute gases, which we require for the Boltzmann equation to apply, but also for much denser fluids.

This is the reason that recently a numerical method called "lattice Boltzmann" has been developed for the simulation of fluids. We will cover the lattice Boltzmann approach in the next lecture.

3.2 The Boltzmann equation

The Boltzmann equation we derived in the last lecture is given by

$$\partial_t f + v\partial_x f + F\partial_v f = \int dv_1' dv_2' dv_2 (f_1' f_2' - f_1 f_2) P_{12\to 1'2'} \tag{3.1}$$

Solving this equation analytically is very challenging and can only be done for special cases.

3.2.1 The BGK approximation

There Bhatnagar, Gross and Krook noticed, however, that the main effect of the collision term is to bring the velocity distribution function closer to the equilibrium distribution. The equilibrium distribution is given by (2.7)

$$f^0(v) = \frac{n}{(2\pi\theta)^{3/2}} e^{-(v-u)^2/2\theta} \tag{3.2}$$

where n is the number density of particle, u is the mean velocity and $\theta = kT$ is the temperature. These macroscopic quantities are given by moments of the distribution function:

$$\int f = n \tag{3.3}$$

$$\int f v_\alpha = n u_\alpha \tag{3.4}$$

$$\int f v^2 = n u^2 + 3n\theta \tag{3.5}$$

The simplest way of approximating the the collision term is by using a single relaxation time approximation

$$\int dv_1' dv_2' dv_2 (f_1' f_2' - f_1 f_2) P_{12\rightarrow 1'2'} \approx \frac{1}{\tau}(f^0 - f) \tag{3.6}$$

One important feature of this approximation is that mass, momentum and energy are are still exactly conserved by the collision term. With this approximation the Boltzmann equation reads

$$\partial_t f + v\partial_x f + F\partial_v f = \frac{1}{\tau}(f^0 - f) \tag{3.7}$$

We can use this expression to approximate the probability density function by the equilibrium distribution and its derivatives:

$$f = f^0 - \tau(\partial_t f + v\partial_x f + F\partial_v f) \tag{3.8}$$

3.2.2 Moments of the equilibrium distribution function

Because we can express the distribution function in terms of the equilibrium distribution function we will require several velocity moments of the equilibrium distribution function (2.7). They are

$$\begin{aligned}
\int f^0 &= n, && (3.9)\\
\int f^0(v_\alpha - u_\alpha) &= 0, && (3.10)\\
\int f^0(v_\alpha - u_\alpha)(v_\beta - u_\beta) &= n\theta\delta_{\alpha\beta}, && (3.11)\\
\int f^0(v_\alpha - u_\alpha)(v_\beta - u_\beta)(v_\gamma - u_\gamma) &= 0, && (3.12)\\
\int f^0(v_\alpha - u_\alpha)(v_\beta - u_\beta)(\mathbf{v} - \mathbf{u})^2 &= 5n\theta^2\delta_{\alpha\beta}. && (3.13)
\end{aligned}$$

This is easily derived using Gaussian integrals (see appendix D).

3.3 Mass conservation

Integrating over the Boltzmann equation we obtain for the mass conservation equation

$$\begin{aligned}
\partial_t \int dv f + \partial_\alpha \int dv f v_\alpha + F \int dv \partial_v f &= \frac{1}{\tau}\int dv (f^0 - f)\\
\Leftrightarrow \partial_t n + \partial_\alpha (n u_\alpha) &= 0 && (3.14)
\end{aligned}$$

which is the continuity equation.

3.4 Momentum conservation

Multiplying the Boltzmann equation with v_α and integrating we obtain as a momentum conservation equation

$$\begin{aligned}
\partial_t \int dv f v_\alpha + \partial_\beta \int dv f v_\alpha v_\beta + F_\beta \int dv \partial_{v_\beta} f v_\alpha &= \frac{1}{\tau}\int dv (f^0 - f) v\\
\partial_t (n u_\alpha) + \partial_\beta \int dv f v_\alpha v_\beta - n F_\alpha &= 0
\end{aligned}$$

We now need to use equation (3.8) to approximate

$$\int dv f v_\alpha v_\beta = \int dv f^0 v_\alpha v_\beta - \tau(\partial_t \int dv f^0 v_\alpha v_\beta + \partial_\gamma \int dv f^0 v_\alpha v_\beta v_\gamma + nF_\alpha u_\beta + n u_\alpha F_\beta) + O(\partial^2) \tag{3.15}$$

To first order in derivatives our conservation equation now reads

$$\partial_t(nu_\alpha) + \partial_\beta(nu_\alpha u_\beta) = -\partial_\alpha(n\theta) + nF_\alpha. \tag{3.16}$$

This equation is known as the Euler equation. Using the continuity equation (3.14) we can also write it as

$$\partial_t u_\alpha + u_\beta \partial_\beta u_\alpha = -\frac{1}{n}\partial_\alpha(n\theta) + F_\alpha. \tag{3.17}$$

To calculate the equations of motion to second order in the derivatives we need to evaluate the higher order terms in eqn. (3.15).

$$\begin{aligned}
& \partial_t(\int f^0 v_\alpha v_\beta) \\
= \; & \partial_t(nu_\alpha u_\beta + n\theta\delta_{\alpha\beta}) \\
= \; & \partial_t(nu_\alpha)u_\beta + nu_\alpha\partial_t u_\beta + \partial_t n\theta\delta_{\alpha\beta} + n\partial_t\theta\delta_{\alpha\beta} \\
= \; & -\partial_\gamma(nu_\alpha u_\gamma)u_\beta - \partial_\alpha(n\theta)u_\beta - nF_\alpha u_\beta \\
& -nu_\alpha u_\gamma\partial_\gamma u_\beta - u_\alpha\partial_\beta(n\theta) - nu_\alpha F_\beta \\
& -\partial_\gamma(nu_\gamma)\theta\delta_{\alpha\beta} - nu_\gamma\partial_\gamma(\theta\delta_{\alpha\beta}) - \frac{2}{3}\partial_\gamma u_\gamma\theta \\
= \; & -\partial_\gamma(nu_\alpha u_\beta u_\gamma) - \partial_\beta(n\theta)u_\alpha - \partial_\alpha(n\theta)u_\beta - n(F\alpha u_\beta + u_\alpha F_\beta) \\
& -\partial_\gamma(n\theta u_\gamma)\delta_{\alpha\beta} - \frac{2}{3}n\theta\partial_\gamma u_\gamma
\end{aligned} \tag{3.18}$$

where we used equations (3.16),(3.17) and (3.24). We need to combine this with

$$\begin{aligned}
& \partial_\gamma \int f^0 v_\alpha v_\beta v_\gamma \\
= \; & \partial_\beta(n\theta u_\alpha) + \partial_\alpha(n\theta u_\beta) + \partial_\gamma(n\theta u_\gamma)\delta_{\alpha\beta} + \partial_\gamma(nu_\alpha u_\beta u_\gamma)
\end{aligned} \tag{3.19}$$

where we used equation (3.12). Combining the terms of the two previous equations we have

$$\begin{aligned}
& \partial_t \int dv f^0 v_\alpha v_\beta + \partial_\gamma \int dv f^0 v_\alpha v_\beta v_\gamma \\
= \; & n\theta(\partial_\alpha u_\beta + \partial_\beta u_\alpha) - \frac{2}{3}n\theta\partial_\gamma u_\gamma
\end{aligned} \tag{3.20}$$

Now we can use this expression to obtain the first order momentum conservation from equation (3.15):

$$n\partial_t u_\alpha + nu_\beta\partial_\beta u_\alpha = -\partial_\alpha(n\theta) + nF_\alpha + \partial_\beta[\eta(\partial_\beta u_\alpha + \partial_\alpha u_\beta - \frac{2}{3}\partial_\gamma u_\gamma\delta_{\alpha\beta})] \tag{3.21}$$

where $\eta = n\theta\tau$ is the viscosity. This equation is known as the Navier-Stokes Equation.

