COMMUNICATIONS IN COMPUTATIONAL PHYSICS
Vol. **1**, No. 4, pp. 616-655

Commun. Comput. Phys.
August 2006

Elements of the Lattice Boltzmann Method I: Linear Advection Equation

Iliya V. Karlin[1,*], Santosh Ansumali[2], Christos E. Frouzakis[1] and Shyam Sunder Chikatamarla[1]

[1] *Institute of Energy Technology, ETH Zurich, 8092 Zurich, Switzerland.*
[2] *School of Chemical and Biomedical Engineering, Nanyang Technological University, Singapore 639798.*

Received 25 November 2005; Accepted (in revised version) 14 March 2006

Communicated by Sauro Succi

Abstract. This paper opens a series of papers aimed at finalizing the development of the lattice Boltzmann method for complex hydrodynamic systems. The lattice Boltzmann method is introduced at the elementary level of the linear advection equation. Details are provided on lifting the target macroscopic equations to a kinetic equation, and, after that, to the fully discrete lattice Boltzmann scheme. The over-relaxation method is put forward as a cornerstone of the second-order temporal discretization, and its enhancement with the use of the entropy estimate is explained in detail. A new asymptotic expansion of the entropy estimate is derived, and implemented in the sample code. It is shown that the lattice Boltzmann method provides a computationally efficient way of numerically solving the advection equation with a controlled amount of numerical dissipation, while retaining positivity.

Key words: Lattice Boltzmann method; implicit schemes; advection; entropy; invariant manifold; kinetic theory.

1 Introduction

One of the most important achievements in the physical sciences is that many phenomena become understandable if one succeeds to recognize a particles' picture behind it. Particles (point masses) are the gift of Newton's classical mechanics. The corpuscular picture of light enabled Planck and Einstein to pioneer quantum mechanics. Some particle-based

[*] Correspondence to: Iliya V. Karlin, Institute of Energy Technology, ETH Zurich, 8092 Zurich, Switzerland. Email: `karlin@lav.mavt.ethz.ch`

programs established links between different fields of science, and explained phenomenology on a simpler (particle-based) level. Such are the achievements of Gibbs, Boltzmann, Hilbert and Enskog who linked thermodynamics and fluid dynamics to the particles' dynamics. Some others programs still require further effort as, for example, eddy viscosity models of fluid turbulence dating back to Prandtl.

Computational physics is a large 'laboratory' where particles are used as an ingredient for the creation of numerical methods. Computational physics is led by efficiency and accuracy of computations; thus, in many situations 'good' computational particles are only remote relatives of the 'true' physical particles. These notes are about the lattice Boltzmann method (LBM) for solving partial differential equations. LBM evolved from a particles' picture of lattice gas automata, something which only barely resembles physical particles. For the history of the lattice Boltzmann the reader is directed to the book [1]. We believe that many of the readers of this paper have either heard of the keyword 'lattice Boltzmann method' or have their own experience about using a lattice Boltzmann scheme in some problem. This paper intends to shed some light on the lattice Boltzmann method for a newcomer.

The best way to explain a method or idea is to address a small understandable problem, and to consider in full detail how the method works to solve it. Certainly, one should not forget that once things are cleaned up on the level of a toy problem, different new ideas might be (and usually are) required when stepping into a 'real' problem. Yet, intuition gained from solving small problems can be of substantial help.

We have chosen the simplest possible equation - the linear advection equation in one spatial direction. While this is indeed a simple equation, it is often used to discuss numerical methods for solving partial differential equations. The problem of creating accurate and efficient numerical schemes for solving the linear advection equation is not simple at all. In this paper we systematically introduce the lattice Boltzmann scheme for the advection equation.

The linear advection equation is used as a showcase in order to highlight some elements of the lattice Boltzmann schemes, especially those which contribute to certain, probably unique, features of these schemes. The presentation therefore differs significantly from other expositions of LBM. After introducing the lifting of the advection equation to a kinetic system with three velocities (Section 2), and explaining how to tune the equilibrium in order to recover the advection equation from it (Section 3), we proceed directly to the heart of the lattice Boltzmann schemes, the over-relaxation mechanism of temporal discretization (Section 4). We discuss this in detail, and explain how entropy enters the game in order to control positivity of the particle's populations (Section 6). We derive a new asymptotic expansion of the entropy estimate. A sample code is provided in order to illustrate how a lattice Boltzmann code looks in practice, and how to incorporate the entropy estimate into it.

In the first place, we tried to make a 'demo-tour' over the lattice Boltzmann terrain without making it too technical. We also tried to make it possible for a reader to compile a concise glossary of notions used in kinetic theory; all of these notions are illustrated with simple examples of usage. Eventually, we believe that most of the lattice Boltzmann method as a numerical technique is covered in this paper in such a way as to make it possible to read and understand the current technical literature on the subject. A 'gain-oriented' reader may ask: What do we gain from the lattice Boltzmann method for solving the linear advection equation? Eventually, the lattice Boltzmann approach delivers a simple second-order scheme for these equations, without adjustable parameters and with a controlled amount of diffusion. The latter is maybe most significant: The amount of diffusion is not dictated by the grid, as in many other methods, but by the hydrodynamic limit of the kinetic model. We did not attempt any extensive comparison of the lattice Boltzmann scheme with other methods, though some pictures at the end of the paper present also the Essentially Non-Oscillatory (ENO) scheme. Generally speaking, LBM provides a reasonable alternative to these more sophisticated higher-order schemes.

Finally, we made every effort to make the presentation self-contained; thus, references are kept at a minimal level. For a further reading on the lattice Boltzmann method, we direct the reader to the papers [2–5] and reviews [6–8]. The development of the entropic lattice Boltzmann method can be found in [9–16].

2 Free flight and discretization

2.1 Free flight

A point mass m moves with a constant velocity v along a line. This motion is described by Newton's equation:

$$m\ddot{x} = 0, \tag{2.1}$$

or, equivalently, by Hamilton's equations:

$$\dot{x} = p/m, \quad \dot{p} = 0. \tag{2.2}$$

Given the initial position at time $t = 0$, at any time t we have $x(t) = x(0) + vt$. A cloud of noninteracting particles is characterized by its density $\rho(x,t)$. By the physical sense of this quantity, ρ is a nonnegative function of space x for any time t. The advection of the (passive scalar) density field $\rho(x,t)$ by a flow of constant velocity v is described by the linear advection equation:

$$\partial_t \rho(x,t) + v\partial_x \rho(x,t) = 0. \tag{2.3}$$

Equation (2.3) (also, advection equation, free flight equation, ballistic equation...) is the simplest instance of the *Liouville equation.* For Hamilton's equations, $\dot{q}_i = \partial H/\partial p_i$, $\dot{p}_i = -\partial H/\partial q_i$, $i = 1,\dots,N$, where H is a Hamiltonian (energy), the Liouville equation for the phase-space density $\rho(q_1,\dots,q_N,p_1,\dots,p_N,t)$ reads:

$$\partial_t \rho + \sum_{i=1}^{N} [(\partial H/\partial p_i)(\partial \rho/\partial q_i) - (\partial H/\partial q_i)(\partial \rho/\partial p_i)] = 0.$$

Liouville introduced this equation while solving a formal problem of how to put a linear partial differential equation into correspondence with a set of ordinary differential equations of first order (not necessarily Hamilton's equations). A century after that, Liouville's equation became a cornerstone of statistical mechanics..

Another very useful way to write (2.3) is the following:

$$\partial_t \rho(x,t) + \partial_x j(x,t) = 0, \tag{2.4}$$

$$j(x,t) = v\rho(x,t). \tag{2.5}$$

Equation (2.4) is a *conservation law* and was formulated with the help of an intermediate object - *the flux j*.

Equation (2.4) tells us that the rate of change of the density of particles in an infinitesimal volume equals the divergence of the flux of particles. That is, particles are neither created nor destroyed, but the change of their amount in an infinitesimal volume is simply due to the fact that particles flow in and out of this volume.

The conservation law cannot be solved because it contains two unknowns, ρ and j. The situation is improved by the *constitutive equation* (2.5), which defines how the flux depends on density. As the name suggests, in the course of time, the density and the flux vary, depending on the initial conditions, but, whatever this variation is, the relation between the flux and the density obeys (2.5).

Given the density at the initial time $t = 0$, $\rho(x,0) = \rho_0(x)$, the analytical solution at any time is embarrassingly simple:

$$\text{if } \rho(x,0) = \rho_0(x), \text{ then } \rho(x,t) = \rho_0(x - vt).$$

However, it is much more difficult to solve the advection equation numerically: *'There is no such thing as a perfect advection scheme – only differing degrees of badness'* [17]. Any discretization scheme of space and time introduces numerical diffusion that tends to smooth the discontinuities, and undesirable dispersion behavior creating waves that move with velocities that depends on their wavenumber. The literature on the numerical solution of the advection equation is vast, and different approaches have been devised to cope with these difficulties.

In this paper, we illustrate how 'another particles' picture is used to solve numerically the advection equation in a way that minimizes the effect of discretization at the lowest computational cost.

2.2 Kinetic representation of free flight

For simplicity we consider first the one-dimensional case. Let us consider fictitious particles which we will call 'quarks'. This will have some (linguistic) analogy to the composition of elementary particles out of quarks. Quarks are not observed separately, so there will be no point to think of them as individual particles. In the one-dimensional case, there are three quarks, right moving (+), left moving (−), and stationary (0), with velocities c, 0 and

$-c$, respectively. Quarks are characterized by their (nonnegative) populations, $f_-(x,t)$, $f_0(x,t)$, and $f_+(x,t)$. The density of the real particles is the sum of the populations of quarks,

$$\rho(x,t) = f_-(x,t) + f_0(x,t) + f_+(x,t). \tag{2.6}$$

The aim is to define the dynamics of the quarks in such a way that in the long-time large-scale limit we obtain the advection equation as closely as possible. The dynamics of the quarks populations is a combination of a free flight and a relaxation to equilibrium. The simplest set of *kinetic equations* can be written as:

$$\partial_t f_-(x,t) - c\partial_x f_-(x,t) = -\frac{1}{\tau}\left(f_-(x,t) - f_-^{\mathrm{eq}}(f(x,t))\right), \tag{2.7a}$$

$$\partial_t f_0(x,t) = -\frac{1}{\tau}\left(f_0(x,t) - f_0^{\mathrm{eq}}(f(x,t))\right), \tag{2.7b}$$

$$\partial_t f_+(x,t) + c\partial_x f_+(x,t) = -\frac{1}{\tau}\left(f_+(x,t) - f_+^{\mathrm{eq}}(f(x,t))\right). \tag{2.7c}$$

The free flight operators on the left-hand side are the same as in equation (2.3), while the right-hand side describes relaxation to the *equilibrium* $f^{\mathrm{eq}}_{-,0,+}$. These equilibrium populations are a major part of the construction, and will be specified below. The parameter $\tau > 0$ has the dimension of time, and is the relaxation time to equilibrium. Thus, the quark dynamics (2.7) is irreversible in time; the relaxation terms break the time symmetry.

For brevity, we can use a short-hand notation for the three-component vector of the populations,

$$f(x,t) = [f_-(x,t), f_0(x,t), f_+(x,t)]. \tag{2.8}$$

The dependence of the equilibrium functions $f^{\mathrm{eq}}_{-,0,+}$ on space and time is *mediated* by the dependence on the density ρ. In other words, we assume (first) that the equilibrium is a function of ρ:

$$f_-^{\mathrm{eq}}(\rho),\ f_0^{\mathrm{eq}}(\rho),\ f_+^{\mathrm{eq}}(\rho). \tag{2.9}$$

To this end, we have not specified the dependence of the equilibrium on the density, and tuning this dependence in a proper way will be the goal of the construction. However, one crucial property of the equilibrium must be highlighted from the very beginning: whatever the equilibrium function we choose, it must satisfy the *consistency condition*:

$$f_-^{\mathrm{eq}}(\rho) + f_0^{\mathrm{eq}}(\rho) + f_+^{\mathrm{eq}}(\rho) = \rho. \tag{2.10}$$

This should hold for any value of ρ: if we consider the equilibrium at any given ρ and compute the corresponding density, we must get back the same density. Symbolically,

$$\rho[f^{\mathrm{eq}}(\rho)] = \rho, \tag{2.11}$$

where $\rho[f]$ is the operation (2.6).

For now, assume that the equilibrium functions (2.9) are given. Then (second), in order to evaluate the right-hand sides of the kinetic equations (2.7) for the population vector

$f(x,t)$, one needs to compute the density $\rho(x,t)$ corresponding to this vector (according to (2.6)), and substitute the result in the equation for the equilibria (2.9). Symbolically, we write this sequence of operations as follows:

$$f(x,t) \to \rho(f(x,t)) \to f^{\mathrm{eq}}(\rho(f(x,t))) = f^{\mathrm{eq}}(f(x,t)). \tag{2.12}$$

With the relation (2.12), the set of equations (2.7) is the simplest example of the Bhatnagar-Gross-Krook kinetic (BGK) model. The meaning of the right-hand side of (2.7) (*the BGK collision integral*) is the following: the "current" populations $f(x,t)$ "see" the corresponding equilibrium $f^{\mathrm{eq}}(f(x,t))$, and tend to it according to the right-hand side of (2.7) with a rate proportional to the deviation of $f(x,t)$ from $f^{\mathrm{eq}}(f(x,t))$.

The equilibria at various densities are the *stationary points* of the relaxation term: the right-hand sides of equation (2.7) become equal to zero if we substitute

$$f_{-,0,+} = f^{\mathrm{eq}}_{-,0,+}(\rho(f)),$$

and use the consistency condition (2.10).

