Chapter 6
Forces

Abstract After reading this chapter, you will be able to add forces to lattice Boltzmann simulations while retaining their accuracy. You will know how a forcing scheme can be derived by including forces in the derivation of the lattice Boltzmann equation, though you will also know that there are a number of other forcing schemes available. You will understand how to investigate forcing models and their errors through the Chapman-Enskog analysis, and how initial and boundary conditions can be affected by the presence of forces.

Forces play an important role in many hydrodynamic problems (Sect. 6.1). Therefore, a proper discussion of force implementation in the LB algorithm is essential. Section 6.2 contains quick start instructions to implement an LB algorithm with forces. In Sect. 6.3 we show how to extend the force-free LBE derivation (i.e. the LBE derived in Chap. 3) to also reproduce a macroscopic body force at the hydrodynamic level. This derivation is based on the same discretisation steps (velocity followed by space-time) that are also used for the force-free LBE. Section 6.4 contains an overview of existing forcing schemes and a discussion of their differences and similarities. We will see that many of those schemes are equivalent if higher-order terms are neglected. In Sect. 6.5 we extend the Chapman-Enskog analysis to situations with forces to point out the detailed links between the LBE and the macroscopic PDEs it approximates. Furthermore, we investigate the errors associated with the selected forcing schemes. We analyse the influence of the forcing term on simulation initialisation and two types of boundary conditions in Sect. 6.6. In particular, we show how the bounce-back and the non-equilibrium bounce-back methods account for the presence of a force. Finally, in Sect. 6.7 we use a simple Poiseuille flow to demonstrate the previous theoretical elements in benchmark simulations.

T. Krüger et al., *The Lattice Boltzmann Method*, Graduate Texts in Physics,
DOI 10.1007/978-3-319-44649-3_6

6.1 Motivation and Background

Forces play a central role in many hydrodynamic problems. A prominent example is the gravitational acceleration $\boldsymbol{g}$ which can be cast into a force *density* $\boldsymbol{F}_g$ by multiplying it with the fluid density ρ:

$$\boldsymbol{F}_g = \rho \boldsymbol{g}. \tag{6.1}$$

In fact, in hydrodynamics we will mostly encounter force densities rather than forces since the momentum equation is a PDE for the momentum *density*. Forces are obtained by integrating surface stresses or bulk force densities. Mathematically, a force (density) is a momentum (density) source term, as can be seen from the Cauchy equation in (1.57).

Gravity leads to a number of effects which LBM can successfully simulate. If two fluids with different densities are mixed or if the temperature in a fluid is non-homogeneous, density gradients in the gravitational field lead to buoyancy effects and phenomena like the *Rayleigh-Bénard instability* [1] (cf. Sect. 8.4.1) or the *Rayleigh-Taylor instability* [2]. In the Rayleigh-Bénard instability, which is essential in studies of heat transfer, convection patterns develop when warmed fluid rises from a hot surface and falls after cooling. The Rayleigh-Taylor instability can occur when a layer of denser fluid descends as lower-density fluid below it rises. Gravity waves at a free water surface are another example [3].

Apart from gravity [4], there are several other physical problems where forces are important. Fluids in rotating reference frames are subject to radial and Coriolis forces [5–7]. Charged or magnetic particles immersed in a fluid exert forces on each other, and they may also be forced by external electromagnetic fields. This is particularly important for modelling the effects of external electric fields on regions of unbalanced charges (the electrical double layer, EDL) in electrolytes near a charged solid surface or liquid-liquid interface [8–11].

In incompressible flows, the driving mechanism of the pressure gradient field may be equivalently described by any *divergence-free* body force [12]. There exist cases where the problem physics specify pressure gradients, but where it is convenient to replace these with forces [12–14]. One reason for this is that the LB method may lose accuracy when solving pressure fields due to compressibility errors (cf. Sect. 4.5). While this change is possible in arbitrarily complex flow geometries, the task of finding an equivalent driving force field is only trivial in periodic flow configurations. This is often explored in LB simulations of porous media flows [14].

We will see in Chap. 9 that forces are also commonly used to model multi-phase or multi-component flows, although a mathematical description of these phenomena is usually based on the stress tensor. Furthermore, some algorithms for fluid-structure interactions, e.g. the immersed boundary method, rely on forces mimicking boundary conditions. We will discuss this in Sect. 11.4.

6.2 LBM with Forces in a Nutshell

We summarise the most important information about the implementation of forces in the bulk LBM and what a complete time step with forces looks like.

Assuming the BGK collision operator, we can write the order of operations in a single time step including forces, also illustrated in Fig. 6.1, in the following way:

1. Determine the force density $\boldsymbol{F}$ for the time step (e.g. gravity).
2. Compute the fluid density and velocity from

$$\rho = \sum_i f_i, \quad \boldsymbol{u} = \frac{1}{\rho}\sum_i f_i \boldsymbol{c}_i + \frac{\boldsymbol{F}\Delta t}{2\rho}. \tag{6.2}$$

3. Compute the equilibrium populations $f_i^{\mathrm{eq}}(\rho, \boldsymbol{u})$ to construct the collision operator

$$\Omega_i = -\frac{1}{\tau}\left(f_i - f_i^{\mathrm{eq}}\right). \tag{6.3}$$

4. If desired, output the macroscopic quantities. If required, the deviatoric stress is calculated as

$$\sigma_{\alpha\beta} \approx -\left(1 - \frac{\Delta t}{2\tau}\right)\sum_i f_i^{\mathrm{neq}} c_{i\alpha} c_{i\beta} - \frac{\Delta t}{2}\left(1 - \frac{\Delta t}{2\tau}\right)\left(F_\alpha u_\beta + u_\alpha F_\beta\right). \tag{6.4}$$

5. Compute the source term

$$S_i = \left(1 - \frac{\Delta t}{2\tau}\right) w_i \left(\frac{c_{i\alpha}}{c_\mathrm{s}^2} + \frac{\left(c_{i\alpha}c_{i\beta} - c_\mathrm{s}^2\delta_{\alpha\beta}\right)u_\beta}{c_\mathrm{s}^4}\right) F_\alpha, \tag{6.5}$$

where the source S_i and forcing F_i terms are related as $S_i = (1 - \frac{1}{2\tau})F_i$

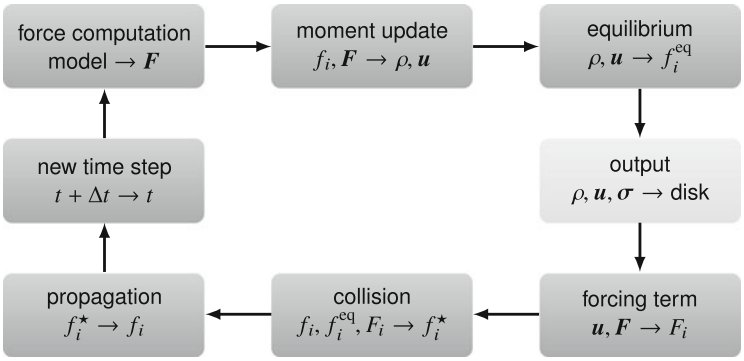

Fig. 6.1 An overview of one cycle of the LB algorithm, considering forces but not boundary conditions. The *light grey box* shows the optional output sub-step

6. Apply collision and source to find the post-collision populations:

$$f_i^\star = f_i + (\Omega_i + S_i)\Delta t. \tag{6.6}$$

7. Propagate populations.
8. Increment the time step and go back to step 1.

There are a few important remarks:

- The form of the force $\boldsymbol{F}$ depends on the underlying physics and is not itself given by the LB algorithm. Gravity is the simplest example.
- The velocity $\boldsymbol{u}$ in (6.2) contains the so-called *half-force correction*. This velocity $\boldsymbol{u}$ enters the equilibrium distributions and is also the macroscopic fluid velocity solving the Navier-Stokes equation. Using the *bare* velocity $\boldsymbol{u}^\star = \sum_i f_i \boldsymbol{c}_i/\rho$ would lead to first-order rather than second-order space-time accuracy (Sect. 6.3.2). The velocity $\boldsymbol{u}$ can be interpreted as the average velocity during the time step, i.e. the average of pre- and post-collision values.
- The forcing scheme presented here is based on a Hermite expansion (Sect. 6.3.1) and is the same as proposed by Guo et al. [15]. There are alternative ways to include forces, as discussed in Sect. 6.4.
- Any cyclic permutation of the above steps is permitted, as long as the simulation is properly initialised.