3.5 Energy conservation

Multiplying the Boltzmann equation with $(v-u)^2$ and integrating we obtain for the energy equation

$$\begin{aligned}
\int dv\partial_t f(v-u)^2 + \int dv\partial_\alpha f v_\alpha(v-u)^2 + F_\alpha \int dv\partial_{v_\alpha} f(v-u)^2 &= \frac{1}{\tau}\int dv(f^0 - f)(v-u)^2 \\
\Leftrightarrow \qquad \partial_t \int dv f(v-u)^2 + \int dv f 2(v_\alpha - u_\alpha)\partial_t u_\alpha & \\
+\partial_\alpha \int dv f v_\alpha(v-u)^2 + \int dv f v_\alpha 2(v_\beta - u_\beta)\partial_\alpha u_\beta &= 0 \\
\Leftrightarrow \; \partial_t \int dv f(v-u)^2 + \partial_\alpha \int dv f v_\alpha(v-u)^2 + 2n\theta\partial_\alpha u_\alpha & \\
-\tau\left[\int \partial_t(v_\alpha - u_\alpha)(v_\beta - u_\beta) \int \partial_\gamma f^0 v_\gamma(v_\alpha - u_\alpha)(v_\beta - u_\beta)\right]\partial_\alpha u_\beta &= 0
\end{aligned}$$

$$\begin{aligned}\Leftrightarrow \qquad & 3\partial_t(n\theta) + \partial_\alpha \int dv f v_\alpha (v-u)^2 + 2n\theta\partial_\alpha u_\alpha \\ & -\tau\partial_\alpha u_\beta \left(\partial_\alpha u_\beta + \partial_\beta u_\alpha - \frac{2}{3}\partial_\gamma u_\gamma \delta_{\alpha\beta}\right) \quad = \quad 0 \end{aligned} \tag{3.22}$$

We now need to approximate the remaining integral using eqn. (3.8).

$$\begin{aligned} & \int dv f v_\alpha (v-u)^2 \\ = \quad & \int dv f^0 v_\alpha (v-u)^2 - \tau[\int dv \partial_t f v_\alpha (v-u)^2 + \int dv \partial_\beta f v_\alpha v_\beta (v-u)^2 \\ & + \int dv \partial_v f v_\alpha (v-u)^2] \\ = \quad & 3n\theta u_\alpha - \tau[\int dv \partial_t f^0 v_\alpha (v-u)^2 + \int dv \partial_\beta f^0 v_\alpha v_\beta (v-u)^2 \\ & -5n\theta F_\alpha] + O(\partial^2). \end{aligned} \tag{3.23}$$

So to zeroth order we can write the energy conservation equation as

$$\partial_t\theta + u_\alpha\partial_\alpha\theta = -\frac{2}{3}\partial_\alpha u_\alpha \theta + O(\partial^2). \tag{3.24}$$

where we have again used the continuity equation (3.14). To obtain the first order equation we now need to evaluate the two integrals in equation (3.23).

$$\begin{aligned} & \int dv \partial_t f^0 v_\alpha (v-u)^2 \\ = \quad & \partial_t \int f^0 v_\alpha (v-u)^2 + \int f^0 v_\alpha 2(v_\gamma - u_\gamma)\partial_t u_\gamma \\ = \quad & \partial_t(3n\theta u_\alpha) + 2n\theta\partial_t u_\alpha \\ = \quad & 3\theta[-\partial_\beta(n u_\alpha u_\beta) - \partial_\alpha(n\theta) + nF_\alpha] + 3nu_\alpha(-u_\beta\partial_\beta\theta - \frac{2}{3}\partial_\beta u_\beta \theta) \\ & +2n\theta[-u_\beta\partial_\beta u_\alpha - \frac{1}{n}\partial_\alpha(n\theta) + F_\alpha] \\ = \quad & \partial_\beta(-3\theta n u_\alpha u_\beta) - 2n\theta\partial_\beta(u_\alpha u_\beta) - 5\theta\partial_\alpha(n\theta) + 5nF_\alpha \end{aligned} \tag{3.25}$$

For the second integral we get

$$\begin{aligned} & \int dv \partial_\beta f^0 v_\alpha v_\beta (v-u)^2 \\ = \quad & \partial_\beta \int f^0 v_\alpha v_\beta (v-u)^2 + \int f^0 v_\alpha v_\beta 2(v_\gamma - u_\gamma)\partial_\beta u_\gamma \\ = \quad & \partial_\alpha \int dv f^0 (v_\alpha - u_\alpha)(v_\beta - u_\beta)(v-u)^2 - \partial_\beta(3n\theta u_\alpha u_\beta) \\ & +2n\theta(u_\alpha\partial_\beta u_\beta + u_\beta\partial_\beta u_\alpha) \\ = \quad & \partial_\alpha(5n\theta^2) + \partial_\beta(3n\theta u_\alpha u_\beta) + 2n\theta\partial_\beta(u_\alpha u_\beta) \end{aligned} \tag{3.26}$$

Combining both integrals from equation (3.23) and the forcing term we obtain

$$\partial_\alpha(5n\theta^2) - 5\theta\partial_\alpha(n\theta) \quad = \quad 5n\theta\partial_\alpha\theta \tag{3.27}$$

So that we get the heat conduction equation

$$\partial_t\theta + u_\alpha\partial_\alpha\theta = -\frac{2}{3}\partial_\alpha u_\alpha\theta + \frac{1}{n}\partial_\alpha(\frac{5n\theta}{3}\partial_\alpha\theta) + \tau\partial_\alpha u_\beta \left(\partial_\alpha u_\beta + \partial_\beta u_\alpha - \frac{2}{3}\partial_\gamma u_\gamma \delta_{\alpha\beta}\right) \tag{3.28}$$

Problems

3.1: Derive (3.9–3.13).

3.2: Show that

$$\begin{aligned}\int f^0 &= n,\\ \int f^0 v_\alpha &= n u_\alpha,\\ \int f^0 v_\alpha v_\beta &= n u_\alpha u_\beta + n\theta\delta_{\alpha\beta},\\ \int f^0 v_\alpha v_\beta v_\gamma &= n\theta(u_\alpha\delta_{\beta\gamma} + u_\beta\delta_{\alpha\gamma} + u_\gamma\delta_{\alpha\beta}) + n u_\alpha u_\beta u_\gamma\end{aligned}$$

for the Maxwell-Boltzmann distribution of equation (2.7).

3.3: Derive the hydrodynamic equations for a BGK Boltzmann equation (3.7) with an equilibrium distribution given by

$$f^0(\mathbf{v}) = n\exp(-\pi\mathbf{v}^2) \tag{3.29}$$

where $n = \int f d\mathbf{v}$.

HINTS: Consider what quantities (if any) are conserved for this evolution equation. What quantities are therefore appropriate fundamental variables? When deriving the hydrodynamic equation(s) for this model make sure that you replace all non-conserved quantities with appropriate approximations (in the same way that we used (3.8) to approximate $\int f\mathbf{v}\mathbf{v}d\mathbf{v}$ in (3.15).

Chapter 4

Lattice Boltzmann

In the last chapter we have seen that the Boltzmann equation describes a dynamics that includes Newtonian hydrodynamics in the long wavelength limit. This may at first sight seem surprising since the Boltzmann equation was derived only in the limit for rare gases and fluids are dense. And if we look closely we see that the transport coefficients, the viscosity ν, the heat conductivity κ and the speeds of sound are all closely related in a way that is not true for all fluids. But the general structure of the conservation laws is so general that they apply for most continuous media.

This opens the way for an alternative way to simulate fluids. Instead of trying to discretize the continuity, Navier-Stokes and heat equations directly a simple discretization of the Boltzmann equation surfices.

Let us briefly review what we required to derive the conservation equations. It is important to notice that we reduced all our calculations to calculations over the equilibrium distribution. And we only required some basic moments of the equilibrium distribution which are given by eqns. (3.9–3.13). So we can conclude that *any* distribution function with these moments would lead to the same macroscopic equations.

4.1 The lattice Boltzmann equation

Let us now write down a simple discretization of the Boltzmann equation with BGK approximation(3.7) for the collision term

$$f(x+v_i, v_i, t+1) - f(x, v_i, t) + F(v_i) = \frac{1}{\tau}[f^0(n, u, \theta) - f(x, v_i, t)] \tag{4.1}$$

Here we have discretized velocity space to a finite number of velocity vectors v_i, space to a lattice where we require that $x + v_i$ is again a lattice position and time only takes on integer values. Since the velocity vectors are fixed now we usually denote $f(x, v_i, t) \equiv f_i(x, t)$ and $F(v_i) \equiv F_i$. The force terms F_i are defined as a generalization of the force of (3.7), *i.e.* $F_i \leftrightarrow F_\alpha \partial_{v_\alpha} f$. In particular we demand that the moments match up:

$$\begin{aligned}
\sum_i F_i &= & \int d\mathbf{v} F_\alpha \partial_{v_\alpha} f & &= 0, && (4.2)\\
\sum_i F_i v_{i\alpha} &= & \int d\mathbf{v} F_\beta \partial_{v_\beta} f v_\alpha & &= -nF_\alpha, && (4.3)\\
\sum_i F_i v_{i\alpha} v_{i\beta} &= & \int d\mathbf{v} F_\gamma \partial_{v_\gamma} f v_\alpha v_\beta & &= -n(F_\alpha u_\beta + u_\alpha F_\beta), && (4.4)\\
\sum_i F_i v_{i\alpha} v_{i\beta} v_{i\gamma} &= & \int d\mathbf{v} F_\delta \partial_{v_\delta} f v_\alpha v_\beta v_\gamma & &= -n[F_\alpha(\theta\delta_{\beta\gamma} + u_\beta u_\gamma) + F_\beta(\theta\delta_{\alpha\gamma} + u_\alpha u_\gamma) & &\\
& & & & +F_\gamma(\theta\delta_{\alpha\beta} + u_\alpha u_\beta)]. && (4.5)
\end{aligned}$$

We now need to show that the hydrodynamic limit of this this discretized version of the Boltzmann are still the well known equations for fluid flow.