We have assumed that equilibrium depends on the *locally conserved* quantity (the density ρ). The term 'local' reflects the fact that in equations (2.7) and (2.12), f^{eq} depends only on the values of density at position x and time t, but not on its derivatives. This is one of the most basic features of models like (2.7). In order to check that density is indeed conserved by the dynamics, we add the three equations (2.7) to obtain an equation for the density

$$\partial_t \rho + \partial_x j = \mathrm{RHS}, \tag{2.13}$$

where we introduced the *momentum flux* of the quarks j, $j(x,t) = -cf_-(x,t) + cf_+(x,t)$. The right-hand side, RHS, is the sum of all the terms on the right-hand side of equations (2.7) and describes how density is changing by the BGK "collisions"

$$\begin{aligned}\mathrm{RHS} =& -\frac{1}{\tau}\left\{\big(f_-(x,t) - f_-^{\mathrm{eq}}(f(x,t))\big) + (f_0(x,t) - f_0^{\mathrm{eq}}(f(x,t))) + \big(f_+(x,t) - f_+^{\mathrm{eq}}(f(x,t))\big)\right\}\\ =& -\frac{1}{\tau}\left\{f_-(x,t) + f_0(x,t) + f_+(x,t) - (f_-^{\mathrm{eq}}(f(x,t)) + f_0^{\mathrm{eq}}(f(x,t)) + f_+^{\mathrm{eq}}(f(x,t)))\right\}\\ =& -\frac{1}{\tau}\{\rho(x,t) - \rho(x,t)\} = 0.\end{aligned} \tag{2.14}$$

The crucial role in the evaluation is played by the consistency condition (2.10) which implies that the density ρ is a *locally conserved quantity*; mass is neither created nor destroyed during relaxation to the local equilibrium.

It is important to distinguish between the cases when the density in the equilibrium is constant, and when it varies in space. In the first case, we speak of *global equilibrium* (or uniform equilibrium, or simply equilibrium), in the second case one speaks of the *local equilibrium* (or non-uniform equilibrium). The difference between these two notions is crucial: Global equilibria are *solutions* of the kinetic equation, whereas local equilibria are not. In other words, the global equilibrium annuls both the right- and the left-hand sides of the kinetic equation, whereas the local equilibrium only the right-hand side. So, if we take as the initial condition the uniform equilibrium, no further change of this state will ever occur. On the contrary, if we take a local equilibrium with a non-uniform density as the initial condition, its spatial derivative will be the driving force which will move the solution out of the local equilibrium.

Under certain assumptions we will be able to compute the solution of the kinetics equations in the next section. Before doing so, however, we introduce a slightly more general and compact notation. We denote by n_d the number of discrete velocities ($n_d = 3$ in our model), and label the discrete velocities and the corresponding populations by an integer i, $i = 1, \ldots, n_d$. In our case we can assign, for example (though any other assignment will also do) $1 \to -,\ 2 \to 0, 3 \to +$. In this notation, the kinetic equations (2.7) read:

$$\partial_t f_i(x,t) + c_i \partial_x f_i(x,t) = -\frac{1}{\tau}\left[f_i(x,t) - f_i^{\mathrm{eq}}(\rho(f(x,t)))\right], \tag{2.15}$$

where, according to the assignment adopted, $c_1 = -c,\ c_2 = 0,\ c_3 = c$.

3 Hydrodynamic limit

3.1 Invariance equation and its solution

In order to complete the construction of the kinetic equations (2.7) we must specify the equilibrium. This requires an analysis of the kinetic equation to obtain information about the equilibrium expressions that leads to the constitutive equation (2.5).

It will be slightly more instructive to rewrite the kinetic equations (2.7) using three linear functions of the populations instead of the three populations:

$$\rho(x,t) = f_-(x,t) + f_0(x,t) + f_+(x,t), \tag{3.1a}$$

$$j(x,t) = -cf_-(x,t) + cf_+(x,t), \tag{3.1b}$$

$$P(x,t) = c^2 f_-(x,t) + c^2 f_+(x,t). \tag{3.1c}$$

The density and the momentum flux were already introduced above. The third function, P, is the flux of the momentum flux, or the pressure. By differentiating in time the functions (3.1), and substituting the time derivatives of the populations from the kinetic equations (2.7), we arrive at the following system of equations (model A):

$$\partial_t \rho + \partial_x j = 0, \tag{3.2a}$$

$$\partial_t j + \partial_x P = -(j - j^{\mathrm{eq}})/\tau, \tag{3.2b}$$

$$\partial_t P + c^2 \partial_x j = -(P - P^{\mathrm{eq}})/\tau. \tag{3.2c}$$

Linear functions of the populations like (3.1) are called *moments*, and the representation of the kinetic equations (2.7) as the set of equations (3.2a), (3.2b) and (3.2c) is termed the *moment system*. The set of the three kinetic equations (2.7) is completely equivalent to the three equations of the moment system: it is just a matter of convenience (or taste) which one to use in the analysis (but not for the numerical implementation!).

Let us now introduce our target constitutive relation (2.5) into the equation for the momentum flux (3.2b): $j^{\mathrm{eq}} = v\rho$. What we expect to happen is that, if the relaxation time τ is small, the right-hand side of the momentum flux equation (3.2b) will dominate all other contributions, and, to leading order, $j \approx v\rho$. That is, if the relaxation of j to its target value is fast, then j is *slaved* by the dynamics of ρ on a larger time scale. The constitutive equation (2.5) will then be satisfied *dynamically*.

For the pressure, P^{eq}, we can infer that it will also be some function of ρ (since the equilibrium populations depend solely on ρ). We can even assume that P^{eq} will be proportional to ρ, but we are interested in how it should depend on the parameter v (the constant advection velocity) as well.

In order to obtain the constitutive relations for $j(\rho)$ and $P(\rho)$ which emerge as the result of the dynamics at $\tau \ll 1$, we need a way to zoom into the long-time large-scale dynamics of (3.2). Since we are led by the idea of slaving, the constitutive relations $j(\rho)$ and $P(\rho)$ mean that j and P will not have independent dynamics. Instead, their time derivatives will be dictated by the time derivative of density. We then compute the time derivative of j and P by the chain rule:

$$\partial_t j = \left(\frac{dj}{d\rho}\right)\partial_t\rho, \quad \partial_t P = \left(\frac{dP}{d\rho}\right)\partial_t\rho. \tag{3.3}$$

Substituting these expressions for the time derivatives of the constitutive relations into equations (3.2b) and (3.2c) we obtain:

$$\begin{aligned}\left(\frac{dj}{d\rho}\right)\partial_t\rho &= -\partial_x P - \frac{1}{\tau}(j - v\rho),\\ \left(\frac{dP}{d\rho}\right)\partial_t\rho &= -c^2\partial_x j - \frac{1}{\tau}(P - P^{\mathrm{eq}}).\end{aligned} \tag{3.4}$$

Since the time derivative of the density is given by the conservation law (3.2a), the latter system can be written as:

$$\begin{aligned}\left(\frac{dj}{d\rho}\right)(-\partial_x j) &= -\partial_x P - \frac{1}{\tau}(j - v\rho),\\ \left(\frac{dP}{d\rho}\right)(-\partial_x j) &= -c^2\partial_x j - \frac{1}{\tau}(P - P^{\mathrm{eq}}).\end{aligned} \tag{3.5}$$

Finally, the spatial derivatives of the the momentum flux and of the pressure can be again computed by the chain rule:

$$\begin{aligned}-\left(\frac{dj}{d\rho}\right)^2\partial_x\rho &= -\left(\frac{dP}{d\rho}\right)\partial_x\rho - \frac{1}{\tau}(j - v\rho),\\ -\left(\frac{dP}{d\rho}\right)\left(\frac{dj}{d\rho}\right)\partial_x\rho &= -c^2\left(\frac{dj}{d\rho}\right)\partial_x\rho - \frac{1}{\tau}(P - P^{\mathrm{eq}}).\end{aligned} \tag{3.6}$$

The system of relations for the constitutive relations (3.6) is known as *invariance condition* and should be read as follows: For *any* spatial dependence $\rho(x)$, functions $j(\rho)$ and $P(\rho)$ should be such that they satisfy (3.6). What is very important in (3.6) is the lack of time derivatives of j and P which have been eliminated in favor of the spatial derivatives.

The invariance condition is an exact statement (identity) about the constitutive relations: any constitutive relation $j(\rho)$, and $P(\rho)$ has to satisfy (3.6) at least to some order of accuracy. However, the invariance condition itself does not provide the constitutive relation. In order to get the answer, we must consider it as an equation (*invariance equation* [19]) for the unknown constitutive relations $j(\rho)$ and $P(\rho)$, and try to solve it. This solution can be obtained approximately by a perturbation method, exploiting the smallness of the relaxation time τ and expanding the constitutive relation in power series,

$$\begin{aligned} j &= j^{(0)} + \tau j^{(1)} + \tau^2 j^{(2)} + ... \\ P &= P^{(0)} + \tau P^{(1)} + \tau^2 P^{(2)} + ... \end{aligned} \tag{3.7}$$

Upon substitution of the expansion (3.7) into the system (3.6), we equate the terms of same order in τ. The leading order terms, $j^{(0)}$ and $P^{(0)}$, must cancel the $1/\tau$ terms in the right-hand side of equations (3.6). This gives:

$$j^{(0)} = v\rho,\ P^{(0)} = P^{\mathrm{eq}}. \tag{3.8}$$

Based on linearity and dimensionality arguments, we now assume that

$$P^{\mathrm{eq}}(\rho, v) = P^{(0)}(\rho, v) = \rho U^2(v), \tag{3.9}$$

where the velocity U can be a function of v that will be specified below. In our constant advection velocity case, this dependence is not so crucial, but we shall keep it for the general case. The equilibrium pressure (3.9) is the last input needed to establish the constitutive relations: once the leading-order terms in the expansion (3.7) are specified, all subsequent terms are computed recurrently from the invariance equation. The first-order term $j^{(1)}$ can be obtained from the first equation of system (3.6),

$$-\left(\frac{dj^{(0)}}{d\rho}\right)^2 \partial_x \rho = -\left(\frac{dP^{(0)}}{d\rho}\right) \partial_x \rho - j^{(1)}, \tag{3.10}$$

resulting in

$$j^{(1)} = -\left(U^2(v) - v^2\right) \partial_x \rho. \tag{3.11}$$

The approximate constitutive relation for the flux, up to linear order in τ, is then:

$$j = j^{(0)} + \tau j^{(1)} = v\rho - \tau\left(U^2(v) - v^2\right) \partial_x \rho. \tag{3.12}$$

With the constitutive relation (3.12), the final equation for the density of the 'true' particles (3.2a) becomes:

$$\partial_t \rho + v \partial_x \rho - D(v) \partial_x^2 \rho = 0. \tag{3.13}$$

This is the *advection-diffusion equation* with the *diffusion coefficient* $D(v)$,

$$D(v) = \tau\left(U^2(v) - v^2\right). \tag{3.14}$$

The method used to derive the advection-diffusion equation (3.13) by the expansion in τ is the simplest instance of the *Chapman-Enskog method* [20] of solving the invariance equation associated with the kinetic equations. For the sake of completeness, let us also find the first-order solution for the pressure $P^{(1)}$. From the second equation in (3.6), we obtain

$$-\left(\frac{dP^{(0)}}{d\rho}\right)\left(\frac{dj^{(0)}}{d\rho}\right)\partial_x\rho = -c^2\left(\frac{dj^{(0)}}{d\rho}\right)\partial_x\rho - P^{(1)}. \tag{3.15}$$

Substituting zeroth-order expressions (3.8), we find

$$P^{(1)} = v\left[c^2 - U^2(v)\right]\partial_x\rho. \tag{3.16}$$

It is crucial that the diffusion coefficient is nonnegative. Then, equation (3.13) will be well-behaved; it will show the propagation of the density profile ρ together with its smearing. The larger the diffusion coefficient, the more the smearing. However, if D is negative, this immediately leads to anti-diffusion, instead of a smearing we will see sharpening of the peaks in the profile to the extend it will become non-smooth, not differentiable and so on, something we would like to avoid. The diffusion coefficient is an example of a *transport coefficient*, and it reflects the fact that we have now quarks which brought in an irreversible behavior.

Since it is crucial that the diffusion coefficient is nonnegative, the velocity $U(v)$ must be greater than the velocity of the particles v. The simplest guess for $U(v)$ would be to write $P^{\mathrm{eq}} = c_{\mathrm{s}}^2\rho + \rho v^2$. This form of the equilibrium pressure contains the *hydrostatic pressure*, $p = c_{\mathrm{s}}^2\rho$, where c_{s} is the *speed of sound.* The unknown velocity $U(v)$ in equation (3.9) is then

$$U(v) = c_{\mathrm{s}}\sqrt{1 + \mathrm{Ma}^2}, \tag{3.17}$$

where $\mathrm{Ma} = |v|/c_{\mathrm{s}}$ is the *Mach number.* With this choice, the diffusion coefficient D in the advection-diffusion equation becomes independent of the velocity of the particles,

$$D = \tau c_{\mathrm{s}}^2. \tag{3.18}$$

However, we could also address other possibilities, not necessarily leading to (3.18), since a diffusion coefficient independent of v is not our particular goal here.

What remains is to find equilibrium populations which respect all these properties. The construction of the equilibria is, in general, one of the most important ingredients in the lattice Boltzmann method. We will not discuss in this paper how equilibria are derived; rather, will use one of the possibilities [21]:

$$f_0^{\mathrm{eq}}(\rho,v) = 2\rho\left[2-\sqrt{1+v^2/c_{\mathrm{s}}^2}\right]/3, \tag{3.19a}$$

$$f_+^{\mathrm{eq}}(\rho,v) = \rho\left[(vc-c_s^2)/2c_s^2+\sqrt{1+v^2/c_{\mathrm{s}}^2}\right]/3, \tag{3.19b}$$

$$f_-^{\mathrm{eq}}(\rho,v) = \rho\left[-(vc+c_s^2)/2c_s^2+\sqrt{1+v^2/c_{\mathrm{s}}^2}\right]/3, \tag{3.19c}$$

where the value of the speed of sound c_{s} is $c_{\mathrm{s}} = c/\sqrt{3}$. The equilibrium pressure corresponding to the equilibrium populations (3.19) reads

$$P^{\mathrm{eq}} = \rho c_{\mathrm{s}}^2\left(2\sqrt{1+v^2/c_{\mathrm{s}}^2}-1\right), \tag{3.20}$$

while the corresponding diffusion coefficient D is a function of the Mach number:

$$D = \tau c_{\mathrm{s}}^2\left(2\sqrt{1+\mathrm{Ma}^2}-(1+\mathrm{Ma}^2)\right). \tag{3.21}$$

We remark that the diffusion coefficient (3.21) differs from (3.18) only by the terms of order Ma^4 and higher:

$$D = \tau c_{\mathrm{s}}^2\left(1-\frac{1}{4}\mathrm{Ma}^4+\mathcal{O}(\mathrm{Ma}^6)\right). \tag{3.22}$$

Thus, for a *low Mach number flow* ($\mathrm{Ma} \ll 1$) the difference between the diffusion coefficients (3.21) and (3.18) is *negligible.*

In above, we have used the moment system to study the long-time dynamics but the same result can be derived directly from the kinetic equation (2.7). For the sake of completeness, we present the corresponding derivation in Appendix A.