6.3 Discretisation

In Chap. 3 we have shown how to derive the LBE from the continuous Boltzmann equation in the absence of forces. Here we will revisit that derivation from Sect. 3.4 and Sect. 3.5, now highlighting the required steps to include forces. The two main steps are the discretisation in velocity space (Sect. 6.3.1) and the space-time discretisation (Sect. 6.3.2).

6.3.1 Discretisation in Velocity Space

Let us briefly recall the velocity space discretisation of the (force-free) Boltzmann equation as explained in Sect. 3.4. The objective was to reduce the continuous velocity space $\boldsymbol{\xi}$ to a finite set of discrete velocities $\boldsymbol{c}_i$ while preserving the model's ability to capture the desired macroscopic physics *via* velocity moments.

A natural and systematic approach is to represent the equilibrium distribution function f^{eq} as a truncated Hermite expansion. This permits an *exact* evaluation of macroscopic quantities (e.g. density and velocity) through a Gauss-Hermite quadrature. This procedure led to two important results: (i) a polynomial representation of f^{eq} in velocity space (cf. Sect. 3.4.5) and (ii) the description of particles' motion

through a discrete velocity set (cf. Sect. 3.4.7). The question we aim to answer here is: *what is the equivalent polynomial representation in velocity space of the forcing term in the Boltzmann equation?* The following explanation is based on [16, 17].

Let us recall the continuous Boltzmann equation with a forcing term:

$$\frac{\partial f}{\partial t} + \xi_\alpha \frac{\partial f}{\partial x_\alpha} + \frac{F_\alpha}{\rho} \frac{\partial f}{\partial \xi_\alpha} = \Omega(f). \tag{6.7}$$

Our goal is to find the discrete velocity structure of the forcing term F_α which aligns with the velocity space discretisation of f^{eq} in Sect. 3.4.5. An evident problem is that F_α, contrarily to f^{eq}, does not appear as isolated term in (6.7). Rather, to deal with F_α we have to discretise the full term $\frac{F_\alpha}{\rho}\frac{\partial f}{\partial \xi_\alpha}$. Its discretisation in velocity space is simple if we keep the following two mathematical results in mind:

1. The Hermite series expansion of the distribution function $f(\boldsymbol{\xi})$ is

$$f(\boldsymbol{x}, \boldsymbol{\xi}, t) \approx \omega(\boldsymbol{\xi}) \sum_{n=0}^{N} \frac{1}{n!} \boldsymbol{a}^{(n)}(\boldsymbol{x}, t) \cdot \boldsymbol{H}^{(n)}(\boldsymbol{\xi}). \tag{6.8}$$

2. The derivative property of Hermite polynomials reads

$$\omega(\boldsymbol{\xi}) \boldsymbol{H}^{(n)} = (-1)^n \nabla_{\boldsymbol{\xi}}^n \omega(\boldsymbol{\xi}). \tag{6.9}$$

With their help we can rewrite the Hermite expansion of $f(\boldsymbol{\xi}_i)$ as follows:

$$f \approx \sum_{n=0}^{N} \frac{(-1)^n}{n!} \boldsymbol{a}^{(n)} \cdot \nabla_{\boldsymbol{\xi}}^n \omega. \tag{6.10}$$

This representation allows us to simplify the forcing contribution in (6.7):

$$\begin{aligned} \frac{\boldsymbol{F}}{\rho} \cdot \nabla_{\boldsymbol{\xi}} f &\approx \frac{\boldsymbol{F}}{\rho} \cdot \sum_{n=0}^{N} \frac{(-1)^n}{n!} \boldsymbol{a}^{(n)} \cdot \nabla_{\boldsymbol{\xi}}^{n+1} \omega \\ &\approx -\frac{\boldsymbol{F}}{\rho} \cdot \omega \sum_{n=1}^{N} \frac{1}{n!} n \boldsymbol{a}^{(n-1)} \cdot \boldsymbol{H}^{(n)}. \end{aligned} \tag{6.11}$$

The discretisation in velocity space can now be performed directly, by replacing the continuous $\boldsymbol{\xi}$ by a discrete set of $\boldsymbol{c}_i$. We rescale the velocities according to $\boldsymbol{c}_i = \boldsymbol{\xi}_i/\sqrt{3}$ and then renormalise the result by the lattice weights w_i. Recalling Sect. 3.4.5, this is similar to what we did in the construction of f^{eq}. Based on this procedure, the discrete form of the forcing term becomes:

$$F_i(\boldsymbol{x}, t) = -\frac{w_i}{\omega(\boldsymbol{\xi})} \frac{\boldsymbol{F}}{\rho} \cdot \nabla_{\boldsymbol{\xi}} f \Big|_{\boldsymbol{\xi} \to \sqrt{3}\boldsymbol{c}_i}, \tag{6.12}$$

with the right-hand side given in (6.11). This way, we can write the discrete velocity Boltzmann equation with a forcing term similarly to (3.58):

$$\partial_t f_i + c_{i\alpha}\partial_\alpha f_i = \Omega_i + F_i, \quad i = 0, \ldots, q-1. \tag{6.13}$$

The **truncation of the forcing term** up to second velocity order ($N = 2$), corresponding to the expansion of f^{eq}, reads

$$F_i = w_i \left(\frac{c_{i\alpha}}{c_\mathrm{s}^2} + \frac{\left(c_{i\alpha}c_{i\beta} - c_\mathrm{s}^2\delta_{\alpha\beta}\right)u_\beta}{c_\mathrm{s}^4} \right) F_\alpha. \tag{6.14}$$

Its first three velocity moments are

$$\sum_i F_i = 0, \tag{6.15a}$$

$$\sum_i F_i c_{i\alpha} = F_\alpha, \tag{6.15b}$$

$$\sum_i F_i c_{i\alpha} c_{i\beta} = F_\alpha u_\beta + u_\alpha F_\beta. \tag{6.15c}$$

Exercise 6.1 Write down the explicit form of the forcing term F_i in (6.14) for the velocity sets D1Q3 (cf. Table 3.2) and D2Q9 (cf. Table 3.3). Compare the results obtained for F_i with the structure of f_i^{eq} expressed by (3.64) and (3.65), respectively.

The zeroth-order moment, (6.15a), denotes a mass source; it is zero in the present situation. The first-order moment, (6.15b), is a momentum source; it appears as a body force in the NSE. Finally, the second-order moment, (6.15c), is an energy source describing the power flux the body force exerts on the fluid [18].

The role of the second-order moment, (6.15c), is subtle. Its appearance, at first glance, may seem surprising as LBM is typically built upon an isothermal assumption. However, the (weakly) compressible regime reproduced by LBM with standard equilibrium still preserves a (weak) link to energy transport, although an isothermal one [19]. The purpose of (6.15c) is simply to remove the undesirable footprint left by this connection on the momentum equations. Otherwise, a spurious term, given by $F_\alpha u_\beta + u_\alpha F_\beta$, appears at the viscous stress level [15, 18, 20]. We explain this error source in more depth in Sect. 6.5.1.

On the other hand, in the incompressible regime the energy transport is totally decoupled from the momentum equation [21, 22]. As pointed out in Sect. 4.3.2, for steady problems the LBM with the incompressible equilibrium can reproduce the true incompressible NSE. Therefore, the cancelling of errors with link to compressibility is not required in this case, where we would expect the condition $\sum_i F_i c_{i\alpha} c_{i\beta} = 0$ instead. According to the force discretisation process above, this is equivalent to saying that F_i should be expanded only up to first order in velocity space:

$$F_i = w_i \frac{c_{i\alpha}}{c_s^2} F_\alpha. \tag{6.16}$$

This duality in the expansion order of F_i is explained in more detail in [5, 23].