Problems

4.1.1: Show that the moments given in (4.2) to (4.5) are indeed what I claim they are.

4.2 Taylor expansion

To determine what the macroscopic equations are that the lattice Boltzmann equation simulates we perform a Taylor expansion of equation (4.1). We obtain to second order

$$\partial_t f_i + v_{i\alpha}\partial_\alpha f_i + \frac{1}{2}\left[\partial_t(\partial_t f_i + v_{i\alpha}\partial_\alpha f_i) + \partial_\beta(\partial_t f_i v_{i\beta} + v_{i\beta}v_{i\alpha}\partial_\alpha f_i)\right] + F_i + O(\partial^3) = \frac{1}{\tau}(f_i^0 - f_i) \tag{4.6}$$

We notice that this is not quite the Boltzmann equation (3.8) that we set out to simulate because there are a large number of additional terms with the second derivative. These terms are discretization error because of the simple discretization scheme we used. However, we will not be deterred by that for the moment and we can still write

$$f_i = f_i^0 - \tau(\partial_t f + v_{i\alpha}\partial_\alpha f_i + F_i) \tag{4.7}$$

to express the f_i in terms of the equilibrium distribution f_i^0. Now expressing everything (exccept the collision term) in terms of the equilibrium distribution we get

$$\begin{aligned}\partial_t f_i^0 - \tau\partial_t(\partial_t f_i + v_{i\alpha}\partial_\alpha f_i + F_i) + v_{i\alpha}\partial_\alpha f_i^0 + \partial_\alpha(\partial_t f_i^0 v_{i\alpha} + v_{i\alpha}v_{i\beta}\partial_\beta f_i^0 + F_i)&\\ +\frac{1}{2}\left[\partial_t(\partial_t f_i + v_{i\alpha}\partial_\alpha f_i) + \partial_\beta(\partial_t f_i^0 v_{i\beta} + v_{i\beta}v_{i\alpha}\partial_\alpha f_i^0)\right] + O(\partial^3) &= \frac{1}{\tau}(f_i^0 - f_i)\end{aligned} \tag{4.8}$$

We note now that, through a lucky coincidence, the discretization errors are of exactly the same form as the higher order terms for the expression of the distribution function in terms of the equilibrium distribution function[1]. We can now write

$$\partial_t f_i^0 + v_{i\alpha}\partial_\alpha f_i + \left(\tau - \frac{1}{2}\right)\left[\partial_t(\partial_t f_i + v_{i\alpha}\partial_\alpha f_i) + \partial_\beta(\partial_t f_i^0 v_{i\beta} + v_{i\beta}v_{i\alpha}\partial_\alpha f_i^0)\right] + O(\partial^3) = \frac{1}{\tau}(f_i^0 - f_i) \tag{4.9}$$

which is exactly the same equation we would have obtained for the Boltzmann equation, exccept that the relaxation time is renormalized to be $\tau - 1/2$. So if we choose an equilibrium distribution with the appropriate moments (3.9–3.13) we will automatically simulate the hydrodynamic equations to the same order that we derived the hydrodynamic limit.

4.3 One dimensional implementation

So in order to implement this we only need to define an equilibrium distribution which fulfills the equivalent definition of (3.9–3.13 which are:

$$\sum_i f_i^0 = n, \tag{4.10}$$

$$\sum_i f_i^0(v_{i\alpha} - u_\alpha) = 0, \tag{4.11}$$

$$\sum_i f_i^0(v_{i\alpha} - u_\alpha)(v_{i\beta} - u_\beta) = n\theta\delta_{\alpha\beta}, \tag{4.12}$$

$$\sum_i f_i^0(v_{i\alpha} - u_\alpha)(v_{i\beta} - u_\beta)(v_{i\gamma} - u_\gamma) = 0, \tag{4.13}$$

$$\sum_i f_i^0(v_{i\alpha} - u_\alpha)(v_{i\beta} - u_\beta)(\mathbf{v_i} - \mathbf{u})^2 = n\theta^2\delta_{\alpha\beta} \tag{4.14}$$

---

[1] You also need to notice that $F_i = O(\partial)$, but that is clear from (4.6).

where the difference between (4.14) and (3.13) is due to the fact that we are considering a one-dimensional model instead of a three dimensional one (see problem 4.1). Since these are 5 equations (in one dimension) we can expect that we will require at least, and probably exactly, a set of 5 velocities v_i and corresponding equililibrium distribution terms f_i^0. If we choose the symmetric velocity set

$$\{v_i\} = \{-2c, -c, 0, c, 2c\} \tag{4.15}$$

we obtain for the equilibrium distribution

$$f_0^0 = \frac{n\left(4\,c^4 + 3\,\theta^2 + 6\,\theta\,u^2 + u^4 - 5\,c^2\left(\theta + u^2\right)\right)}{4\,c^4} \tag{4.16}$$

$$f_1^0 = \frac{n\left(-3\,\theta^2 + 4\,c^3\,u - 6\,\theta\,u^2 - u^4 + 4\,c^2\left(\theta + u^2\right) - c\left(3\,\theta\,u + u^3\right)\right)}{6\,c^4} \tag{4.17}$$

$$f_{-1}^0 = \frac{n\left(-3\,\theta^2 - 4\,c^3\,u - 6\,\theta\,u^2 - u^4 + 4\,c^2\left(\theta + u^2\right) + c\left(3\,\theta\,u + u^3\right)\right)}{6\,c^4} \tag{4.18}$$

$$f_2^0 = \frac{n\left(3\,\theta^2 - 2\,c^3\,u + 6\,\theta\,u^2 + u^4 - c^2\left(\theta + u^2\right) + 2\,c\left(3\,\theta\,u + u^3\right)\right)}{24\,c^4} \tag{4.19}$$

$$f_{-2}^0 = \frac{n\left(3\,\theta^2 + 2\,c^3\,u + 6\,\theta\,u^2 + u^4 - c^2\left(\theta + u^2\right) - 2\,c\left(3\,\theta\,u + u^3\right)\right)}{24\,c^4} \tag{4.20}$$

This just leaves us with the actual implementation of the lattice Boltzmann method defined in (4.1).

One possible implementation, which also employs my GUI, is given in appendix F.

Problems

4.3.1: Show that for a one-dimensional system you obtain (4.14) instead of (3.13).

4.3.2: How would you write a lattice Boltzman code (say one dimensional for simplicity) for an isothermal system? In such a system the energy is not conserved but instead we force the equilibrium temperature to be a constant θ_0.

a) Which macroscopic equations will you need to simulate?

b) What is the form of the Navier-Stokes equation?
Hint: you will want to consider the derivation of (3.21). How does this derivation change for an isothermal system? (*difficult*)

c) What are the required moments of the equilibrium distribution?

d) Hence, what is the minimum number of velocities you are likely to require?

e) Calculate the equilibrium distribution for this model.

f) Implement the model in C. (See Appendix F for an example implementation of the D1Q5 model.)

Please note that this is an enourmous simplification and the math required will be much simplified also.

4.4 Isothermal Lattice Boltzmann

Most lattice Boltzmann simulations are used to only simulate the continuity and Navier-Stokes equations. The temperature is assumed to be constant and the equilibrium distribution will no longer conserve energy; instead it serves as a thermostat. This removes the requirement for the equilibrium equations to fulfill equation (4.14). This moment was only needed to calculate the heat equations. For simplicity let us now consider a one-dimensional LB model. For the full thermo-hydrodynamic model we needed 5 velocities. Now that we have dropped one constraint, you would expect that we need a 4 velocity model to fulfill the remaining 4 constraints. But if there was a way to reduce the number of required velocities further we could save some memory and computation time. If you now consider that you will not be interested in the absolute value of the temperature, you can use the determination of the temperature as an additional degree of freedom. You use the 4 equation to determine f^0_{-1}, f^0_0 f_1, and θ. This will certainly work but we need to remember that we also want θ to be a constant independent of n and u. We will now see that this nearly works.