The construction of the kinetic equation for the quarks is now complete: Kinetic equations (2.7) with the equilibria (3.19). The kinetic equations lead (if the relaxation time is small) to the advection-diffusion equation. The advection-diffusion equation is a very useful physical model on its own, but our target equation was actually the advection equation (2.3). Thus, we need to find a way to numerically attain as small as possible values of the diffusion coefficient. This will be our primary goal in the next section.

4 Over-relaxation

4.1 Implicit made explicit

In the previous section, we have devised the set of kinetic equations for fictitious particles (quarks) which recovers the advection-diffusion equation. Equation (3.19), suggests that we can factor the equilibrium function in the following way:

$$f_i^{\mathrm{eq}}(f) = \rho(f)\varphi_i(v) = [f_- + f_0 + f_+]\varphi_i(v). \tag{4.1}$$

Here i takes values $-$, 0, $+$, and the functions $\varphi_i(v)$ are the equilibrium populations at unit density,

$$\varphi_-(v) + \varphi_0(v) + \varphi_+(v) = 1, \tag{4.2}$$

which for the equilibrium (3.19) are:

$$\varphi_0(v) = 2\left[2 - \sqrt{1 + v^2/c_\mathrm{s}^2}\right]/3, \tag{4.3a}$$

$$\varphi_+(v) = \left[(vc - c_s^2)/2c_s^2 + \sqrt{1 + v^2/c_\mathrm{s}^2}\right]/3, \tag{4.3b}$$

$$\varphi_-(v) = \left[-(vc + c_s^2)/2c_s^2 + \sqrt{1 + v^2/c_\mathrm{s}^2}\right]/3. \tag{4.3c}$$

Furthermore, we shall denote, when necessary, the three-dimensional vector of populations f by

$$f(x,t) = \{f_j(x,t)\}. \tag{4.4}$$

For example, the shift of the arguments x and t in the population vector is denoted by

$$\{f_-(x - c\delta t, t + \delta t), f_0(x, t + \delta t), f_+(x + c\delta t, t + \delta t)\} = \{f_j(x + c_j\delta t, t + \delta t)\}. \tag{4.5}$$

The system of kinetic equations then reads:

$$\partial_t f_-(x,t) - c\partial_x f_-(x,t) = Q_-(\{f_j(x,t)\}), \tag{4.6a}$$

$$\partial_t f_0(x,t) = Q_0(\{f_j(x,t)\}), \tag{4.6b}$$

$$\partial_t f_+(x,t) + c\partial_x f_+(x,t) = Q_+(\{f_j(x,t)\}), \tag{4.6c}$$

where $Q_* = -\left(f_*(x,t) - \sum_{j=1}^{3} f_j(x,t)\varphi_*(v)\right)/\tau$, with $* = -, 0$ or $+$. Now, we have a set of three partial differential equations (4.6) and two natural time scales in the problem, which must be respected for stable and accurate simulations. One of the time scales is associated with the free flight of the quarks, and the other with relaxation. The relaxation time scale τ is much smaller than that of the free flight. Such a situation when there are two or more very different time scales in the problem is usually referred to as *stiffness*. In order to have an efficient simulation scheme for hydrodynamics, it is desirable to follow the time scale of the free flight, that is, we should be able to use a time step δt with $\delta t \gg \tau$. In this section, we describe such a discretization scheme, known as *over-relaxation*.

Integrating (4.6) for the time δt, we obtain:

$$f_i(x + c_i\delta t, t + \delta t) = f_i(x,t) + \int_0^{\delta t} Q_i\left(\{f_j(x + c_j s, t + s)\}\right) ds. \tag{4.7}$$

The time integrals of the relaxation terms can be evaluated by the trapezoidal rule[†]:

$$f_i(x + c_i\delta t, t + \delta t) \approx f_i(x,t) + \frac{\delta t}{2}\left[Q_i\left(\{f_j(x,t)\}\right) + Q_i\left(\{f_j(x + c_j\delta t, t + \delta t)\}\right)\right], \tag{4.8}$$

[†] The trapezoidal rule for the evaluation of an integral, $\int_t^T f(s)ds \approx (1/2)(f(T) + f(t))(T - t)$, assumes a linear interpolation between the 'present' (t) and the 'future' (T)

with the accuracy of the integration being $\mathcal{O}(\delta t^3)$. At first glance, this approximation does not seem very useful because the relationship between populations at two different times is *implicit*. In order to create an efficient explicit numerical scheme, we introduce a new set of functions g_i [22],

$$g_i(x,t) = f_i(x,t) - \frac{\delta t}{2} Q_i\left(\{f_j(x,t)\}\right). \tag{4.9}$$

Although we shall discuss this and related transformations in more details below, one remark should already be made here: While we started with *populations* f_i (that is, f_i are non-negative functions), the functions g_i (4.9) are *not*, strictly speaking, populations. The sign of the collision integral Q_i can be positive (and indeed at least one of Q_i *must* be positive if $f \neq f^{\text{eq}}$ because Q_i's must sum up to zero by the mass conservation). So, if δt is fixed at some value, we *can* get negative values of g_i, and in general we do need to worry about this. We will return to this later.

It should be noted that local conservation is the same for f and g. Indeed, by the conservation of mass, $\sum_{i=1}^{n_d} Q_i(f) = 0$. Hence, $\sum_{i=1}^{n_d} f_i = \sum_{i=1}^{n_d} g_i$, implying that at equilibrium both variables are equal, $f_i^{\text{eq}} = g_i^{\text{eq}}$.

For the BGK model, we find upon direct computation (we apply the notation convention (4.4) to the three-component vector function g):

$$Q_i\left(g(x,t)\right) = \left(1 + \delta t/2\tau\right) Q_i\left(f(x,t)\right). \tag{4.10}$$

In terms of g, equation (4.8) (omitting the $\mathcal{O}(\delta t^3)$ errors) can be rewritten as:

$$g_i(x + c_i\delta t, t + \delta t) = g_i(x,t) + \left(\frac{2\tau\delta t}{2\tau + \delta t}\right) Q_i\left(\{g_j(x,t)\}\right). \tag{4.11}$$

Expanding the BGK relaxation terms Q_i, equation (4.11) becomes:

$$g_i(x + c_i\delta t, t + \delta t) = g_i(x,t) - \omega\Big[g_i(x,t) - g_i^{\text{eq}}(g(x,t))\Big], \tag{4.12}$$

where ω (*discrete inverse relaxation time*) is a non-linear function of the inverse relaxation time $1/\tau$ and the time step δt:

$$\omega(\tau, \delta t) = \frac{2\delta t}{2\tau + \delta t}. \tag{4.13}$$

At any fixed δt, the range of ω is the interval $[0,2]$ (*linear stability interval*), and we have the following asymptotic limits *for a fixed δt*:

$$\omega \to 0 \text{ when } \tau \to \infty; \quad \omega \to 2 \text{ when } \tau \to 0. \tag{4.14}$$

The diffusion coefficient can be written in terms of δt and ω by solving (4.13) for τ: $\tau = \delta t\left(\frac{1}{\omega} - \frac{1}{2}\right)$. The general formula for the diffusion coefficient (3.14) then becomes

$$D = \delta t\left(\frac{1}{\omega} - \frac{1}{2}\right)\left(U^2(v) - v^2\right). \tag{4.15}$$

Equation (4.12) is one of the key results about the numerical implementation of the lattice Boltzmann models. We conclude this section with a discussion of some of the features of the discrete-time equation (4.12).

4.2 Discussion of the over-relaxation

A remarkable property the discrete-time equation (4.12) is its superficial similarity with the first-order Euler method, and it is very instructive to make a clear distinction between them. The Euler method, as applied to the evaluation of the integral in (4.7) gives

$$f_i(x + c_i\delta t, t + \delta t) = f_i(x,t) + \delta t Q_i\left(\{f_j(x,t)\}\right) + \mathcal{O}(\delta t^2), \tag{4.16}$$

and the corresponding scheme reads

$$f_i(x + c_i\delta t, t + \delta t) = f_i(x,t) - \delta t(f_i(x,t) - f_i^{\mathrm{eq}}(f(x,t))/\tau, \tag{4.17}$$

or

$$f_i(x + c_i\delta t, t + \delta t) = (1 - \delta t/\tau)\, f_i(x,t) + \delta t f_i^{\mathrm{eq}}(f(x,t))/\tau. \tag{4.18}$$

The latter form of the Euler scheme is particularly revealing because the right-hand side is simply a convex linear combination between the current population $f(x,t)$ and the equilibrium corresponding to this $f(x,t)$ [‡]. This puts a very severe restriction on the ratio of the time step to the relaxation time:

$$\delta t/\tau \leq 1. \tag{4.19}$$

If we want to achieve minimum possible values of the diffusion coefficient (proportional to τ), then we also must decrease the time step δt in order to respect (4.19).

On the other hand, our second-order accurate scheme (4.12) can be written as a convex linear combination of a different pair of functions:

$$g_i(x + c_i\delta t, t + \delta t) = \left(1 - \frac{\delta t}{2\tau + \delta t}\right) g_i(x,t) + \frac{\delta t}{2\tau + \delta t}\left(2g_i^{\mathrm{eq}}(g(x,t)) - g_i(x,t)\right), \tag{4.20}$$

or,

$$g_i(x + c_i\delta t, t + \delta t) = (1 - \beta)\, g_i(x,t) + \beta g_i^{\mathrm{mir}}(g(x,t)), \tag{4.21}$$

where $\beta(\tau, \delta t) = \delta t/(2\tau + \delta t)$. Here, we have introduced another vector g^{mir} (this is the simplest case of the *mirror state*): $g^{\mathrm{mir}} = 2g^{\mathrm{eq}}(g) - g$. Now, as long as g^{mir} is a population vector, the outcome of the right hand side in (4.21) will also be a population vector if $0 \leq \beta \leq 1$, and

$$\frac{\delta t}{2\tau + \delta t} \leq 1. \tag{4.22}$$

[‡] The convex linear combination between two vectors, $f^{(1)}$ and $f^{(2)}$, is the segment between them, $l(\alpha) = (1-\alpha)f^{(1)} + \alpha f^{(2)}$, where the parameter α takes values from 0 to 1. If all the components of the vectors $f^{(1)}$ and $f^{(2)}$ are nonnegative, then also all the components of any of the vectors $l(\alpha)$ are nonnegative.

We can now see that *small values of the diffusion coefficient can be attained because the time step δt does not need to be small as $\tau \to 0$*:

$$\beta \to 1 \quad \text{as} \ \ \tau \to 0 \ \ \text{for any} \ \ \delta t \ ! \tag{4.23}$$

Using the parameter β instead of ω, the diffusion coefficient (4.15) is written as:

$$D = \delta t \left(\frac{1}{2\beta} - \frac{1}{2} \right) \left(U^2(v) - v^2 \right). \tag{4.24}$$

The time step δt is no longer restricted to be smaller than the relaxation time τ, and we have achieved our desired objective of obtaining a discrete kinetic equation with a large time step. In practical terms, this means that we have an explicit second-order accurate numerical scheme (4.12) with the computational cost of a first-order scheme. This scheme is called *over-relaxation*. The reasoning for this terminology is rather obvious: The solution to the space-independent kinetic equation

$$\partial_t f = -(f - f^{\mathrm{eq}})/\tau, \tag{4.25}$$

with the initial condition f^{in} is: $f(t) = \left(1 - e^{-t/\tau}\right) f^{\mathrm{eq}} + e^{-t/\tau} f^{\mathrm{in}}$. Geometrically, we can consider a ray with the origin at f^{in} and directed towards f^{eq}:

$$l(\alpha) = (1-\alpha) f^{\mathrm{in}} + \alpha f^{\mathrm{eq}}, \quad \alpha \geq 0. \tag{4.26}$$

The trajectory of this solution is the segment of this ray between f^{in} and f^{eq}. Starting at f^{in}, it is not possible to cross along the ray “over” the equilibrium, since $f \to f^{\mathrm{eq}}$ as $t \to \infty$. This is why the first-order Euler (4.18) scheme which refers to the same segment between f^{in} and f^{eq} is sometimes called ‘under-relaxation’.

On the contrary, the right-hand side of equation (4.21) suggests that the segment between f^{in} and f^{eq} is only *half* the segment between f^{in} and f^{mir}, because f^{eq} corresponds to $\beta = 1/2$. All the points on the ray for $\beta > 1/2$ are “over-relaxed” states, and the mirror state is *maximally over-relaxed.*

From this discussion it becomes once again clear that over-relaxed states cannot be obtained from a “classical” relaxation, and one has to be careful in interpreting them as populations because they may become negative valued even if the initial state and the equilibrium are populations. We shall explain in detail in Section 6 how to extend the present over-relaxation scheme in order to guarantee that the over-relaxed states are also populations. Before doing so, we will discuss the spatial discretization, without distinguishing between over-relaxed states and populations.