6.3.2 Discretisation in Space and Time

We discussed the space-time discretisation of the force-free Boltzmann equation in Sect. 3.5. The idea was to replace the continuous space and time derivatives in the discrete-velocity Boltzmann equation, (6.13), by difference operators with discrete space and time steps (Δx and Δt). In the standard LBM, these discretisation steps are linked to the velocity space discretisation to ensure that populations f_i, travelling with discrete velocities $\boldsymbol{c}_i$, always reach neighbouring lattice sites within one time step Δt.

We seek a similar result in the presence of forces. The task consists of two parts [24–26]:

1. Advection, the left-hand side of (6.13), is identical to the force-free case (cf. Sect. 3.5). By applying the method of characteristics, i.e. defining $f_i = f_i(\boldsymbol{x}(\zeta), t(\zeta))$, where ζ parametrises a trajectory in space and time, the propagation step is *exact*, without any approximation:

$$\int_t^{t+\Delta t} \frac{\mathrm{d}f_i}{\mathrm{d}\zeta}\,\mathrm{d}\zeta = f_i(\boldsymbol{x} + \boldsymbol{c}_i \Delta t, t + \Delta t) - f_i(\boldsymbol{x}, t). \tag{6.17}$$

2. The only approximation appears in the treatment of the right-hand side of (6.13), collision, which now includes the forcing term F_i:

$$\int_t^{t+\Delta t} (\Omega_i + F_i)\,\mathrm{d}\zeta. \tag{6.18}$$

We can evaluate this integral in different ways [26]. We will now discuss two approximations, as already described for the force-free case in Sect. 3.5.

6.3.2.1 First-Order Integration

The least accurate procedure employs a rectangular discretisation. Here, the integral of collision and forcing terms is approximated by just one point:

$$\int_t^{t+\Delta t} (\Omega_i + F_i)\, \mathrm{d}\zeta = \left[\Omega_i(\boldsymbol{x}, t) + F_i(\boldsymbol{x}, t)\right] \Delta t + O(\Delta t^2). \tag{6.19}$$

Using this first-order approximation and the BGK collision operator, the LBE with a force assumes a form where all terms on the right-hand side are evaluated at $(\boldsymbol{x}, t)$:

$$f_i(\boldsymbol{x} + \boldsymbol{c}_i \Delta t, t + \Delta t) - f_i(\boldsymbol{x}, t) = -\frac{\Delta t}{\tau}\left(f_i - f_i^{\mathrm{eq}}\right) + F_i \Delta t. \tag{6.20}$$

Apart from the inclusion of $F_i \Delta t$ in (6.20), everything else is exactly as the unforced case in Chap. 3.

While being fully explicit, this scheme is only first-order accurate in time. In the absence of forces, this is not harmful since we can still obtain second-order accuracy providing the $\Delta t/2$ shift is considered in the viscosity-relaxation relation [26]: $\nu = c_\mathrm{s}^2(\tau - \frac{\Delta t}{2})$ instead of $\nu = c_\mathrm{s}^2 \tau$. The reason for this accuracy improvement is that both the "physical" viscous term and its leading-order error have the same functional form; the latter can be absorbed as a "physical" contribution by redefining the viscosity.

This "trick" does not work in the presence of forces, though. Hence, the first-order accuracy inevitably leads to macroscopic solutions corrupted by discrete lattice artefacts [4, 15]. We show their mathematical form in Sect. 6.5.1 and illustrate their quantitative effects in Sect. 6.7. We can eliminate these undesired artefacts by employing a second-order space-time discretisation.

6.3.2.2 Second-Order Integration

The trapezoidal discretisation is more accurate than the rectangular discretisation:

$$\int_t^{t+\Delta t} (\Omega_i + F_i)\, \mathrm{d}\zeta = \left(\frac{\Omega_i(\boldsymbol{x}, t) + \Omega_i(\boldsymbol{x} + \boldsymbol{c}_i \Delta t, t + \Delta t)}{2} + \frac{F_i(\boldsymbol{x}, t) + F_i(\boldsymbol{x} + \boldsymbol{c}_i \Delta t, t + \Delta t)}{2} \right) \Delta t + O(\Delta t^3). \tag{6.21}$$

However, we obtain second-order accuracy at the expense of a time-implicit scheme. Fortunately, this is not a problem since, as explained in Sect. 3.5, we can recover the explicit form by introducing a smart change of variables [19, 27]:

$$\bar{f}_i = f_i - \frac{(\Omega_i + F_i)\Delta t}{2}. \tag{6.22}$$

Using (6.22) and some simple algebra, the LBE for $\bar{f}_i$ takes the familiar form

$$\bar{f}_i(\boldsymbol{x} + \boldsymbol{c}_i\Delta t, t + \Delta t) - \bar{f}_i(\boldsymbol{x}, t) = \left[\Omega_i(\boldsymbol{x}, t) + F_i(\boldsymbol{x}, t)\right]\Delta t. \tag{6.23}$$

With the BGK collision operator this simplifies to

$$\bar{f}_i(\boldsymbol{x} + \boldsymbol{c}_i\Delta t, t + \Delta t) - \bar{f}_i(\boldsymbol{x}, t) = -\frac{\Delta t}{\tau + \Delta t/2}\left(\bar{f}_i - f_i^{\mathrm{eq}} - \tau F_i\right) \tag{6.24}$$

where, once again, all terms on the right-hand side are given at $(\boldsymbol{x}, t)$. The extension to other collision operators is straightforward (Sect. 10.5).

The **second-order accurate discretisation of the LBGK equation with forcing term** reads

$$\bar{f}_i(\boldsymbol{x} + \boldsymbol{c}_i\Delta t, t + \Delta t) - \bar{f}_i(\boldsymbol{x}, t) = -\frac{\Delta t}{\bar{\tau}}\left(\bar{f}_i - f_i^{\mathrm{eq}}\right) + \left(1 - \frac{\Delta t}{2\bar{\tau}}\right)F_i\Delta t \tag{6.25}$$

with a redefined relaxation parameter $\bar{\tau} = \tau + \Delta t/2$. Based on the new variable $\bar{f}_i$, the leading macroscopic moments are

$$\rho = \sum_i \bar{f}_i + \frac{\Delta t}{2}\sum_i F_i, \tag{6.26a}$$

$$\rho\boldsymbol{u} = \sum_i \bar{f}_i\boldsymbol{c}_i + \frac{\Delta t}{2}\sum_i F_i\boldsymbol{c}_{i\alpha}, \tag{6.26b}$$

$$\boldsymbol{\Pi} = \left(1 - \frac{\Delta t}{2\bar{\tau}}\right)\sum_i \bar{f}_i\boldsymbol{c}_i\boldsymbol{c}_i + \frac{\Delta t}{2\bar{\tau}}\sum_i f_i^{\mathrm{eq}}\boldsymbol{c}_i\boldsymbol{c}_i + \frac{\Delta t}{2\bar{\tau}}\left(1 - \frac{\Delta t}{2\bar{\tau}}\right)\sum_i F_i\boldsymbol{c}_i\boldsymbol{c}_i. \tag{6.26c}$$

In most cases, the notation of the redefined variables is dropped for convenience and f_i and τ are written instead of $\bar{f}_i$ and $\bar{\tau}$. The equilibrium populations f_i^{eq} have the same functional form as before. However, the velocity entering $f_i^{\mathrm{eq}}(\rho, \boldsymbol{u})$ is now given by (6.26b). The redefinition of the velocity in (6.26b) can be interpreted as averaging the velocity before and after forcing [29, 30]. If F_i is chosen to incorporate non-zero mass sources in addition to forces, the density entering $f_i^{\mathrm{eq}}(\rho, \boldsymbol{u})$ must also be redefined according to (6.26a) [28]. Sometimes, for convenience, the outcome from the space-time discretisation of the forcing term, as given in (6.25), is shortened to a source term S_i notation, with the two related as $S_i = (1 - \frac{1}{2\tau})F_i$.