Using the D1Q3 velocity set $v_i = \{-1, 0, 1\}$ it is easy to see that (4.10) to (4.12) require

$$\begin{aligned} f^0_{-1} &= \frac{1}{2}n(-u+\theta+u^2) && (4.21) \\ f^0_0 &= n(1-\theta-u^2) && (4.22) \\ f^0_1 &= \frac{1}{2}n(u+\theta+u^2) && (4.23) \\ & && (4.24) \end{aligned}$$

Using these solutions for the f^0_i we can calculate θ from (4.13):

$$\theta = \frac{1}{3} - \frac{u^2}{3}. \tag{4.25}$$

We know that the velocity has to be much smaller than the lattice velocity $c = 1$ and θ is nearly constant. Most standard lattice Boltzmann models use these smaller velocity sets. For models in an arbitrary number of dimensions this usually means that the third moment of the equilibrium distribution function is modified to

$$\sum_i f^0_i (v_{i\alpha} - u_\alpha)(v_{i\beta} - u_\beta)(v_{i\gamma} - u_\gamma) = n u_\alpha u_\beta u_\gamma \tag{4.26}$$

and it it assumed that the term in u^3 can be neglected. If it is not negligible this terms will lead to violations of Galilean invariance.

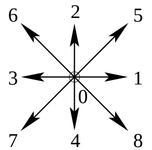
Depending on the dimensionality of the space you want to simulate there are a variety of velocity sets and corresponding equilibrium distributions that are frequently used in the literature.

Equiblibrium distribution is given by:

$$f^0_i = n w_i \left[1 + \frac{3}{c^2}\mathbf{u}.\mathbf{v}_i + \frac{9}{2c^4}(\mathbf{u}.\mathbf{v}_i)^2 - \frac{3}{2c^2}\mathbf{u}.\mathbf{u} \right]. \tag{4.27}$$

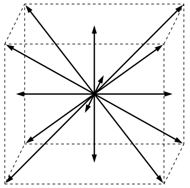
The weights w_i depend on the set of velocites. The values for commonly used models are given below.

For D2Q9 we have



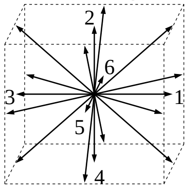
$$w_i = \begin{cases} 4/9 & i = 0 \\ 1/9 & i = 1, 2, 3, 4 \\ 1/36 & i = 5, 6, 7, 8 \end{cases} \tag{4.28}$$

For D3Q15 the weights are:



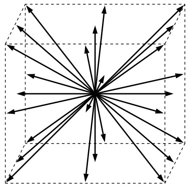
$$w_i = \begin{cases} 2/9 & i = 0 \\ 1/9 & i = 1 - 6 \\ 1/72 & i = 7 - 14 \end{cases} \tag{4.29}$$

For D3Q19 the weights are:



$$w_i = \begin{cases} 1/3 & i = 0 \\ 1/18 & i = 1 - 6 \\ 1/36 & i = 7 - 18 \end{cases} \tag{4.30}$$

For D3Q27 the weights are:



$$w_i = \begin{cases} 8/27 & i = 0 \\ 2/27 & i = 1-6 \\ 1/216 & i = 7-14 \\ 1/54 & i = 15-26 \end{cases} \tag{4.31}$$

Note: The moments of the equilibrium distribution function (4.10) to (4.12) and (4.26) are not sufficient to determine the w_i for large velocity sets. You can see this easily if you consider that the weights to D3Q15 are a subset of D3Q27. You could simply set the weights such that you recover the D3Q15 model from D3Q27 and you would still have the same moments.

These weights are determined by considerations that go beyond the Taylor expansion presented here.

Problems

4.4.1: Show that the constraint (4.26) is not Galilean invariant.

4.4.2: Calculate the weights w_i in (4.27) for the D1Q3 model.

4.5 Non-ideal fluids

We discussed that the Boltzmann equation leads to the Navier-Stokes equation for an ideal gas. It would, of course, be much more useful if we could also simulate non-ideal systems. For such systems the Navier-Stokes equation is given by

$$n\partial_t u_\alpha + n u_\beta \partial_\beta u_\alpha = -\partial_\beta P_{\alpha\beta} + \partial_\beta[\eta(\partial_\beta u_\alpha + \partial_\alpha u_\beta - \frac{2}{3}\partial_\gamma u_\gamma \delta_{\alpha\beta})] \tag{4.32}$$

Comparing this to the ideal Navier-Stokes equation (3.21) we see that we can formally recover the non-ideal behavior if we choose

$$nF_\alpha = -\partial_\beta(P_{\alpha\beta} - n\theta\delta_{\alpha\beta}) \tag{4.33}$$

for any given non-ideal pressure tensor $P_{\alpha\beta}$. These pressure tensors for an iso-thermal system can be derived from the Free energy of the system.

You can implement a forcing term $\mathbf{F}$ by

$$F_i^0 = n w_i \left[\frac{3}{c^2} F_\alpha v_{i\alpha} + \frac{9}{2c^4}(F_\alpha u_\beta + F_\beta u_\alpha) v_{i\alpha} v_{i\beta} - \frac{3}{c^2} F_\alpha u_\alpha\right]. \tag{4.34}$$

Notice: This description of the simulation of non-ideal fluids is quite superficial. In particular we have only considered the bulk terms of the pressure. There are additional terms which relate to the surface tension. Introducing a forcing of the form of (4.33) will lead to a surface tension, but you can not derive what that surface tension is from the analysis presented here.

The terms needed are formally of higher order (like $\nabla^3 n$). But since these gradiens are not small near the interface they are important for determining the shape of the interface and the surface tension. Up to know nobody seems to have a way of consistently deriving the hydrodynamic terms in a way that includes all these terms and work in this area is ongoing.

Problems

4.5.1: Show that (4.34) fulfills the equations (4.2)–(4.4).
Hint: Consider the moments of (4.27).

4.5.2: What are the forcing terms F_i for a D1Q3 model.

4.5.3: Write a D1Q3 model for a Van der Waals gas and show that you can observe phase-separation into a liquid and a gas phase. The Van der Waals equation is given by

$$(V-b)\left(P+\frac{a}{V^2}\right)=NkT \tag{4.35}$$

where a and b are parameters related to the attraction of the particles and the excluded volume respectively. We make the assumption that we can define a local pressure that is related to the local density by $n = N/V$ where N would be the number of particles at the lattice size and V the volume of the lattice size.

a) Show that the Pressure is also given by

$$P=\frac{nkT}{1-nb/N}-\frac{an^2}{N^2} \tag{4.36}$$

b) Calculate the critical temperature Tc, the critical density n_c and the critical pressure P_c.
HINT: Consider that at the critical point $\partial_V P = 0$ and $\partial_v^2 P = 0$.

c) Show that $b/N = 1/(3n_c)$ and $a/N^2 = 9/8\ kT_c/n_c$. Hence you can use n_c and T_c as the variables in your program. (This makes it easier for you to decide whether you would expect phase-separation in a given system). Calculate (4.33) using the approximation $\partial_x P = (P[x+1]-P[x-1])/2$.

d) Calculate the numerical phase diagram (i.e. $n_{liquid}(T/Tc)$ and $n_{gas}(T/Tc)$) using your program. What do you observe about the accuracy of the method?

4.5.4: You will have notice that the approach in the previous problem was not very accurate. The reason for this is that we only used a first order accurate discretization for the pressure for our second order accurate lattice Boltzmann method. What is a second order accurate definition of the first-order derivative of P? Use your new definition in your code and calculate the phase-diagram again.

4.6 Boundaries

In many cases we will want to implement some boundaries in the fluid. It is usually assumed that there is a non-slip boundary condition desired at those boundaries. The simplest way of implementing such a boundary is to draw the boundary and then mark all links that are cut by this boundary. Instead of free streaming on these links the densities are "bounced-back" i.e.

$$f_i(\mathbf{x},t+1)=f_{-i}(\mathbf{x},t) \tag{4.37}$$

where the velocity index $-i$ is defined through $v_{-i} = -v_i$. The effective boundary then lies halfway between the links.