5 Advantage of the lattice

5.1 The lattice Boltzmann method

In the previous section, we have established a promising second-order accurate in time numerical scheme for solving the kinetic equations. Now we need to pick the time step

δt in equation (4.12). To this end, we notice that we can endow the line with a lattice structure with spacing c, and assign the grid points at the nodes of the lattice. That is, we apply the rule:

$$\text{if } x \text{ is a grid point, then } x \pm c \text{ are also grid points.} \tag{5.1}$$

For such a grid, the time step δt is chosen so that any grid point is matched by the propagation, i.e.

$$\delta t = 1. \tag{5.2}$$

Thus, the kinetic equations (4.12) become the following *lattice Boltzmann scheme*,

$$g_i(x + c_i, t + 1) = g_i(x, t) - \omega \left(g_i(x, t) - g_i^{\rm eq}(f(x, t)) \right), \tag{5.3}$$

or, in terms of the mirror state, $g_i^{\rm mir} = 2g_i^{\rm eq}(g) - g_i$,

$$g_i(x + c_i, t + 1) = (1 - \beta) g_i(x, t) + \beta g_i^{\rm mir}(g(x, t)). \tag{5.4}$$

The diffusion coefficient is given by the formulas of the previous section for $\delta t = 1$ according to (5.2): equations (3.14) and (4.24) become, respectively,

$$D = \left(\frac{1}{\omega} - \frac{1}{2} \right) \left(U^2(v) - v^2 \right), \tag{5.5}$$

$$D = \left(\frac{1}{2\beta} - \frac{1}{2} \right) \left(U^2(v) - v^2 \right). \tag{5.6}$$

Note that this scheme is linear. We stress that it is written not for the populations but for the functions g_i (4.9) which might not be populations at all. Having said this, we will still call them populations (as it is done in the lattice Boltzmann literature).

The implementation of the lattice Boltzmann scheme is based on repeating the two steps:

• *Propagation step.* Populations of the links c_i are moved to the corresponding adjacent links:

$$g_i(x, t) \to g_i(x + c_i, t), \tag{5.7}$$

resulting in a repopulation of the lattice, $g_i^*(x, t)$.

•*Relaxation step.* Populations $g_i^*(x, t)$ are equilibrated by the over-relaxation rule:

$$g_i^*(x, t) \to (1 - \beta) g_i^*(x, t) + \beta g_i^{\rm mir}(g^*(x, t)). \tag{5.8}$$

This can be equivalently rewritten as

$$g_i^*(x, t) \to g_i^*(x, t) + \alpha\beta(g_i^{\rm eq}(g^*(x, t)) - g^*(x, t)), \;\; \alpha = 2. \tag{5.9}$$

We shall use the latter form of the relaxation step below in the sample code, and earmark the notation α for the maximal over-relaxation step. Later on we shall see how the entropy estimate enhances α but in the lattice Boltzmann scheme right now it is just a constant, $\alpha = 2$. The value of the parameter β is chosen between 0 and 1, so as to reproduce the desired value of the diffusion coefficient (5.6).

Thus, if the set of the discrete velocities can be matched on a regular lattice, and if the time step is chosen as (5.2), no further error is introduced by the spatial discretization of equation (4.12). The lattice Boltzmann scheme can be thus called *second-order accurate in time, accurate in space method* for solving the kinetic equations, and the implementation of the algorithm (5.3) becomes extremely simple on serial as well as on parallel computer architectures. However, the key ingredient of the algorithm is the over-relaxation rather than the lattice itself. The lattice just provides an important and very desirable advantage. In a sense, the 'infinitely accurate' spacial discretization is even an "overkill", because the accuracy of the scheme is anyway dictated by the second-order accuracy of the temporal discretization. Moreover, in hydrodynamic simulations, it is not always possible to map the computational domain on a regular lattice (e.g. in the presence of internal boundaries), and following the discretization of the particle's velocities while discretizing space is not always optimal.

5.2 Lattice Boltzmann code and simulation

Implementation of the lattice Boltzmann scheme for the transformed population consists of two repetitive operations, namely, the collision and the propagation step. Apart from these two operations, one also needs to apply the boundary conditions at the appropriate lattice sites. A pseudo code for the current LB scheme would look like:

```
Initialize the lattice with the equilibrium population, $f^{eq} = \rho^{in}\varphi(v^{in})$,
Loop for $N$ time steps
{
Perform relaxation step (5.9) at each lattice cite;
Propagate populations (5.7) to their neighboring lattices;
Apply boundary conditions;
}
Post process the populations to obtain the pressure and velocity fields.
```

The most efficient data structure for storing the populations would be a multi-dimensional array with the innermost dimension (contiguous data) used to store populations at one lattice cite (i.e. $f[3][Nx]$). An **C** realization of the pseudo code, combining the propagation step with periodic boundary conditions, looks like

```
void advect( double f[3][Nx] ) {
  int i;
  double temporary_left,temporary_right;
  temporary_right = f[1][Nx-1];  /* For periodic boundary condition */
  temporary_left = f[2][0];
  for ( i=Nx-1; i>0; i-- )
   {
     f[1][i] = f[1][i-1]; /* advection for direction 'right' */
   }
  for ( i=0; i< Nx-1; i++)
   {
     f[2][i] = f[2][i+1] ;  /* advection for direction 'left' */
   }
  f[1][0] = temporary_right;    /* Complete periodicity */
  f[2][L-1] = temporary_left;   }
```

Since the propagation step involves simply a transfer of the populations from one lattice cite to another and the boundary conditions are applied on only a small fraction of the lattice nodes, the relaxation step is the only computationally intensive part of the algorithm. The task is further simplified because of the locality of the relaxation process. A simple **C** realization of the relaxation step could be as follows:

```
void collide(double f[3][Nx], double Ma,  double beta  ) {
  int i;
  double feq[3];
  double rho,v,cs;
  cs = 1/sqrt(3);
  v = Ma*cs;
  for ( i=0; i< Nx; i++)
   {
     rho = f[0][i] + f[1][i] + f[2][i];
     feq[0] = 2*rho/3 * ( 2 - sqrt(1 + Ma*Ma) );
     feq[1] = rho/3 * ((v - cs*cs)/(2*cs*cs) + sqrt(1 + Ma*Ma) );
     feq[2] = rho/3 * ((-v - cs*cs)/(2*cs*cs) + sqrt(1 + Ma*Ma) );
     alpha = 2.0   /* setting the maximal over-relaxation */
     f[0][i] = f[0][i] - (alpha*beta) * ( f[0][i] - feq[0] );
     f[1][i] = f[1][i] - (alpha*beta) * ( f[1][i] - feq[1] );
     f[2][i] = f[2][i] - (alpha*beta) * ( f[2][i] - feq[2] );
   }   }
```

This lattice Boltzmann code can be ran with different initial profiles of $\rho(x,0)$ like Gaussian peak, square wave, hyperbolic tangent etc. Let us consider several examples.

Consider first the advection of a steep Gaussian profile,

$$\rho(x,0) = 1.0 + 0.5\exp\left(-5000\,(x/N - 0.25)^2\right), \tag{5.10}$$

with periodic boundary conditions (N is the number of grid points). In the simulation presented in Fig. 1(a), the advection velocity is $v = 0.1$, corresponding to $\mathrm{Ma} = 0.17$, and

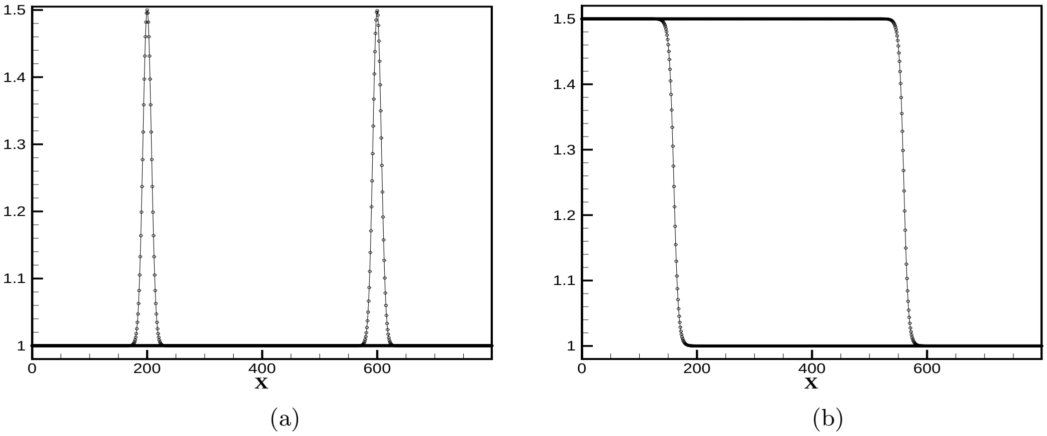


Figure 1: Advection of (a): the Gaussian peak with the speed $v = 0.1$ and (b): the $tanh$ profile ($\delta = 0.01$) after $t = 4000$. Diffusion coefficient $D = 5 \times 10^{-8}$. Grid size $N = 800$.

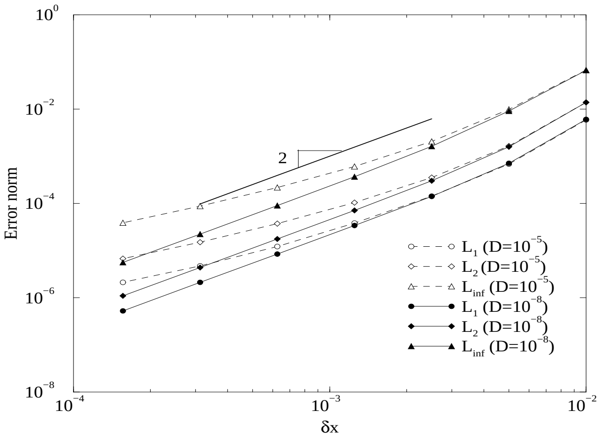


Figure 2: Error norms for the advection of the Gaussian peak after one period.

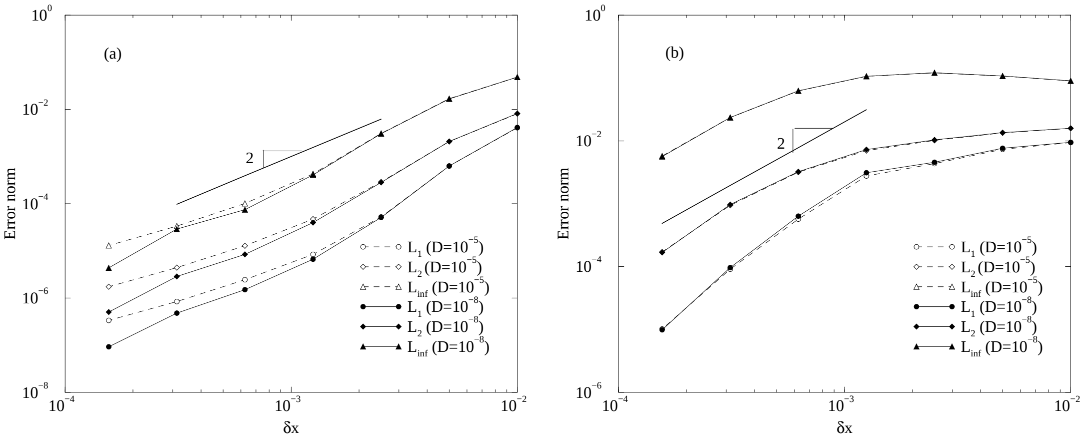
the diffusion coefficient $D = 5 \times 10^{-8}$. Long-time propagation ($t = 4000$ lattice units) does not show any significant distortions of the initial profile.

The effect of the size of the discretization interval and of the diffusion coefficient on the L_1, L_2 and L_∞ error norms can be seen in Fig. 2 after one return of the Gaussian to the initial location (this can be qualified as a long time integration). It can be clearly seen that for low enough diffusion coefficient, all three error norms follow the expected second-order accuracy.

The advection of hyperbolic tangent profiles,

$$\rho(x,0) = 1.0 + 0.5\left(1 - \tanh\left(\left(x/N - 0.2\right)/\delta\right)\right), \tag{5.11}$$

with variable thickness, δ, were also computed with the lattice Boltzmann code. In this case, zero-gradient boundary conditions (populations at node 0 are taken from node 1,



(a) (b)

Figure 3: Error norms for the advection of the tanh profile: (a) $\delta = 0.1$, (b) $\delta = 0.001$.

and populations at node N are taken from node $N-1$) were used instead of periodic ones, and the error after 4000 time steps was analyzed. In Fig. 1(b), propagation of the tanh profile with $\delta = 0.01$ is shown with the advection velocity is $v = 0.1$, corresponding to Mach number $\mathrm{Ma} = 0.17$, and the diffusion coefficient $D = 5 \times 10^{-8}$. Again, the quality of the numerical solution is very good.

The error norms for two profiles with decreasing δ are shown in Fig. 3. In both cases, second-order accuracy is recovered, provided that the steep layer is adequately resolved. In the steeper profile case, the error is insensitive to both values of the diffusion coefficient considered and is determined by the amplitude of the oscillations close to the steep increase.

As we can see from these examples, the lattice Boltzmann scheme runs stably, very low values of the diffusion coefficients can be achieved, and the quality of the result depends only on the resolution of the steepest gradients of the initial density profile.

In order to discuss the oscillations caused by under-resolution, we consider the final case of the advection of a square density profile,

$$\rho(x,0) = \begin{cases} 1.5, & \text{if } \; 200 \le x \le 400 \\ 1.0, & \text{otherwise} \end{cases} \tag{5.12}$$

with periodic boundary conditions. This is a somewhat pathological case, since the derivative of the density profile is infinite at the discontinuity, regardless of how fine is the grid. This is clearly the case which will always be beyond the applicability of the hydrodynamic limit, no matter how small the diffusion coefficient is taken. Small values of the diffusion coefficient can be attained; however, oscillations start growing at the discontinuities (see Fig. 4). Below we shall demonstrate how a refined scheme – the entropic lattice Boltzmann method – solves this problem.