Equation (6.26b) can lead to difficulties when $\boldsymbol{F}$ depends on $\boldsymbol{u}$, e.g. in Brinkman models [31–34] or Coriolis forces [5–7, 35]. Such a velocity-dependent force leads to an implicit form of (6.26b) in $\boldsymbol{u}$. For linear relations, $\boldsymbol{F} \propto \boldsymbol{u}$, and other analytically invertible dependencies we can easily solve (6.26b) for $\boldsymbol{u}$, e.g. [5, 31]. In more general cases, however, $\boldsymbol{u}$ has to be found numerically, e.g. [7, 35].

6.4 Alternative Forcing Schemes

In Sect. 6.3, we have shown how the forcing scheme can be constructed through a systematic procedure consistent with the overall LBE. However, there is a flood of articles about other LB forcing schemes.

This section aims at clarifying differences and similarities among some of the most popular forcing schemes. After recollecting important consequences of the presence of a force in Sect. 6.4.1, we show a few alternative forcing schemes in Sect. 6.4.2. We focus on results rather than on those lengthy calculations that can be found in the cited literature. The articles by Guo et al. [15] and Huang et al. [36] provide derivations and more detailed discussions. Also helpful in this context is the work by Ginzburg et al. [37] that discusses different, yet equivalent, ways of introducing the force in the LB equation.

6.4.1 General Observations

Based on the second-order velocity and space-time discretisations, the LBE with a force can be expressed as

$$f_i(\boldsymbol{x} + \boldsymbol{c}_i\Delta t, t + \Delta t) - f_i(\boldsymbol{x}, t) = \left[\Omega_i(\boldsymbol{x}, t) + S_i(\boldsymbol{x}, t)\right]\Delta t \tag{6.27}$$

where Ω_i is the BGK collision operator and $S_i = (1 - \frac{1}{2\tau})F_i$ denotes a source, with the forcing F_i given by (6.14). Guo et al. [15] derived the same result following an approach different from that in Sect. 6.3. Therefore, this scheme is often called *Guo forcing*.

It is important that the fluid velocity in the presence of a force is redefined to guarantee the second-order space-time accuracy (Sect. 6.3.2):

$$\boldsymbol{u} = \frac{1}{\rho}\sum_i \boldsymbol{c}_i f_i + \frac{\boldsymbol{F}\Delta t}{2\rho}. \tag{6.28}$$

This velocity also enters the equilibrium populations $f_i^{\mathrm{eq}} = f_i^{\mathrm{eq}}(\rho, \boldsymbol{u})$ and therefore the BGK collision operator $\Omega_i = -(f_i - f_i^{\mathrm{eq}})/\tau$. Thus we can say that the fluid velocity in (6.28) and the equilibrium velocity $\boldsymbol{u}^{\mathrm{eq}}$ (i.e. the velocity entering f_i^{eq}) are the same for Guo forcing.

The complexity in the LB literature is caused by the fact that there exist different force algorithms that decompose Ω_i and S_i differently but lead to essentially the same results on the Navier-Stokes level. To generalise the forcing method, let us write

$$\boldsymbol{u}^{\mathrm{eq}} = \frac{1}{\rho}\sum_i f_i \boldsymbol{c}_i + A\frac{\boldsymbol{F}\Delta t}{\rho} \tag{6.29}$$

for the equilibrium velocity. A is a model-dependent parameter. For Guo forcing, we already know that $A = \frac{1}{2}$. Deviating from this value means that the collision operator Ω_i is modified. In turn, also the source term S_i has to be redefined to keep the sum $\Omega_i + S_i$ unchanged, at least to leading order.

Naively we can expect that we cannot distinguish forcing schemes macroscopically as long as the sum $\Omega_i + S_i$ is the same, no matter which individual forms Ω_i and S_i assume. In fact, there exist several forcing schemes for which $\Omega_i + S_i$ *nearly* has the same form as Guo forcing, only up to deviations of order F^2 or u^3. Therefore, all those methods can be considered equivalent as long as F and u are sufficiently small, which cannot always be guaranteed. Furthermore, there are other forcing schemes that result in different behaviour on the F and u^2 orders (or even worse); those methods are generally less accurate and should be avoided.

6.4.2 Forcing Schemes

Each different LB forcing scheme has a different set of expressions for A in (6.29) and the source term S_i. But not all of the proposed methods lead to acceptable hydrodynamic behaviour. In the following we collect a few selected forcing schemes that do recover the correct macroscopic behaviour. Table 6.1 provides a summary.

The **fluid velocity** needs to assume the form in (6.28), **independently** of the chosen forcing scheme. This is a pure consequence of the second-order time integration and not affected by details of the forcing scheme.

Table 6.1 Overview of accurate forcing schemes and how they modify the collision operator in (6.27), both directly and through the equilibrium velocity defined in (6.29). In any case the fluid velocity must obey (6.28) to ensure second-order time accuracy

Method	A	S_i
Guo et al. [15]	$1/2$	$\left(1-\frac{\Delta t}{2\tau}\right) w_i \left(\frac{\boldsymbol{c}_i-\boldsymbol{u}}{c_s^2} + \frac{(\boldsymbol{c}_i\cdot\boldsymbol{u})\boldsymbol{c}_i}{c_s^4}\right)\cdot\boldsymbol{F}$
Shan and Chen [38]	$\tau\Delta t$	0
He et al. [39]	$1/2$	$\left(1-\frac{\Delta t}{2\tau}\right)\frac{f_i^{\mathrm{eq}}}{\rho}\frac{\boldsymbol{c}_i-\boldsymbol{u}}{c_s^2}\cdot\boldsymbol{F}$
Kupershtokh et al. [40]	0	$f_i^{\mathrm{eq}}(\rho,\boldsymbol{u}^\star+\Delta\boldsymbol{u}) - f_i^{\mathrm{eq}}(\rho,\boldsymbol{u}^\star)$

All forcing schemes in Table 6.1 are equivalent up to terms of order F^2 or u^3 [36].[1] In the limit of small Mach number and small forces, all these methods yield basically the same results. The situation is different for multi-phase flows where forces in the vicinity of fluid-fluid interfaces can become large so that terms $\propto F^2$ (and also $\propto \nabla^2 \boldsymbol{F}$) are important.[2] We will not discuss the choice of the forcing scheme in the context of multi-phase flows here and refer to Sect. 9.3.2 and [36, 41] instead.

6.4.2.1 Guo et al. (2002)

This method is the same as derived in Sect. 6.3. Based on the Chapman-Enskog analysis, Guo et al. [15] performed a thorough analysis of the lattice effects in the presence of a force. In their article, which is an extension of previous work by Ladd and Verberg [20], the parameters assume the values $A = 1/2$ and $S_i = (1 - \frac{1}{2\tau})F_i$ with F_i as in (6.14). Guo et al. [15] showed that these choices remove undesired derivatives in the continuity and momentum equation due to time discretisation artefacts (cf. Sect. 6.3.2). In particular, $A = 0$ would lead to a term $\propto \nabla \cdot \boldsymbol{F}$ in the continuity equation and another term $\propto \nabla \cdot (\boldsymbol{uF} + \boldsymbol{Fu})$ in the momentum equation (cf. Sect. 6.5.2).

6.4.2.2 Shan and Chen (1993, 1994)

Shan and Chen [38] proposed $A = \tau/\Delta t$ and $S_i = 0$. Although their motivation was the simulation of multi-phase fluids (cf. Chap. 9), Shan and Chen's method is applicable to single-phase fluids as well.

6.4.2.3 He et al. (1998)

The essential idea of He et al. [39] was to approximate the forcing term in the kinetic equation by assuming a situation close to equilibrium:

$$\boldsymbol{F} \cdot \nabla_{\boldsymbol{c}} f \approx \boldsymbol{F} \cdot \nabla_{\boldsymbol{c}} f^{\mathrm{eq}} = -\boldsymbol{F} \cdot \frac{\boldsymbol{c} - \boldsymbol{u}}{c_{\mathrm{s}}^2} f^{\mathrm{eq}}. \tag{6.30}$$

[1] Showing that these forcing schemes are equivalent to leading order is straightforward but involves lengthy calculations. We will not delve into details here and refer to [36] for a more qualitative analysis.