If the boundary is moving, we clearly need to modify this boundary condition. But what will it be modified to? The simplest argument I could come up with is the following: we perform a Galilean transformation of the distribution into the rest frame of the boundary, then we perform the bounce-back operation, and then we transform the flow back into the original frame of reference[2]. Let us define a Galilean transform as

$$f'(v_i)=f_i(v_i)+G(v_i,U) \tag{4.38}$$

We can then write the moving bounce back boundary condition as

$$f(v_i,t+1)=f'_{-i}+G(-v_i,-U)=f_{-i}+G(v_i,U)+G(-v_i,-U) \tag{4.39}$$

---

[2] This derivation follows roughly my argument from a paper on Lees-Edwards boundary conditions[2].

For the moments of the Galilean transformation we obtain

$$\begin{aligned}\sum_i (G(v_i,U)+G(-v_i,-U)) &= 0 && (4.40)\\ \sum_i (G(v_i,U)+G(-v_i,-U))v_{i\alpha} &= 2nU_\alpha && (4.41)\\ \sum_i (G(v_i,U)+G(-v_i,-U))v_{i\alpha}v_{i\beta} &= 0(?) && (4.42)\\ \sum_i (G(v_i,U)+G(-v_i,-U))v_{i\alpha}v_{i\beta}v_{i\gamma} &= ? && (4.43)\end{aligned}$$

The usual choice is

$$G(v_i,U)+G(-v_i,-U)=\frac{6}{c^2}w_i n U_\alpha v_{i\alpha} \qquad (4.44)$$

So for each density crossing the moving boundary the streaming step is replaced with

$$f_i(\mathbf{x},t+1)=f_{-i}(\mathbf{x},t)+\frac{6}{c^2}w_i n U_\alpha v_{i\alpha}. \qquad (4.45)$$

As a practical matter it may often be more efficient to perform a streaming step for all densities and then and additional step that corrects for the solid boundary conditions. Assume that a link from point (x,y) to $(x+v_{ix},y+v_{iy})$ is a boundary link associated with a boundary moving with velocity $\mathbf{U}$. If $\hat{f}_i$ are the densities after the streaming step you will want to perform a swap of the densities and add the appropriate velocity terms:

$$\begin{aligned} f_i(\mathbf{x}+\mathbf{v}_i,t+1) &= \hat{f}_{-i}(\mathbf{x},t)+\frac{6}{c^2}w_i n U_\alpha v_{i\alpha}, && (4.46)\\ f_{-i}(\mathbf{x},t+1) &= \hat{f}_i(\mathbf{x}+\mathbf{v}_i,t)-\frac{6}{c^2}w_i n U_\alpha v_{i\alpha}. && (4.47)\end{aligned}$$

Need to double check this formula for signs.

Chapter 5

Applications of Lattice Boltzmann

5.1 Turbulence

5.2 Liquid-gas systams

5.3 Two-fluid systems

Chapter 6

Non-traditional Lattice Boltzmann methods

6.1 Introduction

While lattice Boltzmann methods were derived in the context of Fluid Mechanics the general algorithm can be used to simulate a variety of other physical phenomena. We will focus here on methods that do not conserve momentum and therefore lead to a different set of equations.

6.2 No conserved parameter

Let us first consider what we obtain if we use a lattice Boltzmann methods that has no conservation laws. A physical realization of such a system may be given by a magnetic system. We will identify the magnetization as

$$\sum_i f_i(x,t) = m(x,t) \tag{6.1}$$

If we now choose an equilibrium distribution with

$$\sum_i f_i^0 = m + \mathcal{M}, \quad \sum_i f_i^0 v_i = j, \quad \sum_i f_i^0 v_i v_i = \psi \mathbf{1}, \tag{6.2}$$

we obtain for the first moment

$$\partial_t m + \nabla j - \left(\tau - \frac{1}{2}\right) \nabla^2 \psi = \mathcal{M} + O(\partial^3). \tag{6.3}$$

6.2.1 Magnetic Systems

The typical evolution equation for a magnetic system (also known as modelA [3]) is given by

$$.\partial_t m = -\mu \tag{6.4}$$

A simple form for μ for a ferromagnetic system is given by

$$\mu(m) = Am + Bm^3 - \kappa \nabla^2 m \tag{6.5}$$

This suggests a choice of $j = 0$, $\mathcal{M} = -(Am + Bm^3)$ and $\psi = km/(\tau - 0.5)$. **This actually works!**

This should now be compare to Ising model simulations. This also provides a great opportunity to introduce the noise terms so that we can compare the results to a well studied system.

6.3 Conserved zeroth moment

If we want to conserve the first moment we need to set $\mathcal{M} = 0$. The resulting equation of motion is

$$\partial_t m + \nabla j = \left(\tau - \frac{1}{2}\right) \nabla^2 \psi + O(\partial^3). \tag{6.6}$$

6.3.1 Diffusion

6.3.2 Electrostatics

6.4 Quantum Mechanics

There are a number of approaches to extend the lattice Boltzmann method to Quantum Mechanical systems. We will present here a simple minded approach aimed at obtaining the Schrödinger equation as a diffusion equation in imaginary time. This is inspired by a paper by Zhang *et al.* [4]. The is given by

$$i\hbar\partial_t \Psi = -\frac{\hbar^2}{2m}\nabla^2\Psi + V\Psi \tag{6.7}$$

Apart from the complex i this looks like a diffusion equation for Ψ with an additional source term $V\Psi$. We can therefore obtain it from the usual lattice Boltzmann equation for a diffusive system of

$$f_i(x + v_i, t + 1) = f_i(x, t) + \frac{1}{\tau}(f_i^0(x, t) - f_i(x, t)) \tag{6.8}$$

where the f_i are now complex densities. To simulate the Schrödinger equation we can use the equation of motion for a non-conserved order parameter equation (6.3), but we have to choose $(\tau - 1/2)$ and $\mathcal{M}$ imaginary. This corresponds to

$$\tau = \frac{1}{2} + i\frac{\hbar}{2m} \tag{6.9}$$

and

$$\mathcal{M} = -i\frac{V}{\hbar} \tag{6.10}$$

This, in principle, should allow us to simulate the Schrödinger equation up to second order accuracy. However, written in this way we may have lost one of the main attractive features of the traditional lattice Boltzmann method: the mass will no longer be automatically conserved.

This will require some careful analysis: there are many ways of obtaining a Schrödinger equation. Instead of changing τ we can also change the second moment of the equilibrium distribution to be imaginary. That may be more elegant.

Mass conservation is usually a strength of lattice Boltzmann, but if we define $\Psi = \sum_i f_i$ we get

$$\Psi^*\Psi = \sum_i \sum_j f_i^* f_j \tag{6.11}$$

but this quadratic form is not conserved in the streaming step. So this simple scheme is not expected to obey .

6.4.1 Hydrodynamic formulation

Alternatively we can write the Schrödinger equation in terms of the amplitude and the phase: $\Psi = Ae^{i\phi}$. The Schrödinger equation becomes

$$i\hbar(\partial_t A + iA\partial_t\phi) = -\frac{\hbar^2}{2m}(\nabla^2 A + 2i\nabla A.\nabla\phi + iA\nabla^2\phi - A(\nabla\phi)^2) + VA \tag{6.12}$$

This complex equation is two real equation. The real part of (6.12) gives

$$-\hbar A\partial_t\phi + \frac{\hbar^2}{2m}(\nabla^2 A - A(\nabla\phi)^2) - VA = 0 \tag{6.13}$$

and the imaginary part of (6.12) gives

$$\hbar\partial_t A + \frac{\hbar^2}{2m}(2\nabla A.\nabla\phi + A\nabla^2\phi) = 0 \tag{6.14}$$

We can express this as

$$\frac{\hbar}{2}\partial_t A^2 + \frac{\hbar^2}{2m}\nabla(A^2\nabla\phi) = 0 \tag{6.15}$$

Now if we define $\rho = A^2$, which is the natural definition of the probability density, and $u = (\hbar/m)\nabla\phi$ this is the familiar continuity equation:

$$\partial_t\rho + \nabla(\rho u) = 0 \tag{6.16}$$

With these identification let us examine the real part of the equation by multiplying with A and taking the gradient ∇_γ (in doing so we loose information about an unimportant constant phase):

$$-\hbar\nabla_\gamma(\rho\partial_t\phi) + \frac{\hbar^2}{2m}\nabla_\gamma(A\nabla^2 A - \rho(\nabla\phi)^2) - \nabla_\gamma(V\rho) = 0 \tag{6.17}$$