The error in this case is governed by the error at the discontinuities and, at best, the scheme shows first-order accuracy (Fig. 5). The diffusion coefficient provides a well-

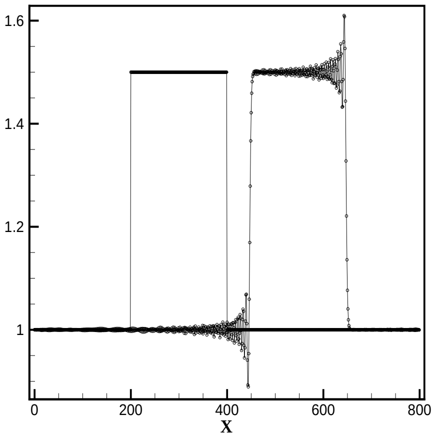


Figure 4: Advection of the step profile after $t = 3000$. Diffusion coefficient $D = 5 \times 10^{-5}$. Grid size $N = 800$.

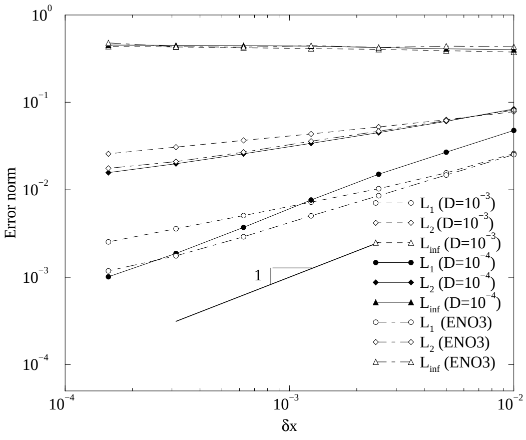


Figure 5: Error norms for the advection of the square density profile after one period with the LB and a 3^{rd}-order ENO scheme.

defined parameter to dump the oscillations at the discontinuities. The error behavior is the same as the 3^{rd}-order essentially non-oscillatory (ENO) method of Harten, Engquist, Chakravarthy, and Osher [23].

To conclude, we see that the lattice Boltzmann code performs excellently if the initial data are resolved. This is good news. However, if there is not enough resolution, that is, not enough grid points in regions where the density varies appreciably (it is just one grid point at the discontinuity, for example), oscillations are triggered. Moreover, given enough time, the amplitude of these oscillations can grow so large that the density will become negative at some points, rendering the solution nonphysical. Our goal is now to refine the lattice Boltzmann algorithm in such a way as to mitigate this problem. In the first place, this requires understanding of the numerically stability of the lattice Boltzmann scheme. This leads us to the notion of *entropy* which we are discuss in the next section.

6 Advantage of the entropy

6.1 Entropy function and the H-theorem

We continue with observations about our system (2.7). Consider a quadratic function of three real variables,

$$H(f) = \frac{f_-^2}{2\varphi_-(v)} + \frac{f_0^2}{2\varphi_0(v)} + \frac{f_+^2}{2\varphi_+(v)}. \tag{6.1}$$

Note that H is defined for all values of f_i, both positive and negative. Let us consider all the vectors f corresponding to the same value of density ρ,

$$f_- + f_0 + f_+ = \rho. \tag{6.2}$$

If the density is fixed, then equation (6.2) is a linear constraint and the vectors f cannot be chosen arbitrarily. In other words, the constraint (6.2) defines a linear subspace L_ρ corresponding to the specified value of ρ:

$$L_\rho = \{\text{All vectors } \ f = (f_-, f_0, f_+) \ \text{ satisfying } \ f_- + f_0 + f_+ = \rho\}. \tag{6.3}$$

If the equilibrium at unit density $\varphi(v)$ is a positive vector (all $\varphi_i(v) > 0$), then the function $H(f)$ (6.1) is *convex*, and the Hessian matrix, G, of H is positive-definite. Indeed, computing the second derivatives $\partial^2 H/\partial f_i \partial f_j$ of (6.1) we obtain

$$G = \begin{pmatrix} \frac{1}{\varphi_-(v)} & 0 & 0 \\ 0 & \frac{1}{\varphi_0(v)} & 0 \\ 0 & 0 & \frac{1}{\varphi_+(v)} \end{pmatrix}. \tag{6.4}$$

It is easy to check that $(f_-, f_0, f_+)G(f_-, f_0, f_+)^T \geq 0$. The equilibrium at density ρ, $f^{\rm eq}(\rho)$, is of course one of the vectors which satisfies the constraint (6.2). However, it is not just one of such vectors but it also furnishes the minimum of the function H on the linear subspace L_ρ. Indeed, the *extremum condition* for the function H subject to the constraint (6.2) reads:

$$\left[\frac{\partial H}{\partial f_-} - \lambda \frac{\partial \rho}{\partial f_-}\right]_{f^{\rm extr}} = 0, \quad \left[\frac{\partial H}{\partial f_0} - \lambda \frac{\partial \rho}{\partial f_0}\right]_{f^{\rm extr}} = 0, \quad \left[\frac{\partial H}{\partial f_+} - \lambda \frac{\partial \rho}{\partial f_+}\right]_{f^{\rm extr}} = 0. \tag{6.5}$$

Here, λ is the *Lagrange multiplier* corresponding to the constraint (6.2), and must also be determined. The extremum condition states that the vector of partial derivatives of the function H is orthogonal to the subspace L_ρ at their common point – the point of the extremum. Computing the derivatives in (6.5), we obtain:

$$\frac{f_-^{\rm extr}}{\varphi_-(v)} - \lambda = 0, \quad \frac{f_0^{\rm extr}}{\varphi_0(v)} - \lambda = 0, \quad \frac{f_+^{\rm extr}}{\varphi_+(v)} - \lambda = 0, \tag{6.6}$$

or,

$$f_-^{\rm extr} = \lambda\varphi_-(v), \ \ f_0^{\rm extr} = \lambda\varphi_0(v), \ \ f_+^{\rm extr} = \lambda\varphi_+(v). \tag{6.7}$$

We now need to determine the Lagrange multiplier from the condition that the extremum (6.7) belongs to L_ρ. We thus write,

$$f_-^{\rm extr} + f_0^{\rm extr} + f_+^{\rm extr} = \rho, \tag{6.8}$$

which gives

$$\lambda(\varphi_-(v) + \varphi_0(v) + \varphi_+(v)) = \rho. \tag{6.9}$$

Finally, taking into account the normalization (4.2), the Lagrange multiplier can be expressed as a function of the constraint: $\lambda=\rho$. Substituting this result back into (6.7), we observe that the extremum is indeed the equilibrium:

$$f_-^{\mathrm{extr}}=\rho\varphi_-(v),\ \ f_0^{\mathrm{extr}}=\rho\varphi_0(v),\ \ f_+^{\mathrm{extr}}=\rho\varphi_+(v). \tag{6.10}$$

This provides an alternative description of the equilibrium. Up till now, the equilibrium was constructed as the stationary point of the relaxation term. We have just found that the equilibrium is also the minimizer of the convex function $H(f)$ (6.1), subject to the constraint of the conservation law. This is reminiscent (up to the sign convention) of thermodynamics where equilibria are defined as maxima of entropy. Following this analogy, and possibly with some abuse of the terminology, we shall call all the convex functions we are going to construct and minimize as H-functions, and the negative of them – the *concave* functions which we then maximize – entropy functions S,

$$S=-H. \tag{6.11}$$

The link between the two specifications of equilibrium is the *H-theorem* (following the fundamental discovery of Boltzmann), which, loosely speaking, states that *entropy grows along solutions of the kinetic equation until it reaches the maximum at equilibrium.* We shall discuss it now for our model.

Above, we considered the density as a constant in the minimization problem. When density and populations are functions of x and t, one considers first the *density of the H-function*,

$$H(x,t)=H(f(x,t))=\frac{f_-^2(x,t)}{2\varphi_-(v)}+\frac{f_0^2(x,t)}{2\varphi_0(v)}+\frac{f_+^2(x,t)}{2\varphi_+(v)}. \tag{6.12}$$

The time derivative of this function due to the kinetic equations (2.7) reads (the derivative of a function $H(z)$ due to a dynamic system $\dot z=F(z)$ is an application of the chain rule: $\dot H=(dH/dz)\dot z=(dH/dz)F(z)$):

$$\partial_t H(x,t)+\partial_x j_H(x,t)=\sigma_H(x,t), \tag{6.13}$$

where

$$j_H(x,t)=\sum_{i=1}^{n_d}c_i\frac{f_i^2(x,t)}{2\varphi_i(v)}, \tag{6.14}$$

is the *flux of the density of the H-function*, and

$$\sigma_H(x,t)=-\frac{1}{\tau}\sum_{i=1}^{n_d}\frac{f_i(x,t)}{\varphi_i(v)}[f_i(x,t)-\rho(x,t)\varphi_i(v)], \tag{6.15}$$

is the *production rate of the density of the H-function.* In the entropy terminology, we have

$$\partial_t S(x,t)+\partial_x j_S(x,t)=\sigma_S(x,t), \tag{6.16}$$

which is called the *entropy balance equation*, with $j_S = -j_H$ the *entropy flux density*, and $\sigma_S = -\sigma_H$ the *entropy production density.*

In physics, the fundamental fact about the entropy production is that it is always nonnegative (second law of thermodynamics). To find out if our model also has a similar property we need to check if the function $\sigma_H(f)$ (6.15) is non-positive for all f's. By a direct computation:

$$\begin{aligned}\sigma_H(f) &= -\frac{1}{\tau}\sum_{i=1}^{n_d}\frac{f_i}{\varphi_i(v)}\left[f_i-\rho(f)\varphi_i(v)\right]\\ &= -\frac{1}{\tau}\left(\sum_{i=1}^{n_d}\frac{f_i}{\varphi_i(v)}\left[f_i-\rho(f)\varphi_i(v)\right]-\rho(f)\sum_{i=1}^{n_d}\left[f_i-\rho(f)\varphi_i(v)\right]\right)\\ &= -\frac{1}{\tau}\sum_{i=1}^{n_d}\frac{\left[f_i-\rho(f)\varphi_i(v)\right]\left[f_i-\rho(f)\varphi_i(v)\right]}{\varphi_i(v)}\leq 0.\end{aligned} \tag{6.17}$$

Note that in the second line we have "subtracted zero", because

$$\sum_{i=1}^{n_d}\left[f_i-\rho(f)\varphi_i(v)\right]=0,$$

by the local mass conservation.

Thus, the H-function production is non-positive (and the entropy production is nonnegative) for any vector f. From (6.17), it is obvious that σ_H becomes equal to zero at equilibrium,

$$\sigma_H(f)=0 \text{ if and only if } f=f^{\mathrm{eq}}. \tag{6.18}$$

Equation (6.18) is the third and final specification of equilibrium: entropy is not produced at these states. So, the complete specification of equilibria reads: *These are states of zero relaxation, which minimize the H-function under fixed conserved quantities, and where the production rate of H vanishes.*

With the result (6.17), let us come back to the entropy balance equation. Now we need to say something about the entropy flux. We shall insulate it from the balance equation by applying boundary conditions which ensure zero entropy flux through the boundaries. For example, we can consider periodic or zero flux at infinity boundary conditions. All this is called *closing the system.* We then introduce the total H-function (and the total entropy), and the total production of these quantities by integrating the corresponding densities over the domain (a segment $L=(L_-,L_+)$; L can also be the whole line):

$$\bar{H}(t)=\int_{L_-}^{L_+}H(f(x,t))dx,\quad \bar{\sigma}_H(t)=\int_{L_-}^{L_+}\sigma_H(f(x,t))dx. \tag{6.19}$$

Integrating the balance equation (6.13) over L, the integral of the spatial derivative of the flux reduces to the boundary values of $j_H(L_\mp,t)$ which drop out, and what remains is:

$$\frac{d\bar{H}(t)}{dt}=\bar{\sigma}(t). \tag{6.20}$$

Finally, since the integrand $\sigma_H(f(x,t))$ is always non-positive, we proved that for closed systems,

$$\frac{d\bar{H}(t)}{dt} \leq 0. \tag{6.21}$$

This is the *H-theorem* for our model. It states that in a closed system (when the entropy flux is of no concern), the total H-function is a non-increasing function of time. One usually refers to (6.21) as to the global H-theorem, and to the production inequality (6.17) as the local H-theorem. This immediately implies that in the space-independent case,

$$\frac{dH(t)}{dt} = \sigma(t) \leq 0. \tag{6.22}$$

The H-theorem is one of the cornerstones of statistical physics because it implies the irreversible behavior of "real" systems. For our purposes, another aspect is more relevant: this is a *stability theorem.* A function which behaves monotonically along solutions (monotonically decreases or increases) is called a *Lyapunov function*, and Lyapunov's direct method to prove stability is based on finding such a function. In our case, one Lyapunov function is the total H-function, and thus we may expect that it will be useful to assess stability of the lattice Boltzmann scheme. We shall explore this in the next section.

6.2 Entropy estimate of the over-relaxation

The over-relaxation scheme and its implication, the lattice Boltzmann method introduced in Sections 4 and 5 is a discrete-in-time kinetic system to which we want to apply the stability analysis of the previous section. In this section we extend the H-theorem to discrete time. We start with the space-independent case. This is appropriate since for the lattice Boltzmann method the update of the populations is split into the propagation step and the relaxation step. It is the relaxation step where we need to care about the entropy growth.

We begin with the definition of the mirror state. This notion was mentioned already in Section 4, and the over-relaxation from a state f was essentially a convex combination between f and f^{mir}. But now we would like to rederive it from stability considerations. For each vector f we define the mirror state f^{mir} as

$$f_i^{\mathrm{mir}}(f) = f_i + \alpha Q_i(f), \tag{6.23}$$

where $\alpha > 0$ is the nontrivial solution to the equation resulting from the *entropy estimate of the maximal over-relaxation*:

$$H\left(f + \alpha Q(f)\right) = H(f). \tag{6.24}$$

What motivates this definition of f^{mir}? Let us imagine the vector $f^{\mathrm{in}} = f$ as a point in the n_d-dimensional space (in the quark model, n_d is three, so this can be even shown graphically). Relaxation moves this point in the direction defined by the vector $Q(f)$.