[2] Also in Brinkman and Coriolis force models, where $\boldsymbol{F} \propto \boldsymbol{u}$, the error term $\propto \nabla^2 \boldsymbol{F}$ is important [33, 34].

In the end this leads to $A = \frac{1}{2}$ and

$$S_i = \left(1 - \frac{\Delta t}{2\tau}\right) \frac{f_i^{\mathrm{eq}}}{\rho} \frac{\boldsymbol{c}_i - \boldsymbol{u}}{c_\mathrm{s}^2} \cdot \boldsymbol{F}. \tag{6.31}$$

6.4.2.4 Kupershtokh (2004)

Kupershtokh [42] proposed a simple forcing method based on kinetic theory, the so-called *exact difference method*. The idea is to include the force density $\boldsymbol{F}$ in such a way that it merely shifts f_i in velocity space. As a consequence, $A = 0$ and

$$S_i = f_i^{\mathrm{eq}}(\rho, \boldsymbol{u}^\star + \Delta \boldsymbol{u}) - f_i^{\mathrm{eq}}(\rho, \boldsymbol{u}^\star) \tag{6.32}$$

where $\boldsymbol{u}^\star = \sum_i f_i \boldsymbol{c}_i / \rho$ and $\Delta \boldsymbol{u} = \boldsymbol{F} \Delta t / \rho$. This essentially means that the equilibrium for a velocity $\boldsymbol{u}^\star$ is directly replaced by the equilibrium for a velocity $\boldsymbol{u}^\star + \Delta \boldsymbol{u}$. In particular, this scheme ensures that an equilibrium distribution remains in equilibrium upon the action of the force, independently of the chosen value of τ.

6.4.2.5 Other, Less Accurate Approaches

There exist several other forcing schemes in the LB literature. Guo et al. [15] reviewed a series of approaches [4, 16, 43, 44] and showed that all of them lead to certain unphysical terms in the continuity or momentum equations of the *weakly compressible* NSE. In other words, those forcing schemes have additional error terms which are more significant than u^3 or F^2.

However, the situation changes when modelling *steady incompressible* hydrodynamics. In this case, the most accurate forcing scheme is no longer Guo's [15], but the scheme proposed by Buick and Greated [4]. We will discuss the reason for this variation in Sect. 6.5.2; see also [5, 23].

Finally, we would like to emphasise that under some circumstances some of these models may still be appropriate choices, for example if the force density $\boldsymbol{F}$ is constant. We illustrate this case in Sect. 6.7 through a numerical example. Still, we strongly recommend to implement one of the generally more accurate models mentioned above since they are usually more accurate when boundary conditions are involved. We will demonstrate this analytically in Sect. 6.6 and numerically in Sect. 6.7.

Concluding, there exist **several different forcing schemes**. Many of these schemes (i.e. Guo, Shan-Chen, He, Kupershtokh) are equivalent up to higher-order terms (u^3 or F^2). Their differences are negligible as long as forces and their gradients are small (e.g. in the case of gravity). Other forcing schemes, however, lead to additional error terms on the Navier-Stokes level.

6.5 Chapman-Enskog and Error Analysis in the Presence of Forces

We look at the macroscopic behaviour of forces in the LBE. First we revisit the Chapman-Enskog analysis (cf. Sect. 4.1) and extend it to situations with forces (Sect. 6.5.1). Based on this analysis, we discuss the structure of errors created at hydrodynamic level due to incorrectly chosen LB force models (Sect. 6.5.2).

6.5.1 Chapman-Enskog Analysis with Forces

The Chapman-Enskog analysis (Sect. 4.1) reveals the consistency between the mesoscopic LBM and the macroscopic NSE. We will now extend the Chapman-Enskog analysis to situations with forces.

Historically, the Chapman-Enskog analysis applied to the forced LBM was pioneered in [45, 46]. Later, a number of authors [16, 20, 39, 42, 44] extended its formulation to include second-order terms, as given by (6.14). A subsequent improvement [4, 15] showed the necessity of correcting discrete lattice effects. These effects can be corrected in an *a priori* fashion through a systematic second-order discretisation of the LBE (Sect. 6.3.2) [24–26]. Even today, the study of a "clean" inclusion of forces in the LBE remains an active research topic involving, for example, perturbation (Chapman-Enskog) analyses [5, 23, 28, 29] or exact solutions of the LBE [32–34, 47].

The Chapman-Enskog analysis of the forced LBE is similar to the force-free case in Sect. 4.1. The difference is that now we are working with (6.25) as evolution equation, together with (6.26) for the velocity moments. Hence, the first question we need to answer is: what should be the expansion order of the forcing term F_i?

> In order to be consistent with the remaining terms in the LBE, the **forcing term** must **scale** as $F_i = O(\epsilon)$ [4]. Therefore, we should at least have $F_i = \epsilon F_i^{(1)}$.

Considering $F_i = \epsilon F_i^{(1)}$, which is a valid assumption for most hydrodynamic problems,[3] the familiar steps from Sect. 4.1 lead to a hierarchy of ϵ-perturbed

[3] In certain cases, the forcing term requires a higher-order expansion. For example, for certain axisymmetric LB models [48, 49], the formal expansion of the forcing term is $F_i = \epsilon F_i^{(1)} + \epsilon^2 F_i^{(2)}$.

equations, similar to (4.9a) and (4.9b), now with a force term:

$$O(\epsilon): \qquad \left(\partial_t^{(1)} + c_{i\alpha}\partial_\alpha^{(1)}\right) f_i^{\mathrm{eq}} - \left(1 - \frac{\Delta t}{2\tau}\right) F_i^{(1)} = -\frac{1}{\tau} f_i^{(1)}, \tag{6.33a}$$

$$O(\epsilon^2): \quad \partial_t^{(2)} f_i^{\mathrm{eq}} + \left(\partial_t^{(1)} + c_{i\alpha}\partial_\alpha^{(1)}\right)\left(1 - \frac{\Delta t}{2\tau}\right)\left(f_i^{(1)} + \frac{\Delta t}{2} F_i^{(1)}\right) = -\frac{1}{\tau} f_i^{(2)}. \tag{6.33b}$$

In the presence of an external force, the hydrodynamic moments are no longer conserved. This leads to a redefinition of the solvability conditions for mass and momentum:

$$\sum_i f_i^{\mathrm{neq}} = -\frac{\Delta t}{2} \sum_i F_i^{(1)}, \tag{6.34a}$$

$$\sum_i \boldsymbol{c}_i f_i^{\mathrm{neq}} = -\frac{\Delta t}{2} \sum_i \boldsymbol{c}_i F_i^{(1)}. \tag{6.34b}$$

Likewise, the extension to "strengthened" order-by-order solvability conditions reads

$$\sum_i f_i^{(1)} = -\frac{\Delta t}{2} \sum_i F_i^{(1)} \qquad \text{and} \qquad \sum_i f_i^{(\mathrm{k})} = 0, \tag{6.35a}$$

$$\sum_i \boldsymbol{c}_i f_i^{(1)} = -\frac{\Delta t}{2} \sum_i \boldsymbol{c}_i F_i^{(1)} \qquad \text{and} \qquad \sum_i \boldsymbol{c}_i f_i^{(\mathrm{k})} = \boldsymbol{0} \tag{6.35b}$$

with $k \geq 2$, [5, 37], which results from $F_i^{(1)} \sim O(\epsilon)$, only affecting $f_i^{(1)}$ and not higher ϵ scales.

In order to proceed, we require the functional form of F_i. We continue with the specific form in (6.14) whose moments are given in (6.15). In particular, there are no mass sources, i.e. the right-hand sides in (6.34a) and (6.35a) vanish.