Using

$$\nabla^2 A = \nabla(\nabla\sqrt{\rho}) = \nabla\left(\frac{1}{2}\frac{\nabla\rho}{\sqrt{\rho}}\right) = \frac{1}{2}\left(\frac{\nabla^2\rho}{\sqrt{\rho}} - \frac{1}{2}\frac{(\nabla\rho)^2}{(\rho)^{3/2}}\right) \tag{6.18}$$

we get

$$-\hbar(\nabla_\gamma\rho)\partial_t\phi - m\rho\partial_t u_\gamma - \frac{m}{2}\nabla_\gamma(\rho uu) + \frac{\hbar^2}{4m}\nabla_\gamma\left(\nabla^2\rho - \frac{1}{2}\frac{(\nabla\rho)^2}{\rho}\right) - \nabla_\gamma(V\rho) = 0 \tag{6.19}$$

We can express $\partial_t\phi$ in terms of spatial derivatives using (6.13) and (6.18) as

$$-\hbar(\nabla_\gamma\rho)\partial_t\phi = -\frac{\hbar^2}{4m}\left(\frac{\nabla_\gamma\rho\nabla^2\rho}{\rho} - \frac{1}{2}\frac{\nabla_\gamma\rho(\nabla\rho)^2}{\rho^2}\right) + \frac{m}{2}(\nabla_\gamma\rho)u^2 + V\nabla_\gamma\rho \tag{6.20}$$

This simplifies to

$$-\hbar(\nabla_\gamma\rho)\partial_t\phi = -\frac{\hbar^2}{4m}\nabla\left(\frac{\nabla_\gamma\rho\nabla\rho}{\rho} - \frac{1}{2}\frac{(\nabla\rho)^2}{\rho}\delta_{.\gamma}\right) + (\nabla_\gamma\rho)\frac{m}{2}u^2 + V\nabla_\gamma\rho. \tag{6.21}$$

We insert this in (6.17) to get

$$-m\rho\partial_t u_\gamma - m\rho u.\nabla u_\gamma + \frac{\hbar^2}{4m}\nabla\left(\nabla^2\rho\delta_{.\gamma} - \frac{\nabla\rho\nabla_\gamma\rho}{\rho}\right) - \rho\nabla_\gamma V = 0 \tag{6.22}$$

or in the more standard form

$$\rho\partial_t u_\gamma + \rho u.\nabla u_\gamma = \frac{\hbar^2}{4m^2}\nabla\left(\nabla^2\rho\delta_{.\gamma} - \frac{\nabla_\gamma\rho\nabla\rho}{\rho}\right) - \rho\nabla_\gamma\frac{V}{m} \tag{6.23}$$

which corresponds to the Euler equation for a fluid with the pressure tensor

$$P_{\alpha\beta} = -\frac{\hbar^2}{4m^2}\left(\nabla^2\rho\delta_{\alpha\beta} - \frac{\nabla_\alpha\rho\nabla_\beta\rho}{\rho}\right) + const. \tag{6.24}$$

We can therefore simulate the Schrödinger equation using a standard non-ideal lattice Boltzmann simulation for the pressure tensor given by (6.24).

In thermodynamics it is always possible to write $\nabla P + \rho\nabla V/m = \rho\nabla\mu$, where P is the Pressure tensor and μ is the chemical potential. Is there an equivalent quantum-chemical potential here?

First let us note that we can write the pressure as

$$p = \frac{\hbar^2}{2m^2}\left(\sqrt{\rho}\,\nabla^2\sqrt{\rho} - \nabla\sqrt{\rho}\,\nabla\sqrt{\rho}\right) \tag{6.25}$$

We can then define a chemical potential as

$$\mu = \frac{\hbar^2}{2m^2}\frac{\nabla^2\sqrt{\rho}}{\sqrt{\rho}} + \frac{V}{m} \tag{6.26}$$

and we have $\nabla p + \rho\nabla V = \rho\nabla\mu$ as required. From this we can define an effective free energy

$$F = \int dr^3 \left(\frac{\rho V}{m} + \frac{\hbar^2}{4m^2}\nabla\sqrt{\rho}.\nabla\sqrt{\rho}\right). \tag{6.27}$$

A quantum mechanical particle will behave like a classical friction-less fluid with a given by (6.27).

Problems

6.4.1: Show that the familiar results for the energy eigenstates correspond to $u = 0$. Show that we therefore we expect

$$-\frac{\hbar^2}{4m^2}\nabla\left(\nabla^2\rho - \frac{(\nabla\rho)^2}{\rho}\right) - \rho\nabla\frac{V}{m} = 0. \tag{6.28}$$

Show that this relation is obeyed by the energy eigenfunction for a square well potential and the harmonic oscillator.

6.4.2: There is one term in the pressure that can be hard to evaluate numerically, *i.e.*

$$\frac{(\nabla\rho)^2}{\rho}, \tag{6.29}$$

particularly in situation where the probability density vanishes. However, there are several other ways to write this term, *e.g.* $(4\nabla\sqrt{\rho})^2$ or $(\nabla\rho)(\nabla\log\rho)$. Test whether you can find a discretization of the pressure term that allows you to simulate a particle in an infinite potential well.

6.4.3: Show that the two expressions for the pressure of Eq. (6.25) and (6.24) are equivalent.,

6.4.4: Show that $\mu = \delta F/\delta\rho$ where the chemical potential is given by (6.26) and the Free energy is given by (6.27).

Bibliography

[1] "Statistical Mechanics", 2nd edition, K. Huang, Wiley.

[2] "Lees-Edwards boundary conditions for lattice Boltzmann", A.J. Wagner and I. Pagonabarraga, *J. Stat. Phys.* **107**, 521 (2002), [also cond-mat/0103218].

[3] Hohenberg and Halperin

[4] L. Zhang, S. Fent, P. Dong and S. Gao, "Lattice Boltzmann schemes for the nonlinear Schrödinger equation", *Phys. Rev. E* **74**, 036704 (2006).

Appendix A

Einstein convention for vectors and tensors

We can write a vector in terms of its components

$$\mathbf{x} = \begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix} \tag{A.1}$$

The scalar product of two vectors can then be written as

$$c = \mathbf{x}.\mathbf{y} = \begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix} . \begin{pmatrix} y_1 \\ y_2 \\ y_3 \end{pmatrix} = x_1 y_1 + x_2 y_2 + x_3 y_3 = \sum_{i=1}^{3} x_i y_i \tag{A.2}$$

The product of a two dimensional tensor (also known as a matrix) A and a vector v is defined as

$$\mathbf{w} = A.\mathbf{v} = \begin{pmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{pmatrix} . \begin{pmatrix} v_1 \\ v_2 \\ v_3 \end{pmatrix} = \begin{pmatrix} a_{11}x_1 + a_{12}x_2 + a_{13}x_3 \\ a_{21}x_1 + a_{22}x_2 + a_{23}x_3 \\ a_{31}x_1 + a_{32}x_2 + a_{33}x_3 \end{pmatrix} . \tag{A.3}$$

The product of two two-dimensional tensors A and B is given by

$$C = A.B = \begin{pmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{pmatrix} . \begin{pmatrix} b_{11} & b_{12} & b_{13} \\ b_{21} & b_{22} & b_{23} \\ b_{31} & b_{32} & b_{33} \end{pmatrix} = \begin{pmatrix} \sum_i a_{1i}b_{i1} & \sum_i a_{1i}b_{i2} & \sum_i a_{1i}b_{i3} \\ \sum_i a_{2i}b_{i1} & \sum_i a_{2i}b_{i2} & \sum_i a_{2i}b_{i3} \\ \sum_i a_{3i}b_{i1} & \sum_i a_{3i}b_{i2} & \sum_i a_{3i}b_{i3} \end{pmatrix} \tag{A.4}$$

As we progress to higher dimensional tensors it becomes more and more cumbersom to write down all these components. There is an easier way of writing this in terms of components. For this we use Greek letters to enumerate the spatial dimensions. For (A.1) we write

$$x_\alpha \tag{A.5}$$

where the free index α indicates that this is a vector expression. For (A.2) we get

$$c = \sum_{\alpha=1}^{3} x_\alpha y_\alpha = x_\alpha y_\alpha \tag{A.6}$$

where we used the Einstein convention which tells us that a sum over repeated indices is implied. Note that this expression is a scalear expression, *i.e.* there are no free indices. For the product of a matrix and a vector (A.3) we again obtain a vector equation

$$w_\alpha = \sum_{\beta=0}^{3} a_{\alpha\beta} v_\beta = a_{\alpha\beta} v_\beta \tag{A.7}$$