In our model, Q is pointing towards the equilibrium point $f^{\rm eq}$. When moving the point in the direction Q, the H-function decreases towards the minimum at $f^{\rm eq}$. This is the 'classical' segment of the linear stability interval. However, if we do not stop at $f^{\rm eq}$ and continue moving in the direction Q, the H-function will start increasing again past $f^{\rm eq}$. This is, of course, a wrong behavior from the classical (*continuous-in-time*) standpoint. However, there is nothing wrong with this from the discrete-time point of view. Indeed, the states between the initial f^{in} and the final point $f^{\rm out}$ along the ray $f + aQ$, $a \geq 0$ where we will finish the update are simply not present, it does not matter whether $f^{\rm out}$ is reachable in the continuous-time dynamics or not. If the value of the H-function at the initial state, $H(f^{\rm in})$, is not larger than its value in the final state, $H(f^{\rm out})$, this move will still be acceptable as the H-function will decrease, $H(f^{\rm out}) - H(f^{\rm in}) \leq 0$, and we can use $f^{\rm out}$ even if it is "behind the mirror" at $f^{\rm eq}$. When we continue moving further and further along our ray, we will reach at some point a mirror state $f^{\rm mir}$ for which the value of the H-function is equal to the initial value $H(f^{\rm in})$. This is the limiting value given by equation (6.24). We cannot place $f^{\rm out}$ beyond $f^{\rm mir}$ because then the value of the H-function will be larger that at the beginning, and such states are ruled out by entropy considerations. The definition of the mirror state (6.23) and (6.24) is the precise statement of the qualitative picture just described.

It is easy to find α from (6.24) when the H-function is quadratic. Starting from

$$\sum_{i=1}^{n_d} \frac{[f_i + \alpha Q_i(f)]^2}{2\varphi_i(v)} = \sum_{i=1}^{n_d} \frac{f_i^2}{2\varphi_i(v)}, \tag{6.25}$$

we obtain a quadratic equation for α:

$$\alpha \left(\alpha \sum_{i=1}^{n_d} \frac{Q_i(f)^2}{2\varphi_i(v)} + 2 \sum_{i=1}^{n_d} \frac{f_i Q_i(f)}{2\varphi_i(v)} \right) = 0. \tag{6.26}$$

The trivial root $\alpha = 0$ is not of interest, as it corresponds to $f_i^{\rm mir} = f_i$. The nontrivial root is:

$$\alpha = -2 \left[\sum_{i=1}^{n_d} \frac{f_i Q_i(f)}{2\varphi_i(v)} \right] \left[\sum_{i=1}^{n_d} \frac{Q_i(f)^2}{2\varphi_i(v)} \right]^{-1}. \tag{6.27}$$

This expression is quite general. Using of the BGK form of Q_i,

$$Q_i(f) = -\frac{1}{\tau}(f_i - \rho(f)\varphi_i(v)),$$

and substituting this expression into equation (6.27), we obtain

$$\alpha = 2\tau \left[\sum_{i=1}^{n_d} \frac{f_i(f_i - \rho(f)\varphi_i(v))}{2\varphi_i(v)} \right] \left[\sum_{i=1}^{n_d} \frac{(f_i - \rho(f)\varphi_i(v))^2}{2\varphi_i(v)} \right]^{-1}. \tag{6.28}$$

The denominator in this expression is further simplified by exploiting mass conservation:

$$\begin{aligned}
&\sum_{i=1}^{n_d} \frac{(f_i - \rho(f)\varphi_i(v))^2}{2\varphi_i(v)} \\
= &\sum_{i=1}^{n_d} \frac{f_i(f_i - \rho(f)\varphi_i(v))}{2\varphi_i(v)} - \frac{\rho(f)}{2}\sum_{i=1}^{n_d}(f_i - \rho(f)\varphi_i(v)) \\
= &\sum_{i=1}^{n_d} \frac{f_i(f_i - \rho(f)\varphi_i(v))}{2\varphi_i(v)}.
\end{aligned} \tag{6.29}$$

Substituting this result back into (6.28), we finally derive the remarkably simple formula:

$$\alpha = 2\tau. \tag{6.30}$$

Substituting (6.30) back into the definition of the mirror state (6.23) and expanding the BGK relaxation term Q_i, we obtain:

$$f_i^{\mathrm{mir}}(f) = 2f_i^{\mathrm{eq}}(f) - f_i. \tag{6.31}$$

This result coincides with the definition of the mirror state obtained earlier in Section 4. However, now we have obtained it using a different and rather direct (entropy) argument. While staying with the space-independent case, we are able to prove the first version of the H-theorem for the discrete-time case.

Let us consider the discrete-time update,

$$f_i(t+\delta t) = (1-\beta)f_i(t) + \beta f_i^{\mathrm{mir}}(f(t)). \tag{6.32}$$

Here, β is a fixed parameter between 0 and 1. Its dependence on the time step δt is unimportant.

Let us denote by $H(t+\delta t)$ the value of the quadratic H-function at $f(t+\delta t)$:

$$H(t+\delta t) = \sum_{i=1}^{n_d} \frac{[f_i(t+\delta t)]^2}{2\varphi_i(v)}. \tag{6.33}$$

Now we come to the local discrete-time H-theorem:

$$H(t+\delta t) - H(t) = -2\beta(1-\beta)\sum_{i=1}^{n_d} \frac{(f_i(t) - f_i^{\mathrm{eq}})^2}{\varphi_i(v)}. \tag{6.34}$$

Thus, if β is between 0 and 1, after the discrete-time update (6.32), H is a non-increasing function:

$$H(t+\delta t) - H(t) \le 0. \tag{6.35}$$

It is also instructive to rewrite equation (6.34) by comparing it to the formula of the entropy production of the continuous case (6.17):

$$H(t+\delta t)-H(t)=2\tau\beta(1-\beta)\sigma_H(t), \tag{6.36}$$

where $\sigma_H(t)$ is the production of the H-function at time t.

Finally, let us prove the global H-theorem in the space-dependent case for the lattice Boltzmann spatial discretization. This is the same as in the continuous case; one needs to insulate the space dependence by summing over all the nodes of the lattice and assuming appropriate boundary conditions (we shall assume periodic boundary conditions). Thus, we introduce the total H-function,

$$\bar{H}(t)=\sum_x\sum_{i=1}^{n_d}\frac{[f_i(x,t)]^2}{2\varphi_i(v)}, \tag{6.37}$$

and consider how it changes during one time step:

$$\begin{aligned}\bar{H}(t+1) &= \sum_x\sum_{i=1}^{n_d}\frac{[f_i(x+c_i,t+1)]^2}{2\varphi_i(v)}\\ &= \sum_x\sum_{i=1}^{n_d}\frac{[(1-\beta)f_i(x,t)+\beta f^{\mathrm{mir}}(x,t)]^2}{2\varphi_i(v)}\\ &= \bar{H}(t)+2\tau\beta(1-\beta)\bar{\sigma}_H(t). \end{aligned}\tag{6.38}$$

Here, $-\bar{\sigma}_H(t)=-\sum_x\sigma_H(x,t)$ is the total entropy production at time t. Therefore,

$$\bar{H}(t+1)-\bar{H}(t)=2\tau\beta(1-\beta)\bar{\sigma}_H(t)\le 0, \tag{6.39}$$

and we have proven the H-theorem for the lattice Boltzmann discretization of the quark model. Thus, we can add one more statement about the lattice Boltzmann scheme for our quark model: It is a second-order accurate linear scheme, *and it is stable.*

What still remains somewhat unsatisfactory is that we have not managed to control positivity of the populations. Although we proved stability of the scheme, we have not proved that during the simulation all the populations f_i remain non-negative on every iteration. This means that negative values of f_i *may* appear during a simulation (we need to stress that this is not necessarily so). Although this will not result in any instability here, the physical meaning of the result will be lost as this may also result in negative density. The reason why we cannot guarantee positivity is that the quadratic H-function supports both positive and negative f_i's, and so the mirror states are also allowed to be both positive and negative. Since we see that the key issue in the definition of the mirror state is the choice of the H-function, we may wish to cure the positivity issue by choosing another H-function (not quadratic) which rules out negative f_i's. This we shall do in Section 6.4. In the next section we shall describe other Lyapunov functions for the kinetic system (2.7).

6.3 Entropy functions and positivity

The quadratic H-function (6.1) is an example of a Lyapunov function for our quark kinetic equations (2.7). Actually, there are many more Lyapunov functions for this model which are constructed from a smooth strictly convex function of one variable, $h(z)$, and the given equilibrium ($h(z)$ is convex if its derivative, $h'(z)$ is monotonic, that is the second derivative is sign-definite, $h''(z) > 0$). In the space-independent case, all these Lyapunov functions have the form

$$H(f) = \sum_{i=1}^{n_d} \varphi_i(v) h\left(\frac{f_i}{\varphi_i(v)}\right). \tag{6.40}$$

Some of the functions h proposed by different authors in various contexts are:

$z \ln z$	Boltzmann-Gibbs-Shannon	
$-\ln z$	Burg	
$az \ln z - (1-a)\ln z,\ 0 \le a \le 1$	Boltzmann-Burg family	(6.41)
$(z^q - 1)/(1-q),\ q > 0$	Tsallis family	
$z^2/2$	quadratic.	

The corresponding entropies are $S = -H$. The Boltzmann entropy is a fundamental quantity in statistical physics and thermodynamics. The Burg entropy is used in signal processing. The Boltzmann and Burg entropy are both additive (the entropy of the system composed of two independent sub-systems is equal to the sum of the entropies of the subsystems). The family obtained as a convex combination between the Boltzmann and Burg entropies is the class of additive entropies (there are no other entropies with the additivity property). The class of the Tsallis entropies is not additive but is widely used in various fits in the physics of complex systems. Finally, the quadratic entropy was used in the previous section.

All H-functions of the form (6.40) are equally well suited to be a Lyapunov function for the kinetic system (2.7). It can be shown that the equilibrium $f_i^{\text{eq}} = \rho\varphi_i(v)$ minimizes each of the functions (6.41) subject to fixed density, and the rate of production σ_H is non-positive (see Appendix B), exactly as was the case for the quadratic H function[§].

However, one distinction should be made: Our linear kinetic equation may have both positive and negative solutions, but some of the H-functions of the family (6.40) are not defined when f_i is negative. In fact, the Boltzmann, Burg and Tsallis (for non-integer q) functions of the list (6.41) are not defined for $z < 0$. Thus, by choosing H which is defined only if f_i's are populations in the construction of the mirror state, we will preclude f's leaking into the negative domain, and thus guarantee positivity of the density ρ.

[§] *Linear* kinetic equations always have a large number of entropy functions. Linear kinetic equations such as the Fokker-Planck equation, Markov chain etc., they all have families of Lyapunov functions of the form closely related to (6.40). The situation changes drastically for *nonlinear* kinetic equations such as the Boltzmann equation which have just one entropy function.

In the sequel, we choose the function $h(z) = z\ln z$. It is defined for $z \geq 0$ (at $z = 0$, the logarithm is not defined but $z\ln z \to 0$ as $z \to 0$, so $z\ln z$ is defined as zero at $z = 0$ by continuity). The H-function thus reads

$$H_{\mathrm{B}} = \sum_{i=1}^{n_d} f_i \ln\left(\frac{f_i}{\varphi_i(v)}\right), \tag{6.42}$$

and in expanded notation,

$$H_{\mathrm{B}} = f_0 \ln\left(\frac{f_0}{\varphi_0(v)}\right) + f_- \ln\left(\frac{f_-}{\varphi_-(v)}\right) + f_+ \ln\left(\frac{f_+}{\varphi_+(v)}\right). \tag{6.43}$$

The choice of (6.43) is not unique (we could use, for instance, the Burg or the Tsallis entropy), and is motivated by the fact that in the nonlinear lattice Boltzmann models [10,11], the H-function is not arbitrary but usually of the Boltzmann form. The Boltzmann H-function for our model will be used in the next section in order to modify the entropy estimate of the over-relaxation scheme in such a way as to make the lattice Boltzmann scheme stable *and* guarantee that density remains positive.

6.4 Entropic lattice Boltzmann scheme

The mirror state of the form (6.31) was derived using the quadratic H-function and resulted in the lattice Boltzmann algorithm. However, the entropy estimate (6.24) itself is not restricted to quadratic functions. Therefore, it allows the extension of the lattice Boltzmann method to any H-function of our choice, and in particular, to the Boltzmann H-function, H_{B}, (6.43) which supports only populations. This extension which results in the *entropic lattice Boltzmann scheme* amounts to replacing the mirror state (6.31) by the mirror state corresponding to H_{B}. The rest of the discretization steps remain as before, and we only write down the following set of equations:

$$\begin{aligned}
f_i(x+c_i,t+1) &= (1-\beta)f_i(x,t) + \beta f_i^{\mathrm{mir}}(f(x,t)), && (6.44)\\
f_i^{\mathrm{mir}}(f(x,t)) &= f_i(x,t) + \alpha(f(x,t))Q_i(f(x,t)), && (6.45)\\
Q_i(f(x,t)) &= -\frac{1}{\tau}(f_i(x,t) - \rho(f(x,t))\varphi_i(v)), && (6.46)\\
H_{\mathrm{B}}(f^{\mathrm{mir}}(x,t)) &= H_{\mathrm{B}}(f(x,t)). && (6.47)
\end{aligned}$$

The value of the parameter β is again fixed by the formula of the diffusion coefficient (5.6); that is,

$$\beta = \frac{1}{1+\frac{2D}{U^2(v)-v^2}}, \quad \text{in terms of the diffusion coefficient,} \tag{6.48}$$

or

$$\beta = \frac{1}{1+2\tau}, \quad \text{in terms of the relaxation time.} \tag{6.49}$$

The rules of the update of the populations are now as follows:

- **Propagation step**. Populations of the links c_i of the lattice are moved to the corresponding adjacent links:
$$f_i(x,t) \to f_i(x+c_i,t), \tag{6.50}$$
resulting in a new set of populations on every link, $f_i^*(x,t)$.

- **Relaxation step.** First, for each set of populations $f^*(x,t)$, the entropy estimate (6.47)
$$\sum_{i=1}^{n_d}(f_i^* + \alpha Q_i(f^*)) \ln\left(\frac{f_i^* + \alpha Q_i(f^*)}{\varphi_i(v)}\right) = \sum_{i=1}^{n_d} f_i^* \ln\left(\frac{f_i^*}{\varphi_i(v)}\right) \tag{6.51}$$
is solved.