By taking the zeroth and first moments of (6.33a), we obtain at $O(\epsilon)$:

$$\partial_t^{(1)} \rho + \partial_\gamma^{(1)} (\rho u_\gamma) = 0, \tag{6.36a}$$

$$\partial_t^{(1)} (\rho u_\alpha) + \partial_\beta^{(1)} \Pi_{\alpha\beta}^{\mathrm{eq}} = F_\alpha. \tag{6.36b}$$

Here, $\Pi^{\mathrm{eq}}_{\alpha\beta} = \sum_i c_{i\alpha}c_{i\beta}f^{\mathrm{eq}}_i = \rho u_\alpha u_\beta + \rho c_s^2\delta_{\alpha\beta}$, according to (4.11a). Similarly, by taking the zeroth and first moments of (6.33b), we obtain at $O(\epsilon^2)$:

$$\partial_t^{(2)}\rho = 0, \tag{6.37a}$$

$$\partial_t^{(2)}(\rho u_\alpha) + \partial_\beta^{(1)}\left(1 - \frac{\Delta t}{2\tau}\right)\Pi^{(1)}_{\alpha\beta} = 0. \tag{6.37b}$$

By combining the mass and momentum equations in (6.36) and (6.37), respectively, we obtain

$$\left(\epsilon\partial_t^{(1)} + \epsilon^2\partial_t^{(2)}\right)\rho + \epsilon\partial_\gamma^{(1)}(\rho u_\gamma) = 0, \tag{6.38a}$$

$$\left(\epsilon\partial_t^{(1)} + \epsilon^2\partial_t^{(2)}\right)(\rho u_\alpha) + \epsilon\partial_\beta^{(1)}\Pi^{\mathrm{eq}}_{\alpha\beta} = \epsilon F_\alpha^{(1)} - \epsilon^2\partial_\beta^{(1)}\left(1 - \frac{\Delta t}{2\tau}\right)\Pi^{(1)}_{\alpha\beta}. \tag{6.38b}$$

To close the moment system in (6.38), we require an expression of $\Pi^{(1)}_{\alpha\beta}$ in terms of known quantities. We can achieve this by taking the second moment of (6.33a),

$$\partial_t^{(1)}\Pi^{\mathrm{eq}}_{\alpha\beta} + \partial_\gamma^{(1)}\Pi^{\mathrm{eq}}_{\alpha\beta\gamma} - \left(1 - \frac{\Delta t}{2\tau}\right)\sum_i F_i^{(1)}c_{i\alpha}c_{i\beta} = -\frac{1}{\tau}\Pi^{(1)}_{\alpha\beta}. \tag{6.39}$$

Here we have used the identity

$$\Pi^{(1)}_{\alpha\beta} = \sum_i f_i^{(1)}c_{i\alpha}c_{i\beta} + \frac{\Delta t}{2}\sum_i F_i^{(1)}c_{i\alpha}c_{i\beta} \tag{6.40}$$

that can be deduced by applying the Chapman-Enskog decomposition to (6.26c).

$\Pi^{(1)}_{\alpha\beta}$ is the contribution responsible for the viscous stress at macroscopic level. Therefore, the role of $\sum_i F_i^{(1)}c_{i\alpha}c_{i\beta}$ is to remove spurious forcing terms possibly appearing in $\Pi^{(1)}_{\alpha\beta}$ so that its form is the same as for the force-free case (cf. (4.15)):

$$\Pi^{(1)}_{\alpha\beta} = -\rho c_s^2\tau\left(\partial_\beta^{(1)}u_\alpha + \partial_\alpha^{(1)}u_\beta\right) + O(u^3). \tag{6.41}$$

Therefore, the viscous stress is still given by $\sigma_{\alpha\beta} = -\left(1 - \frac{\Delta t}{2\tau}\right)\Pi^{(1)}_{\alpha\beta}$, just as in the force-free case, (4.14).

Finally, we can re-assemble $\partial_t = \epsilon\partial_t^{(1)} + \epsilon^2\partial_t^{(2)}$ and use $\Pi_{\alpha\beta}^{\mathrm{eq}}$ and $\Pi_{\alpha\beta}^{(1)}$ to obtain from (6.38) the correct form (up to $O(u^3)$ error terms) of the unsteady NSE with forcing term:

$$\partial_t\rho + \partial_\gamma(\rho u_\gamma) = 0, \tag{6.42a}$$

$$\partial_t(\rho u_\alpha) + \partial_\beta\left(\rho u_\alpha u_\beta + \rho c_{\mathrm{s}}^2\delta_{\alpha\beta}\right) = \partial_\beta\left[\eta\left(\partial_\beta u_\alpha + \partial_\alpha u_\beta\right)\right] + F_\alpha. \tag{6.42b}$$

As usual, the dynamic shear and bulk viscosities are $\eta = \rho c_{\mathrm{s}}^2(\tau - \frac{\Delta t}{2})$ and $\eta_{\mathrm{B}} = 2\eta/3$, respectively (cf. Sect. 4.1).

6.5.2 *Errors Caused by an Incorrect Force Model*

Now that we know how to perform the Chapman-Enskog analysis with forces, we can evaluate whether the selected forcing scheme introduces errors in the recovered hydrodynamic model. According to Sect. 6.3, the formulation of the force model comprises two steps: (i) velocity space discretisation and (ii) space-time discretisation. Each of these steps comes with different error sources in case we do not deal with them properly.

6.5.2.1 Discretisation of Velocity Space: The Issue of Unsteady and Steady Cases

We can recognise the impact of an incorrect velocity space discretisation by distinguishing between unsteady and steady phenomena.

In *unsteady* state, the term $\partial_t^{(1)}\Pi_{\alpha\beta}^{\mathrm{eq}}$ contains the contribution $F_\alpha u_\beta + u_\alpha F_\beta$ (see Exercise 6.2 below). This contribution can be exactly cancelled by $\sum_i F_i c_{i\alpha} c_{i\beta}$, providing the force term F_i is expanded up to the *second velocity order* as shown in (6.14) [15, 20]. This way, we can correctly recover the unsteady NSE with force, (6.42).

Exercise 6.2 Show that

$$\begin{aligned}\partial_t^{(1)}\Pi_{\alpha\beta}^{\mathrm{eq}} &= \partial_t^{(1)}\left(\rho u_\alpha u_\beta + \rho c_{\mathrm{s}}^2\delta_{\alpha\beta}\right)\\ &= -\partial_\gamma^{(1)}\left(\rho u_\alpha u_\beta u_\gamma\right) - c_{\mathrm{s}}^2\left(u_\alpha\partial_\beta^{(1)}\rho + u_\beta\partial_\alpha^{(1)}\rho\right)\\ &\quad - c_{\mathrm{s}}^2\delta_{\alpha\beta}\partial_\gamma^{(1)}(\rho u_\gamma) + F_\alpha^{(1)}u_\beta + u_\alpha F_\beta^{(1)}.\end{aligned} \tag{6.43}$$

Hint: apply the procedure outlined in Appendix A.2.2, including a forcing term.

In *steady* state, however, the term $\partial_t^{(1)} \Pi_{\alpha\beta}^{\mathrm{eq}}$ is immaterial (cf. Sect. 4.2.3). Hence, we could expect that the contribution $F_\alpha u_\beta + u_\alpha F_\beta$ no longer exists. This is not absolutely true, though.

On the one hand, when using the standard equilibrium, an identical term is retrieved due to the requirement that the shear stress depends on the gradients of the velocity $\boldsymbol{u}$ rather than on the gradients of the momentum $\rho\boldsymbol{u}$. Consequently, to cancel that term, $\sum_i F_i c_{i\alpha} c_{i\beta}$ is still required as a correction.

On the other hand, with the incompressible LB equilibrium, the steady incompressible NSE is recovered with no spurious terms as discussed in Sect. 4.3. Hence, unlike the previous cases, here we must set $\sum_i F_i c_{i\alpha} c_{i\beta} = 0$, i.e. F_i must be expanded only to the first velocity order. A second-order expansion of F_i would lead to an incorrect steady incompressible NSE affected by the divergence of $F_\alpha u_\beta + u_\alpha F_\beta$. We will illustrate this issue in Sect. 6.7 through numerical examples. A more detailed explanation of this subtle point can be found in [5, 23].

6.5.2.2 Discretisation of Space and Time: The Issue of Discrete Lattice Effects

We can understand the effect of an inaccurate space-time discretisation on the forcing term by repeating the Chapman-Enskog analysis, but this time with a first-order time integration scheme (cf. Sect. 6.3.1).