The product of two matrices is a matrix equation (*i.e.* it has two free indices) and (A.4) can be written as

$$c_{\alpha\beta} = \sum_\gamma a_{\alpha\gamma} b_{\gamma\beta} = a_{\alpha\gamma} b_{\gamma\beta} \tag{A.8}$$

One important matrix is the identity matrix. This is represeted by the Kronecker delta in tensor notation:

$$\delta_{\alpha\beta} = \begin{cases} 1 & \text{if } \alpha = \beta \\ 0 & \text{otherwise} \end{cases} \tag{A.9}$$

This often is appears in tensor equations. Consider for instance

$$a_{\alpha\gamma} b_{\epsilon\beta} \delta_{\nu\gamma} \delta_{\epsilon\nu} = a_{\alpha\nu} b_{\nu\beta}. \tag{A.10}$$

You may want to point out that the vector notation is at least as compact as the tensor notation. That is correct, but it becomes difficult to manipulate expressions like

$$\mathrm{tr}\left((\nabla\mathbf{u} + (\nabla\mathbf{u})^T + \nabla.\mathbf{u}\mathbf{1}).\sigma\right). \tag{A.11}$$

It is much easier to manipulate the equivalent

$$(\partial_\alpha u_\beta + \partial_\beta u_\alpha + \partial_\gamma u_\gamma \delta_{\alpha\beta})\sigma_{\alpha\beta} \tag{A.12}$$

if one is familiar with the tensor notation. And for tensors of more than two dimensions vector notation is no longer used!

Problems

A.1: Express $\mathbf{v}.A.\mathbf{w}$, where $\mathbf{v}$ and $\mathbf{w}$ are vectors and A is a two dimensional tensor, in tensor notation.

A.2: Are the following a scalar, vector, two-dimensional tensor or higher dimensional tensor expressions?

a) $a_{\alpha\beta}\, b_{\beta\gamma}\, c_{\gamma\delta}$

b) $a_{\alpha\beta}\, b_\alpha\, c_\gamma$

c) $a_{\alpha\beta\gamma}\, b_{\alpha\delta}\, c_{\gamma\delta}$

d) $a_\alpha b_{\gamma\delta\epsilon} c_\nu e_\delta$

A.3: Simplify $\partial_\gamma u_\alpha \delta_{\beta\gamma} + \partial_\gamma u_\beta \delta_{\alpha\gamma}$.

Appendix B

Functional Derivatives

Recall that the derivative of a function $f(x)$ is defined as

$$\frac{df(x)}{dx} = \lim_{dx \to 0} \frac{f(x+dx) - f(x)}{dx} \tag{B.1}$$

For a functional, *i.e.* a function of a function we can define a functional derivative in a similar way:

$$\frac{\delta \Psi[f(x)]}{\delta f(y)} = \lim_{\delta f(y) \to 0} \frac{\Psi[f(x) + \delta f(y)] - \Psi[f(x)]}{\delta f(y)} \tag{B.2}$$

where the variation $\delta f(y)$ is only different from zero at $x = y$. These functionals are usually integrals and in the simplest case where Ψ is only a functional of f and not its derivatives we have

$$\Psi[f] = \int_a^b \psi(f(x))dx \tag{B.3}$$

The functional derivative is then simply given by

$$\frac{\delta \Psi[f(x)]}{\delta f(y)} = \int_a^b \psi'(f(x)) \frac{\delta f(x)}{\delta f(y)} dx = \int_a^b \psi'(f(x)) \delta(x-y) dx = \psi'(y) \tag{B.4}$$

where ψ' is the ordinary derivative of ψ. The functional derivative of functionals containing spatial derivatives is a little more complicated, and we need to perform a partial integration. For $\Psi[f] = \int_a^b \nabla f(x) . \nabla f(x)$ we obtain

$$\frac{\delta \Psi[f(x)]}{\delta f(y)} = \int_a^b 2\nabla f(x) \frac{\delta \nabla f(x)}{\delta f(y)} dx = -\int_a^b \nabla^2 f(x) \delta(x-y) dx = -\nabla^2 f(y) \tag{B.5}$$

where we have made use of the fact that the boundary term vanishes because of the δ-function. Other derivative terms have to be treated similarly.

Problems

B.1: What is the functional derivative of

a) $\Psi[f] = \int_a^b (f^2(x) - f^5(x) + f^2(x) \nabla^2 f(x)) dx$?

Appendix C

Lagrangian multipliers

If you want to minimize a functional while you have other constraints you can often use Lagrangian multipliers to remove the contraints.

Assume you want to minimize the functional $\Psi[\phi(x)] = \int_a^b \psi(\phi(x))dx$ with the constraint $\int_a^b \phi(x) = c$. We then write a new functional $\hat{\Psi}$ as

$$\hat{\Psi} = \Psi + \lambda(\int_a^b \phi(x)dx - c) \tag{C.1}$$

where λ is known as a Lagrange multiplier. Note that $\Psi = \hat{\Psi}$ if the constraint is fulfilled. Now, however, we relax the constraint and minimize $\hat{\Psi}$ by calculating

$$0 = \frac{\delta\Psi(\phi)}{\delta\phi(y)} = \psi'(\phi(y)) + \lambda \tag{C.2}$$

which we can invert to find $\phi(y, \lambda)$. These solutions will, in general, not fulfill our constraint. However, we can introduce a "cost" of not obeying the constraint by setting different values for λ. In particular we can determine the value of λ where this "cost" is just right so that $\phi(x, \lambda)$ does fulfill the constraing. We do this by inserting $\phi(x, \lambda)$ into the constraint

$$\int_a^b \phi(y, \lambda) = c. \tag{C.3}$$

We can solve this to determine the Lagrangian multiplier λ for which our solution obeys the constraint.

Problems

C.1: Minimize

$$\int_{-a}^{a} -\frac{1}{2}f^2(x) + \frac{1}{4}f^4(x) + \frac{1}{2}\left(\frac{df(x)}{dx}\right)^2 \tag{C.4}$$

with the constraint $\int_{-a}^{a} f(x) = 0$.

SOLUTION: $f(x) = \tanh(x/\sqrt{2})$.

Appendix D

Evaluating Gaussian integrals

First let us review some standard one dimensional Gaussian integrals that we will be using:

$$\begin{aligned}\int_{-\infty}^{\infty} e^{-ax^2} dx &= \sqrt{\frac{\pi}{a}} \\ \int_{-\infty}^{\infty} x^2 e^{-ax^2} dx &= \sqrt{\pi}\frac{1}{2}a^{-3/2} \\ \int_{-\infty}^{\infty} x^4 e^{-ax^2} dx &= \sqrt{\pi}\frac{3}{4}a^{-5/2}\end{aligned}$$

Now let us look at some of the more relevant three dimensional integrals that we can derive using the above equations. Integrating the Gaussian we get

$$\begin{aligned}\int_{-\infty}^{\infty}\int_{-\infty}^{\infty}\int_{-\infty}^{\infty} dx\, dy\, dz\; e^{-a(x^2+y^2+z^2)} &= \int_{-\infty}^{\infty} e^{-ax^2} dx \int_{-\infty}^{\infty} e^{-ay^2} dy \int_{-\infty}^{\infty} e^{-az^2} dz \\ &= \left(\frac{\pi}{a}\right)^{3/2}. \qquad \text{(D.1)}\end{aligned}$$

To calculate the temperature we need

$$\begin{aligned}&\int_{-\infty}^{\infty}\int_{-\infty}^{\infty}\int_{-\infty}^{\infty} dx\, dy\, dz\; (x^2+y^2+z^2) e^{-a(x^2+y^2+z^2)} \\ = &\int_{-\infty}^{\infty} x^2 e^{-ax^2} dx \int_{-\infty}^{\infty} e^{-ay^2} dy \int_{-\infty}^{\infty} e^{-az^2} dz \\ &+\int_{-\infty}^{\infty} e^{-ax^2} dx \int_{-\infty}^{\infty} y^2 e^{-ay^2} dy \int_{-\infty}^{\infty} e^{-az^2} dz \\ &+\int_{-\infty}^{\infty} e^{-ax^2} dx \int_{-\infty}^{\infty} e^{-ay^2} dy \int_{-\infty}^{\infty} z^2 e^{-az^2} dz \\ = &\; 3\frac{1}{2a}\left(\frac{\pi}{a}\right)^{3/2}. \qquad \text{(D.2)}\end{aligned}$$