  Second, once the solution $\alpha(x,t)$ is found for all nodes x, the mirror states are defined,
$$f_i^{\text{mir}}(f^*(x,t)) = f_i^*(x,t) + \alpha(x,t)Q_i(f^*(x,t)), \tag{6.52}$$
and populations $f_i^*(x,t)$ are over-relaxed by the rule:
$$f_i^*(x,t) \to (1-\beta)f_i^*(x,t) + \beta f_i^{\text{mir}}(f^*(x,t)). \tag{6.53}$$

The only difference from the previous lattice Boltzmann algorithm is in the use of another H-function for the estimation of the mirror step. The lattice Boltzmann algorithm can thus be viewed as a particular case of the entropic lattice Boltzmann method equipped with the quadratic entropy function. The nonlinear equation (6.47) determining α in the definition of the mirror population (6.45) must be solved on every iteration of the algorithm and at every node of the lattice x. In the general case, this solution can be obtained only numerically (not analytically as in the case of the quadratic H-function), and we will need to develop reliable ways to do this in order to minimize the computational overhead. We shall return to this issue later.

All Lyapunov functions (6.40) represent the irreversibility of the continuous-time kinetic equation, and there is no reason to prefer one over another. The H-theorem in the continuous case is just a *feature* of the dynamics. However, in the construction of the discrete-time entropic lattice Boltzmann scheme we *use* the H-function in order to *define* the maximum of the allowable over-relaxation. When the H-theorem is used, we have to decide which H-function to pick, and thus entropic lattice Boltzmann schemes with different H-functions are not equivalent. But should we worry about this?

In order to answer this question, let us remind that here we are not so much interested in the details of the dynamics far from the local equilibrium. Rather, we are willing to keep the dynamics close to the local equilibrium where we recover the advection-diffusion equation. So, let us examine what the entropy estimate reports when the states *are* close to the local equilibrium.

Let us consider a small deviation from equilibrium at some ρ, $f_i = f_i^{\mathrm{eq}}(\rho) + \delta f_i$, and expand the generic H-function (6.40) up to second-order around the equilibrium. Neglecting everything of order $\mathcal{O}(\delta f_i^3)$ and higher:

$$H_2(\delta f) = h(\rho) + h'(\rho)\delta\rho + \frac{1}{2}h''(\rho)\sum_{i=1}^{n_d}\frac{\delta f_i^2}{\varphi_i(v)}, \tag{6.54}$$

where $\delta\rho = \rho(\delta f) = \sum_{i=1}^{n_d}\delta f_i$ is the deviation of density from the equilibrium density $\rho(f^{\mathrm{eq}})$. The entropy estimate in the entropic lattice Boltzmann scheme appears at the relaxation step. Thus, for our analysis it is sufficient to consider only such deviations that do not change the density, $\delta\rho = 0$. If the state f is close to the equilibrium, the mirror state f^{mir} will also be close to the same equilibrium:

$$\begin{aligned} f_i^{\mathrm{mir}} &= f_i^{\mathrm{eq}} + \delta f_i - \frac{\alpha}{\tau}(f_i^{\mathrm{eq}} + \delta f_i - f_i^{\mathrm{eq}}(f^{\mathrm{eq}} + \delta f)) \\ &= f_i^{\mathrm{eq}} + (1 - \alpha/\tau)\,\delta f_i. \end{aligned} \tag{6.55}$$

Denoting $\delta f_i^{\mathrm{mir}} = (1 - \alpha/\tau)\,\delta f_i$, the entropy condition for α is written for the deviations as $H_2(\delta f^{\mathrm{mir}}) = H_2(\delta f)$, or,

$$\frac{1}{2}h''(\rho)\left(1 - \frac{\alpha}{\tau}\right)^2\sum_{i=1}^{n_d}\frac{\delta f_i^2}{\varphi_i(v)} = \frac{1}{2}h''(\rho)\sum_{i=1}^{n_d}\frac{\delta f_i^2}{\varphi_i(v)}, \tag{6.56}$$

resulting in the relevant solution being

$$\alpha^{\mathrm{eq}} = 2\tau. \tag{6.57}$$

Thus, *independently of what entropy function is used to determine the maximal over-relaxation*, close to the local equilibrium, all the corresponding entropic lattice Boltzmann schemes give the same answer for the over-relaxation parameter α: it is the same estimate derived earlier from the trapezoidal rule and from the quadratic H-function. Hence, in the domain we are interested in, that is, close to the local equilibria, all the entropic lattice Boltzmann schemes become equivalent and recover the advection-diffusion equation with the same diffusion coefficient when the parameter β is chosen according to the relation (5.6). *The limit of no diffusion, $D \to 0$, corresponds to $\beta \to 1$ for any H used.*

The difference between the schemes is in *how* the system is brought to its hydrodynamic limit. The use of the Boltzmann entropy function precludes appearance of negative populations, while the use of the quadratic cannot guarantee positivity. Thus, we choose to work with the entropic lattice Boltzmann scheme of this section which is then a second-order *nonlinearly* stable scheme *which guarantees positivity.*

Due to their feature of unconditional stability, entropic schemes for solving kinetic equations are clearly preferable. However, their efficiency in computations crucially depends on how efficiently we can solve for the entropy estimate. In our entropic lattice BGK scheme, for example, we need to solve the nonlinear equation (6.47) on every lattice node at each time step. For the Boltzmann entropy function, this equation includes logarithmic operations which are computationally expensive. In the next section we present a fast method for solving the entropy estimate, first outlined in [24].

6.5 Asymptotic expansion of the entropy estimate

The analysis in the previous section revealed that the limit of the entropy estimate when $f_i \to f_i^{\mathrm{eq}}$, $Q_i(f) \to 0$ is $\alpha \to 2\tau$ (6.57). A natural extension of this result is an asymptotic series expansion of the entropy estimate. In order to derive this expansion, let us rewrite the entropy estimate for the populations vector f (6.47) as follows:

$$\alpha \sum_{i=1}^{n_d} Q_i \ln\left(\frac{f_i}{\varphi_i(v)}\right) + \sum_{i=1}^{n_d} f_i \left(1 + \alpha\frac{Q_i}{f_i}\right) \ln\left(1 + \alpha\frac{Q_i}{f_i}\right) = 0. \tag{6.58}$$

We now expand the second term in (6.58) into Taylor series in powers of $x = \alpha\frac{Q_i}{f_i}$ using the formula

$$(1+x)\ln(1+x) = x + \sum_{m=2}^{\infty} \frac{(-1)^{m-2}}{m(m-1)} x^m. \tag{6.59}$$

This gives

$$\alpha\left\{-\sigma(f) + \sum_{n=1}^{\infty} \alpha^n c_n(f)\right\} = 0, \tag{6.60}$$

where $-\sigma \geq 0$ is the entropy production at the state f, and we have introduced the coefficients c_n,

$$c_n = \frac{(-1)^{n-1}}{n(n+1)} \sum_{i=1}^{n_d} \frac{Q_i^{n+1}}{f_i^n}, \quad n \geq 1. \tag{6.61}$$

Equation (6.60) is equivalent to the entropy estimate equation, and the relevant (nonzero) root can be obtained from

$$-\sigma(f) + \sum_{n=1}^{\infty} \alpha^n c_n(f) = 0. \tag{6.62}$$

The equilibrium value $\alpha^{\mathrm{eq}} = 2\tau$ is recovered from equation (6.62). Indeed, when the state is close to equilibrium, $f_i - f_i^{\mathrm{eq}} \sim \delta$, $Q_i \sim \delta$, $\delta/f_i^{\mathrm{eq}} \ll 1$, and the coefficients c_n have the following order of magnitude:

$$c_n \sim \delta^{n+1},$$

whereas $\sigma \sim \delta^2$. Therefore, when $\delta \to 0$, the first term of the infinite sum in (6.62) should balance the entropy production while the rest of the terms vanish and we obtain

$$-\sigma + \alpha_{(0)} c_1 = 0. \tag{6.63}$$

Therefore,

$$\alpha_{(0)} = \frac{\sigma}{c_1} = \frac{-\sum_{i=1}^{n_d} Q_i \ln\left(f_i/\varphi_i(v)\right)}{\frac{1}{2}\sum_{j=1}^{n_d} Q_j^2/f_j}. \tag{6.64}$$

By writing $f_i = f_i^{\rm eq} + \delta f_i$, $Q_i = -(1/\tau)\delta f_i$, and keeping the first non-vanishing terms in the numerator and denominator of (6.64):

$$\sigma = \frac{1}{\tau}\sum_{i=1}^{n_d}\frac{\delta f_i^2}{f_i^{\rm eq}} + o(\delta f_i^2), \quad c_1 = \frac{1}{2\tau^2}\sum_{i=1}^{n_d}\frac{\delta f_i^2}{f_i^{\rm eq}} + o(\delta f_i^2). \tag{6.65}$$

Therefore, the ratio of these expressions becomes independent of the deviation δf_i when $\delta f_i/f_i^{\rm eq} \to 0$, and

$$\lim_{\delta f_i/f_i^{\rm eq}\to 0} \alpha_{(0)} = 2\tau. \tag{6.66}$$

Thus, the solution to equation (6.63) recovers our previous result for the equilibrium value of the entropy estimate.

Next, we are going to gradually introduce corrections to the leading-order result $\alpha_{(0)}$ by exploring further terms in the series in (6.62). The next-order equation reads:

$$c_1\alpha_{(1)} + c_2\alpha_{(0)}^2 = 0, \tag{6.67}$$

where $\alpha_{(0)}$ is already defined by the previous-order equation (6.63), and we have

$$\begin{aligned}\alpha_{(1)} &= -\frac{c_2\alpha_{(0)}^2}{c_1} = -\frac{c_2}{c_1}\left(\frac{\sigma}{c_1}\right)^2 \\ &= \frac{4}{3}\left(\sum_{i=1}^{n_d} Q_i \ln\left(f_i/\varphi_i(v)\right)\right)^2\left(\sum_{j=1}^{n_d} Q_j^3/f_j^2\right)\left(\sum_{k=1}^{n_d} Q_k^2/f_k\right)^{-3}.\end{aligned} \tag{6.68}$$

In order to automate the procedure of finding further corrections, we introduce a book-keeping parameter ϵ into equation (6.62) by re-scaling $\sigma \to \delta^2\sigma$, $c_k \to \delta^{k+1}c_k$, and writing

$$\alpha = \sum_{k=0}^{\infty}\delta^k\alpha_{(k)}. \tag{6.69}$$

Equation (6.62) then becomes

$$-\delta^2\sigma + \sum_{n=1}^{\infty}\left(\sum_{k=0}^{\infty}\delta^k\alpha_{(k)}\right)^n \delta^{n+1}c_n(f) = 0. \tag{6.70}$$

Setting equal to zero terms of the same order in δ in this expression, we derive a sequence of recurrently solvable equations for the coefficients $\alpha_{(k)}$. The first of these equations is (6.63), the second is (6.67), and so on.

The resulting set of equations for the coefficients $\alpha_{(k)}$ is recurrently solvable to any order, and we obtain the solution to the entropy estimate in the form of approximations to the formal series (6.69). This procedure is a progressive refinement of the zero-order estimate (6.64) when deviations of populations away from equilibrium become more pronounced (at the end of computation we set $\delta = 1$). Let us write down, as an example, the equation for the coefficient $\alpha_{(2)}$:

$$c_1\alpha_{(2)} + 2c_2\alpha_{(0)}\alpha_{(1)} + c_3\alpha_{(0)}^3 = 0, \tag{6.71}$$

whereupon

$$\alpha_{(2)} = -(2c_2\alpha_{(0)}\alpha_{(1)} + c_3\alpha_{(0)}^3)/c_1 = \left(\frac{2c_2^2}{c_1^2} - \frac{c_3}{c_1}\right)\left(\frac{\sigma}{c_1}\right)^3. \tag{6.72}$$

By the same token, the function $\alpha_{(3)}$ is found as

$$\alpha_{(3)} = \left(\frac{5c_2c_3}{c_1^2} - \frac{5c_2^3}{c_1^3} - \frac{c_4}{c_1}\right)\left(\frac{\sigma}{c_1}\right)^4. \tag{6.73}$$

From the analysis of the formulas (6.63), (6.68) and (6.72) we can infer that the entropy estimate resulting from the above procedure has the form

$$\begin{aligned} \alpha &= \sum_{k=0}^{\infty}\alpha_{(l)} = \sum_{k=0}^{\infty}\tilde{\alpha}_{(l)}\left(\frac{\sigma}{c_1}\right)^{l+1} \\ &= \frac{\sigma}{c_1} - \frac{c_2}{c_1}\left(\frac{\sigma}{c_1}\right)^2 + \left(\frac{2c_2^2}{c_1^2} - \frac{c_3}{c_1}\right)\left(\frac{\sigma}{c_1}\right)^3 + \dots. \end{aligned} \tag{6.74}$$

In other words, the entropy estimate is represented by formula (6.74) as a series in powers of the entropy production σ, where the expansion coefficients are rational functions of c_k (6.61). Let us remind ourselves that (6.74) obeys the asymptotic property: all coefficients $\alpha_{(k)}$ with $k \geq 1$ tend to zero when the populations tend to local equilibrium, and only the contribution from the first term remains non-vanishing (and tends to 2τ).

From our experience, the zero-order approximation $\alpha_{(0)}$ (6.64) is often rather far from 2τ even if the populations are close to equilibrium. Since the approximation $\alpha_{(0)}$ is specified only by its asymptotic property (6.66), we may gain an even better approximation if we set σ/c_1 just equal to its asymptotic value 2τ in (6.74)

$$\alpha = 2\tau - \frac{c_2}{c_1}(2\tau)^2 + \left(\frac{2c_2^2}{c_1^2} - \frac{c_3}{c_1}\right)(2\tau)^3 + \left(\frac{5c_2c_3}{c_1^2} - \frac{5c_2^3}{c_1^3} - \frac{c_4}{c_1}\right)(2\tau)^4 + \dots. \tag{6.75}$$

This formula does not require computation of the entropy production, thus avoiding logarithmic evaluations altogether.