Let us assume a time-dependent process and a forcing term with second-order velocity discretisation, (6.14). It can be shown, see e.g. [4, 15], that the macroscopic equations reproduced in this case have the following incorrect form:

$$\begin{aligned} \partial_t \rho + \partial_\gamma (\rho u_\gamma) &= -\frac{\Delta t}{2} \partial_\gamma F_\gamma, \\ \partial_t (\rho u_\alpha) + \partial_\beta \left(\rho u_\alpha u_\beta + \rho c_\mathrm{s}^2 \delta_{\alpha\beta} \right) &= \partial_\beta \left[\eta \left(\partial_\beta u_\alpha + \partial_\alpha u_\beta \right) \right] + F_\alpha \\ &\quad - \frac{\Delta t}{2} \left[\partial_t F_\alpha + \partial_\beta \left(u_\alpha F_\beta + F_\alpha u_\beta \right) \right]. \end{aligned} \tag{6.44}$$

The difference between (6.44) and the "true" NSE with a force, (6.42), lies in the added $O(\Delta t)$ error terms [4, 15]. They are called *discrete lattice artefacts* since they act on the same scale as the viscous term $\eta \sim O(\Delta t)$. Thereby, they corrupt the macroscopic equations below the truncation error $O(\Delta t^2)$. These discrete artefacts lead to inconsistencies in the macroscopic equations for *both* mass and momentum. Therefore discrete lattice artefacts are more problematical than an incorrect velocity space discretisation which "only" corrupts the momentum equation.

6.6 Boundary and Initial Conditions with Forces

So far we have limited the discussion about forces to the bulk solution. In Chap. 5 we have already discussed the topic of initial and boundary conditions, but without including the effect of forces. We will now point out the required modifications of initial (Sect. 6.6.1) and boundary conditions (Sect. 6.6.2) due to the presence of forces.

6.6.1 Initial Conditions

Initial conditions are necessary for time-dependent problems. But even steady flows must be subject to a proper initialisation. Otherwise, initial errors may be conserved during the simulation and contaminate the steady-state solution. In Sect. 5.5 we discussed two ways of initialising LB simulations. Let us revisit them and work out the necessary modifications when forces are present.

The simplest strategy is to initiate the populations with their *equilibrium* state, $f_i(\boldsymbol{x}, t = 0) = f_i^{\mathrm{eq}}\left(\rho_0(\boldsymbol{x}), \boldsymbol{u}_0(\boldsymbol{x})\right)$, where ρ_0 and $\boldsymbol{u}_0$ refer to known initial density and velocity fields. We know from Sect. 6.3.2 that for problems with forces the macroscopic velocity is computed from $\rho\boldsymbol{u} = \sum_i f_i \boldsymbol{c}_i + \frac{\Delta t}{2}\boldsymbol{F}$. Therefore, to set an initial velocity $\boldsymbol{u}_0$ consistent with the force field, we take [50]

$$f_i(\boldsymbol{x}, t = 0) = f_i^{\mathrm{eq}}\left(\rho_0(\boldsymbol{x}), \bar{\boldsymbol{u}}_0(\boldsymbol{x})\right), \quad \bar{\boldsymbol{u}}_0 = \boldsymbol{u}_0 - \frac{\boldsymbol{F}\Delta t}{2\rho_0}. \tag{6.45}$$

Obviously, for low-order forcing schemes, where the macroscopic velocity is computed from $\rho\boldsymbol{u} = \sum_i f_i \boldsymbol{c}_i$, the equilibrium initialisation is the same as in the force-free case, i.e. $\bar{\boldsymbol{u}}_0 = \boldsymbol{u}_0$.

As discussed in Sect. 5.5, a more accurate initialisation consists of adding the non-equilibrium populations f_i^{neq} to f_i^{eq}. Given that the leading order of f_i^{neq}, i.e. $f_i^{(1)}$, depends on $\boldsymbol{F}$, cf. (6.33a), the non-equilibrium term added to (6.45) must be redefined [5, 51]:

$$f_i^{\mathrm{neq}} \approx -\frac{w_i \tau}{c_{\mathrm{s}}^2}\rho Q_{i\alpha\beta}\partial_\alpha u_\beta - \frac{w_i \Delta t}{2c_{\mathrm{s}}^2}\left(c_{i\alpha}F_\alpha + \frac{Q_{i\alpha\beta}}{2c_{\mathrm{s}}^2}\left(u_\alpha F_\beta + F_\alpha u_\beta\right)\right) \tag{6.46}$$

where $Q_{i\alpha\beta} = c_{i\alpha}c_{i\beta} - c_{\mathrm{s}}^2\delta_{\alpha\beta}$.

6.6.2 Boundary Conditions

Forces may also affect the operation of boundary conditions (cf. Sect. 5.2.4). We will discuss the consequences for both bounce-back and non-equilibrium bounce-back.

6.6.2.1 Bounce-Back

Although the principle of the bounce-back rule is not changed by the inclusion of forces, its accuracy does depend on the force implementation. If we do not work with the second-order space-time discretisation of the LBE, the macroscopic laws established by the bounce-back rule will be affected by discrete lattice artefacts. We demonstrate this issue by looking at a simple example: a hydrostatic equilibrium where a constant force (e.g. gravity) is balanced by a pressure gradient.

We choose the second-order space-time discretisation for the bulk dynamics (cf. (6.25)). Also, let us consider a time-independent process: $\partial_t f_i = 0$. Then, the Chapman-Enskog analysis yields up to $O(\epsilon)$[4]:

$$f_i^{(1)} = -\tau c_{i\alpha} \partial_\alpha^{(1)} f_i^{\mathrm{eq}} + \left(\tau - \frac{\Delta t}{2}\right) F_i^{(1)}. \tag{6.47}$$

Given that we are interested in the hydrostatic solution, i.e. $\boldsymbol{u} = \boldsymbol{0}$, the equilibrium reduces to $f_i^{\mathrm{eq}} = w_i \rho$ and the forcing term to $F_i = w_i \boldsymbol{c}_i \cdot \boldsymbol{F}/c_{\mathrm{s}}^2$. Inserting f_i^{eq} and F_i into (6.47), we get $f_i^{(1)} = -\tau w_i c_{i\alpha} \partial_\alpha^{(1)} \rho + (\tau - \Delta t/2) c_{i\alpha} F_\alpha / c_{\mathrm{s}}^2$. The macroscopic behaviour of the populations for this hydrostatic problem is completely determined by $f_i = f_i^{\mathrm{eq}} + \epsilon f_i^{(1)}$, without any approximation [4, 29].

The next step is transferring these results to the bounce-back formula applied at a resting wall, i.e. $f_{\bar{i}} = f_i^\star$ (cf. (5.24)). This way, one can describe the closure relation of the bounce-back rule in the form of a Chapman-Enskog decomposition:

$$f_{\bar{i}}^{\mathrm{eq}} + \epsilon f_{\bar{i}}^{(1)} = f_i^{\mathrm{eq}} + \left(1 - \frac{\Delta t}{\tau}\right) \epsilon f_i^{(1)} + \left(\tau - \frac{\Delta t}{2}\right) \Delta t \epsilon F_i^{(1)}. \tag{6.48}$$

After substituting the content of $f_i^{\mathrm{eq}}, f_i^{(1)}$ and $F_i^{(1)}$ into (6.48) and undertaking some algebraic simplifications, we arrive at the hydrostatic solution established by the bounce-back rule at boundary node $\boldsymbol{x}_{\mathrm{b}}$:

$$\left(\tau - \frac{\Delta t}{2}\right) \left(c_{\mathrm{s}}^2 \partial_\alpha \rho - F_\alpha\right)\Big|_{\boldsymbol{x}_{\mathrm{b}}} = 0. \tag{6.49}$$

The first factor in (6.49) is positive due to the stability requirement $\tau > \frac{\Delta t}{2}$ (cf. Sect. 4.4) and can be cancelled. Hence, we conclude that the LBE with the bounce-back rule is *exact* for the hydrostatic pressure solution where we expect the balance $c_{\mathrm{s}}^2 \partial_\alpha \rho = F_\alpha$.