To calculate the heat equation we need to evaluate terms of the form

$$\begin{aligned}&\int_{-\infty}^{\infty}\int_{-\infty}^{\infty}\int_{-\infty}^{\infty} dx\, dy\, dz\; x^2(x^2+y^2+z^2) e^{-a(x^2+y^2+z^2)} \\ = &\int_{-\infty}^{\infty} x^4 e^{-ax^2} dx \int_{-\infty}^{\infty} e^{-ay^2} dy \int_{-\infty}^{\infty} e^{-az^2} dz \\ &+\int_{-\infty}^{\infty} x^2 e^{-ax^2} dx \int_{-\infty}^{\infty} y^2 e^{-ay^2} dy \int_{-\infty}^{\infty} e^{-az^2} dz \\ &+\int_{-\infty}^{\infty} x^2 e^{-ax^2} dx \int_{-\infty}^{\infty} e^{-ay^2} dy \int_{-\infty}^{\infty} z^2 e^{-az^2} dz\end{aligned}$$

$$
\begin{aligned}
&= 3\frac{1}{(2a)^2}\left(\frac{\pi}{a}\right)^{3/2} + \frac{1}{(2a)^2}\left(\frac{\pi}{a}\right)^{3/2} + \frac{1}{(2a)^2}\left(\frac{\pi}{a}\right)^{3/2} \\
&= 5\frac{1}{(2a)^2}\left(\frac{\pi}{a}\right)^{3/2} \qquad \text{(D.3)}
\end{aligned}
$$

Note that in the Gaussian integrals we perform $a = 1/(2\theta)$.

Problems

D.1: Show what the value of the integrals (D.1), (D.2), and (D.3) are in a D dimensional space.

Appendix E

Galilean transformations

Since the velocity set is fixed we clearly can not perform a true Galilean transformation which would simply require $f'(v_i) = f(v_i + U)$. What we can require, however, is that the transformed moments agree with the Galilean transformation, *i.e.* $\sum f_i' v_i^N = \sum f_i(v_i + U)^N$ for an appropriate number of powers N. This gives us

$$\sum f_i' = \sum f_i \tag{E.1}$$

$$\sum f_i' v_{i\alpha} = \sum f_i v_{i\alpha} + nU_\alpha \tag{E.2}$$

$$\sum f_i' v_{i\alpha} v_{i\beta} = \sum f_i v_{i\alpha} v_{i\beta} + n(u_\alpha U_\beta + U_\alpha u_\beta + U_\alpha U_\beta) \tag{E.3}$$

$$\begin{aligned}\sum f_i' v_{i\alpha} v_{i\beta} v_{i\gamma} = {} & \sum f_i v_{i\alpha} v_{i\beta} v_{i\gamma} + n[(u_\alpha u_\beta + \theta\delta_{\alpha\beta})U_\gamma + (u_\alpha u_\gamma + \theta\delta_{\alpha\gamma})U_\beta + (u_\beta u_\gamma + \theta\delta_{\beta\gamma})U_\alpha \\ & + U_\alpha U_\beta U_\gamma]\end{aligned} \tag{E.4}$$

$$\begin{aligned}\sum f_i' v_{i\alpha} v_{i\beta} v_{i\gamma} = {} & \sum f_i v_{i\alpha} v_{i\beta} v_{i\gamma} + n[(u_\alpha u_\beta + \theta\delta_{\alpha\beta})U_\gamma + (u_\alpha u_\gamma + \theta\delta_{\alpha\gamma})U_\beta + (u_\beta u_\gamma + \theta\delta_{\beta\gamma})U_\alpha \\ & + U_\alpha U_\beta U_\gamma]\end{aligned} \tag{E.5}$$

Appendix F

D1Q5 Lattice Boltzmann code

```
/* An implementation of the D1Q5 model for non-ideal systems */

#include <stdlib.h>
#include <math.h>
#include "mygraph.h"
#define xdim 100

double f0[xdim],f1[xdim],f2[xdim],f3[xdim],f4[xdim],n[xdim];
double n0=1, T0=0.33333333, Amp=0.01, omega=1;
double ug[xdim],pg[xdim],tg[xdim];
int ugreq=0,pgreq=0,tgreq=0;
int next=0,pause=1,done=0,Repeat=1,iterations;

void init(){
  int i;
  iterations=0;
  for (i=0;i<xdim;i++){
    n[i]=n0+Amp*sin(2*M_PI*i/xdim);
    f0[i]=n[i]/4 *(4-5*T0+5*T0*T0);
    f1[i]=n[i]/6 *(  4*T0-5*T0*T0);
    f2[i]=n[i]/6 *(  4*T0-5*T0*T0);
    f3[i]=n[i]/24*(   -T0+5*T0*T0);
    f4[i]=n[i]/24*(   -T0+5*T0*T0);
  }
}

void iteration(){
  double u,T,tmp,tmp2;
  int i;

  iterations++;
  for (i=0;i<xdim;i++){
    n[i]=f0[i]+f1[i]+f2[i]+f3[i]+f4[i];
    u=f1[i]-f2[i]+2*f3[i]-2*f4[i];
    T=(f1[i]+f2[i]+4*f3[i]+4*f4[i])/n[i]-u*u;
    f0[i]+=omega*(n[i]/4.*(4-5*T+5*T*T-(5-6*T)*u*u+u*u*u*u)-f0[i]);
    f1[i]+=omega*(n[i]/6.*(4*T-5*T*T+(4-3*T)*u+(4-6*T)*u*u-u*u*u-u*u*u*u)-f1[i]);
    f2[i]+=omega*(n[i]/6.*(4*T-5*T*T-(4-3*T)*u+(4-6*T)*u*u+u*u*u-u*u*u*u)-f2[i]);
    f3[i]+=omega*(n[i]/24.*(-T+5*T*T-(2-6*T)*u-(1-6*T)*u*u+2*u*u*u+u*u*u*u)-f3[i]);
    f4[i]+=omega*(n[i]/24.*(-T+5*T*T+(2-6*T)*u-(1-6*T)*u*u-2*u*u*u+u*u*u*u)-f4[i]);
```

```
  }
  tmp=f1[0];
  memmove(&f1[0],&f1[1],(xdim-1)*sizeof(double));
  f1[xdim-1]=tmp;
  tmp=f2[xdim-1];
  memmove(&f2[1],&f2[0],(xdim-1)*sizeof(double));
  f2[0]=tmp;
  tmp=f3[0];
  tmp2=f3[1];
  memmove(&f3[0],&f3[2],(xdim-2)*sizeof(double));
  f3[xdim-1]=tmp2;
  f3[xdim-2]=tmp;
  tmp=f4[xdim-1];
  tmp2=f4[xdim-2];
  memmove(&f4[2],&f4[0],(xdim-2)*sizeof(double));
  f4[0]=tmp2;
  f4[1]=tmp;
}

void GUI(){
  static int xdimi=xdim;

  DefineGraphN_R("n",&n[0],&xdimi,NULL);
  DefineGraphN_R("u",&ug[0],&xdimi,&ugreq);
  DefineGraphN_R("p",&pg[0],&xdimi,&pgreq);
  DefineGraphN_R("T",&tg[0],&xdimi,&tgreq);
  StartMenu("D1Q3",1);
  DefineInt("iterations",&iterations);
  DefineDouble("omega",&omega);
  DefineDouble("Amp",&Amp);
  DefineDouble("n0",&n0);
  DefineDouble("T0",&T0);
  DefineFunction("init",&init);
  DefineGraph(curve2d_,"Graphs");
  DefineInt("Repeat",&Repeat);
  DefineBool("next",&next);
  DefineBool("pause",&pause);
  DefineBool("done",&done);
  EndMenu();
}

void GetData(){
  int i;

  if (ugreq||tgreq) {
    for (i=0;i<xdim;i++) ug[i]=(f1[i]-f2[i]+2*f3[i]-2*f4[i])/n[i];
    ugreq=0;
  }
  if (pgreq) {
    for (i=0;i<xdim;i++) pg[i]=f1[i]+f2[i]+4*f3[i]+4*f4[i];
    pgreq=0;
  }
  if (tgreq) {
    for (i=0;i<xdim;i++) tg[i]=(f1[i]+f2[i]+4*f3[i]+4*f4[i])/n[i];
    tgreq=0;
```

```
  }
}

int main(int argc, char *argv[]){
  int newdata=1;
  int i;

  init();
  GUI();

  while (done==0){
    Events(newdata);
    GetData();
    DrawGraphs();
    if (next|| !pause){
      newdata=1;
      next=0;
      for (i=0;i<Repeat;i++){
        iteration();
      }
    }
    else sleep(1);
  }

  return 0;
}
```

Index