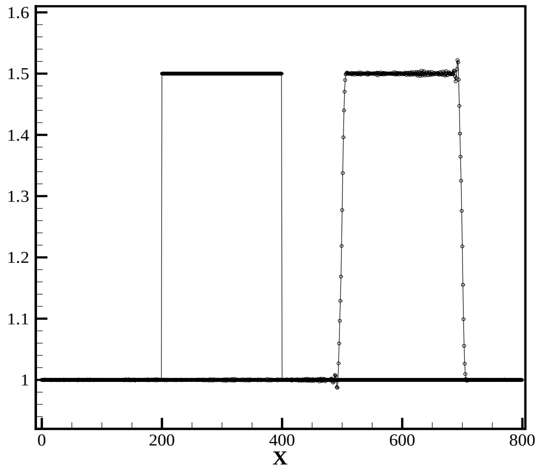


Figure 6: Advection of the step profile by the entropic lattice Boltzmann scheme with velocity $v = 0.1$, after $t = 3000$. Diffusion coefficient $D = 5 \times 10^{-5}$. Grid size $N = 800$.

By repeated application of the binomial formula, the explicit form of the recurrence solution $\alpha_{(k+1)}(\alpha_{(k)}, \ldots, \alpha_{(0)})$ can be obtained. We do not display this solution here not only because it is rather bulky but mainly because the too high orders in the expansion are almost surely useless. The expansion (6.75) is only asymptotic, and it is able to refine the equilibrium value $\alpha^{\text{eq}} = 2\tau$ only if the populations *are* sufficiently close to the (local) equilibrium .

Computing the entropy estimate using the analytic formula (6.75) is advantageous since it brings eventually no overhead in simulations. Application of (6.75) assumes setting a tolerance value for the deviation $|(f_i^{\text{eq}} - f_i)/f_i|$ (in particular, the tolerance in the simulation below was 0.04). Thus, the asymptotic expansion (6.75), if applicable, guarantees positivity of populations and nonlinear stability of the entropic lattice Boltzmann scheme with eventually no impact on the efficiency of the scheme. In practice, we found that the entropy estimate (6.74) is sufficiently accurate in up to some 90% of the lattice nodes. In the cases when the asymptotic expansion (6.75) does not work (large deviations of the populations from local equilibrium), different ways of solving for the entropy estimate should be explored, based, for example, on the positivity estimate of the populations and Newton-Raphson iterations (details on the numerical solution for the entropy estimate can be found in [25]).

In the lattice Boltzmann code of Section 5.2, the value of the maximal over-relaxation α is kept constant at 2.0 (this is without considering the entropic time stepping process). In order to implement an entropic version of the LB scheme, the parameter α has to be replaced by $\alpha(f)$ and calculated as described above.

In Fig. 6, the propagation of the square step profile is recomputed with the entropic lattice Boltzmann scheme. As compared to the same result computed with the lattice Boltzmann scheme (see Fig. 4), we see that oscillations at the discontinuities are suppressed, while no smearing of the profile occurs. This comparison demonstrates superiority of the entropic scheme.

7 Conclusions

Here we conclude our first tour over the lattice Boltzmann method. We presented the detailed derivation of lattice Boltzmann methods for the solution of the linear advection problem. The derivation clearly shows that in essence we solve an advection-diffusion problem with a user-defined diffusivity. In addition, the method can be made nonlinearly stable, more accurate, and guarantee positivity by entropy considerations.

Our plan for the forthcoming contributions to this series of papers is the following: In the second paper, again of a tutorial nature, we shall consider the simplest nonlinear situation (one-dimensional Navier-Stokes equations), and will discuss in detail how one constructs the entropic lattice Boltzmann method. In the third paper, we shall construct three-dimensional entropic lattice Boltzmann models which will be essentially sufficient for a reader to start a 'real' simulation. Further contributions will include the derivation of the lattice Boltzmann method from continuous kinetic theory, and special chapters such as grid refinement and off-lattice solvers.

Acknowledgments

We thank Dr. R. Brownlee for a careful reading of the manuscript. I. V. K. was supported by the BFE Project 100862. S. S. C. was supported by the ETH grant 0-20280-05.

A Invariance equation for populations

1. Write the invariance condition for the populations f_i, assuming they depend on x and t only through their dependence on the density, $f_i(x,t) = f_i(\rho(x,t))$ (following the language of kinetic theory, this is called the *normal solution*), and using the kinetic equation (2.7).

2. Solve the invariance equation for $f_i(\rho(x,t))$ by perturbation to order τ, assuming as above the local equilibrium $f_i^{\mathrm{eq}} = \rho\varphi_i(v)$, where $\sum_{i=1}^{n_d}\varphi_i(v) = 1$, $\sum_{i=1}^{n_d} c_i\varphi_i(v) = v$, as the initial approximation.

Using the chain rule to compute the derivative, $\partial_t f_i = \frac{\partial f_i}{\partial \rho}\partial_t\rho$, replacing $\partial_t\rho$ with the right-hand side of the density balance equation, $\partial_t\rho = -\partial_x\sum_{i=1}^{n_d} c_i f_i$, and equating the result to the time derivative of f_i from the kinetic equation (2.7), we obtain the invariance condition:

$$-\frac{\partial f_i}{\partial \rho}\partial_x\sum_{i=1}^{n_d} c_i f_i = -c_i\partial_x f_i - \frac{1}{\tau}(f_i - f^{\mathrm{eq}}(f)). \tag{A.1}$$

Considering the solution in the form of the expansion around the local equilibrium,

$$f_i = f_i^{\mathrm{eq}} + \tau f_i^{(1)} + \dots, \tag{A.2}$$

and substituting the latter expression into the invariance equation (A.1), we find the solution for $f_i^{(1)}$,

$$f_i^{(1)} = \frac{\partial f_i^{\mathrm{eq}}}{\partial \rho} \partial_x \sum_{i=1}^{n_d} c_i f_i^{\mathrm{eq}} - c_i \partial_x f_i^{\mathrm{eq}}. \tag{A.3}$$

Substituting f_i^{eq}, we find:

$$f_i^{(1)} = \frac{f_i^{\mathrm{eq}}}{\rho}(v - c_i)\partial_x \rho. \tag{A.4}$$

Thus, to the first order in τ, the solution to the invariance equation reads:

$$f_i = f_i^{\mathrm{eq}} - \tau \frac{f_i^{\mathrm{eq}}}{\rho}(c_i - v)\partial_x \rho. \tag{A.5}$$

Now we compute the constitutive relation for the momentum flux from this solution:

$$\begin{aligned} j &= \sum_{i=1}^{n_d} c_i f_i = \sum_{i=1}^{n_d} c_i f_i^{\mathrm{eq}} - \frac{\tau}{\rho}\left(\sum_{i=1}^{n_d} f_i^{\mathrm{eq}} c_i (c_i - v)\right)\partial_x \rho \\ &= \rho v - \tau \frac{P^{\mathrm{eq}} - \rho v^2}{\rho}\partial_x \rho = \rho v - \tau (U(v)^2 - v^2)\partial_x \rho, \end{aligned} \tag{A.6}$$

where in the last line we have used the expression for the equilibrium pressure (3.9),

$$P^{\mathrm{eq}} = \sum_{i=1}^{n_d} f^{\mathrm{eq}} c_i^2 = \rho \sum_{i=1}^{n_d} \varphi_i(v) c_i^2 = \rho U^2(v).$$

The flux (A.6) is the same as the one given by the formula (3.12), and thus we have derived the advection-diffusion equation (3.13).

B Proof of the entropy production inequality

Let us prove that any of the functions (6.40) is a Lyapunov function of the space-independent kinetic equation (2.7). The entropy production due to the relaxation system,

$$\partial_t f_i = -\frac{1}{\tau}(f_i - \rho \varphi_i(v)),$$

reads:

$$\sigma_H(f) = \sum_{i=1}^{n_d} \frac{\partial H}{\partial f_i}\partial_t f_i = -\frac{1}{\tau}\sum_{i=1}^{n_d} h'\left(\frac{f_i}{\varphi_i(v)}\right)[f_i - \rho\varphi_i(v)]\,. \tag{B.1}$$

Now we need to show that the latter expression is non-positive. We transform it as:

$$\begin{aligned}\sigma_H(f) &= -\frac{1}{\tau}\sum_{i=1}^{n_d} h'\left(\frac{f_i}{\varphi_i(v)}\right)\varphi_i(v)\left[\frac{f_i}{\varphi_i(v)}-\rho\right] \\ &= -\frac{1}{\tau}\sum_{i=1}^{n_d} h'\left(\frac{f_i}{\varphi_i(v)}\right)\varphi_i(v)\left[\frac{f_i}{\varphi_i(v)}-\rho\right]+\frac{1}{\tau}\sum_{i=1}^{n_d} h'(\rho)\varphi_i(v)\left[\frac{f_i}{\varphi_i(v)}-\rho\right] \\ &= -\frac{1}{\tau}\sum_{i=1}^{n_d}\varphi_i(v)\left[h'\left(\frac{f_i}{\varphi_i(v)}\right)-h'(\rho)\right]\left[\frac{f_i}{\varphi_i(v)}-\rho\right]. \qquad \text{(B.2)}\end{aligned}$$

In the second line of this computation we have again ‘subtracted zero’ due to mass conservation:

$$0=-\frac{1}{\tau}h'(\rho)\sum_{i=1}^{n_d}(f_i-\rho\varphi_i(v)).$$

Introducing $z_i = f_i/\varphi_i(v)$, we see that $\sigma_H(f)$ (B.2) is proportional to the sum of functions A,

$$A(z_i,\rho)=[h'(z_i)-h'(\rho)][z_i-\rho], \qquad \text{(B.3)}$$

with positive weights $\varphi_i(v)$. We now make use of the strict convexity of the function h which states that its derivative $h'(z)$ is monotonic ($h'' > 0$). This means that if $z > y$, then $h'(z) > h'(y)$. This implies in (B.3):

$$\begin{array}{ll} h'(z_i)-h'(\rho)>0,\ z_i-\rho>0, & \text{if } z_i>\rho, \\ h'(z_i)-h'(\rho)<0,\ z_i-\rho<0, & \text{if } z_i<\rho. \end{array} \qquad \text{(B.4)}$$

Hence, $A(z_i,\rho)$ is the product of two expressions of the same sign for any z_i and ρ, and thus A is positive except for the case $z_i = \rho$ (that is, except for the equilibrium, $f_i = \rho\varphi_i(v)$, in which case it is equal to zero). Thus, $\sigma_H \leq 0$ for any strictly convex function h, and, if the solution of the kinetic equation belongs entirely to the domain of definition of H, function $H(f(t))$ monotonically decreases along this solution, $dH/dt = \sigma_H \leq 0$.

References

[1] S. Succi, The Lattice Boltzmann Equation for Fluid Dynamics and Beyond, Oxford University Press, Oxford, 2001.

[2] F.J. Higuera, S. Succi, R. Benzi, Europhys. Lett. 9, 345-349 (1989).

[3] Y.H. Qian, D. d’Humieres, P. Lallemand, Europhys. Lett. 17 (1992) 479-484.

[4] X. Shan, X. He, Phys. Rev. Lett. 80, 65 (1998).

[5] H. Chen, S. Kandasamy, S. Orszag, R. Shock, R. Succi, V. Yakhot, Science 301 (2003) 633-636.

[6] R. Benzi, S. Succi, M. Vergassola, Phys. Rep. 222 (1992) 145-197.

[7] S. Chen, G.D. Doolen, Annu. Rev. Fluid Mech. 30 (1998) 329.

[8] S. Succi, I.V. Karlin, H. Chen, Rev. Mod. Phys. 74 (2002) 1203-1220.

[9] I.V. Karlin, A. Gorban, S. Succi, V. Boffi, Phys. Rev. Lett. 81 (1998) 6-9.

[10] I.V. Karlin, A. Ferrante, Öttinger, H. C., Europhys. Lett. 47 (1999) 182-188.

[11] S. Ansumali, I.V. Karlin, Öttinger, H. C., Europhys. Lett. 63 (2003) 798-804.

[12] S. Ansumali, I.V. Karlin, Öttinger, H. C., Phys. Rev. Lett. 94 (2005) 080602.

[13] S. Ansumali, I.V. Karlin, Phys. Rev. Lett. 95 (2005) 260605.

[14] B.M. Boghosian, J. Yepez, P.V. Coveney, A.J. Wagner, Proc. Roy. Soc. Lond. A457 (2001) 717-766.

[15] S. Ansumali, I.V. Karlin, Phys. Rev. E 66 (2002) 026311.

[16] S. Ansumali, S.S. Chikatamarla, C. Frouzakis, K. Boulouchos, Int. J. Mod. Phys. C 15 (2004) 435-445 .

[17] P.M. Gresho, R.L. Sani, Incompressible Flow and the Finite Element Method, Wiley, NY, 1998.

[18] P.L. Bhatnagar, E.P. Gross, M. Krook, Phys. Rev. 94 (1954) 511-525.

[19] A.N. Gorban, I.V. Karlin, Invariant Manifolds for Physical and Chemical Kinetics, LNP 660 Springer, Berlin, 2005.

[20] S. Chapman, T.G. Cowling, The Mathematical Theory of Non-Uniform Gases, Third edition, Cambridge University Press, Cambridge, 1970.

[21] S. Ansumali, I.V. Karlin, Phys. Rev. E 62(6) (2000) 7999-8003.

[22] X. He, S. Chen, G.D. Doolen, J. Comput. Phys. 146 (1998) 282-300.

[23] A. Harten, B. Engquist, S. Osher, S.R. Chakravarthy, J. Comput. Phys. 71 (1987) 231-303.

[24] S. Chikatamarla, S. Ansumali, I.V. Karlin, Preprint (2006).

[25] Ansumali, S., Karlin, I. V., J. Stat. Phys. 107 (2002) 291-308.