But does the correct hydrostatic balance also hold for a first-order space-time discretisation of the force? Based on the bulk analysis presented in Sect. 6.5.2, we might conclude that nothing changes because bulk errors have the form of force

[4]Equation (6.47) results from omitting the time derivatives in equation (6.33a) based on the Chapman-Enskog analysis for steady flows discussed in Sect. 4.2.3.

derivatives which in turn vanish for a constant body force. However, the closure in the bounce-back boundary conditions can retain discrete lattice artefacts even for a constant force, as shown in Exercise 6.3. More details can be found in [29, 34].

Exercise 6.3 Repeat the Chapman-Enskog analysis for a first-order time discretisation of the LBE. Show that the hydrostatic balance established by the bounce-back rule at boundary node $\boldsymbol{x}_\mathrm{b}$ is then incorrectly predicted as

$$\left(\tau - \frac{\Delta t}{2}\right)\left(c_\mathrm{s}^2 \partial_\alpha \rho - F_\alpha\right)\Big|_{\boldsymbol{x}_\mathrm{b}} = \frac{\Delta t}{2} F_\alpha(\boldsymbol{x}_\mathrm{b}).$$

6.6.2.2 Non-equilibrium Bounce-Back

The fundamental principle of the wet-node technique is that boundary nodes follow the same rules as bulk nodes. Hence, to be consistent with the bulk, the algorithm for boundary nodes needs to be reformulated to account for the presence of a force as well. We demonstrate this for the non-equilibrium bounce-back (NEBB) method [52].

As we have seen in Sect. 5.3.4, wet boundary nodes must satisfy the macroscopic laws of bulk nodes through the velocity moments. Therefore, the first-order moment for the momentum is modified by the presence of a force when we use the second-order space-time discretisation in (6.26b). This leads to a number of changes in the NEBB algorithm.

Consider the top wall depicted in Fig. 5.21. As in Sect. 5.3.4, we will work in dimensional notation, which is noted by the presence of the particle velocity c that in lattice units is $c = 1$. The determination of the unknown wall density for the force-free case in (5.31) now changes to

$$\begin{aligned}
\rho_\mathrm{w} &= \sum_i f_i = \underbrace{f_0 + f_1 + f_2 + f_3 + f_5 + f_6}_{\text{known}} + \underbrace{f_4 + f_7 + f_8}_{\text{unknown}}, \\
\rho_\mathrm{w} u_y^\mathrm{w} &= \sum_i f_i c_{iy} + \frac{\Delta t}{2} \sum_i F_i c_{iy} = \underbrace{c\,(f_2 + f_5 + f_6)}_{\text{known}} - \underbrace{c\,(f_4 + f_7 + f_8)}_{\text{unknown}} + \frac{F_y^\mathrm{w} \Delta t}{2},
\end{aligned} \tag{6.50}$$

where index w refers to the macroscopic fluid properties evaluated at the wall, where wet boundary nodes lie. By combining these two equations we get

$$\rho_\mathrm{w} = \frac{c}{c + u_y^\mathrm{w}} \left(f_0 + f_1 + f_3 + 2\,(f_2 + f_5 + f_6) + \frac{F_y^\mathrm{w} \Delta t}{2c} \right). \tag{6.51}$$

The unknown boundary populations still have to be determined by the bounce-back of their non-equilibrium components, i.e. (5.42). Yet, compared to the

force-free case, now it is necessary to consider both tangential and normal momentum corrections N_α. The reason for that will become clear shortly. For now, let us consider the top wall in Fig. 5.21 and write the bounce-back of the non-equilibrium populations as[5]

$$\left.\begin{aligned} f_4^{\text{neq}} &= f_2^{\text{neq}} - N_y, \\ f_7^{\text{neq}} &= f_5^{\text{neq}} - N_x - N_y, \\ f_8^{\text{neq}} &= f_6^{\text{neq}} + N_x - N_y. \end{aligned}\right\} \Longrightarrow \begin{cases} f_4 = f_2 + \left(f_4^{\text{eq}} - f_2^{\text{eq}}\right) - N_y, \\ f_7 = f_5 + \left(f_7^{\text{eq}} - f_5^{\text{eq}}\right) - N_x - N_y, \\ f_8 = f_6 + \left(f_8^{\text{eq}} - f_6^{\text{eq}}\right) + N_x - N_y. \end{cases} \tag{6.52}$$

Using the known equilibrium distributions, we get

$$\begin{aligned} f_4 &= f_2 - \frac{2\rho_{\text{w}} u_y^{\text{w}}}{3c} - N_y, \\ f_7 &= f_5 - \frac{\rho_{\text{w}}}{6c}(u_x^{\text{w}} + u_y^{\text{w}}) - N_x - N_y, \\ f_8 &= f_6 - \frac{\rho_{\text{w}}}{6c}(-u_x^{\text{w}} + u_y^{\text{w}}) + N_x - N_y. \end{aligned} \tag{6.53}$$

We compute N_x by resorting to the first-order velocity moment along the boundary tangential direction:

$$\begin{aligned} \rho_{\text{w}} u_x^{\text{w}} &= \sum_i f_i c_{ix} + \frac{\Delta t}{2} \sum_i F_i c_{ix} \\ &= c\,(f_1 + f_5 + f_8) - c\,(f_3 + f_6 + f_7) + \frac{F_x^{\text{w}} \Delta t}{2} \\ &= c\,(f_1 - f_3) - c\,(f_7 - f_5) + c\,(f_8 - f_6) + \frac{F_x^{\text{w}} \Delta t}{2} \\ &= c\,(f_1 - f_3) + \frac{\rho_{\text{w}} u_x^{\text{w}}}{3} + 2cN_x + \frac{F_x^{\text{w}} \Delta t}{2}. \end{aligned} \tag{6.54}$$

This gives

$$N_x = -\frac{1}{2}(f_1 - f_3) + \frac{\rho_{\text{w}} u_x^{\text{w}}}{3c} - \frac{F_x^{\text{w}} \Delta t}{4c}. \tag{6.55}$$

[5]The sign convention for the normal momentum correction is in line with the tangential case, cf. (5.43). If $\boldsymbol{n}$ and $\boldsymbol{t}$ denote the wall normal and the wall tangential vectors and if their positive sign coincides with the positive sign of the Cartesian axis, then the normal and tangential momentum corrections appear in the algorithm as $f_{\bar{i}}^{\text{neq}}(\boldsymbol{x}_{\text{B}}, t) = f_i^{\text{neq}}(\boldsymbol{x}_{\text{B}}, t) - (\boldsymbol{n} \cdot \boldsymbol{c}_i)N_n - (\boldsymbol{t} \cdot \boldsymbol{c}_i)N_t$.

Similarly, we compute N_y based on the first-order velocity moment along the boundary normal direction:

$$\begin{aligned}\rho_{\mathrm{w}} u_y^{\mathrm{w}} &= \sum_i f_i c_{iy} + \frac{\Delta t}{2} \sum_i F_i c_{iy} \\ &= c\,(f_2 + f_5 + f_6) - c\,(f_4 + f_7 + f_8) + \frac{F_y^{\mathrm{w}} \Delta t}{2} \\ &= c\,(f_2 - f_4) - c\,(f_7 - f_5) + c\,(f_6 - f_8) + \frac{F_y^{\mathrm{w}} \Delta t}{2} \\ &= \frac{\rho_{\mathrm{w}} u_y^{\mathrm{w}}}{3} + 3cN_y + \frac{F_y^{\mathrm{w}} \Delta t}{2}.\end{aligned} \tag{6.56}$$

We obtain

$$N_y = -\frac{F_y^{\mathrm{w}} \Delta t}{6c}. \tag{6.57}$$

Clearly, the normal momentum correction N_y is only relevant when forces are included.

In the end, the NEBB prescribes the unknown populations with forces, here for a top wall:

$$\begin{aligned} f_4 &= f_2 - \frac{2\rho_{\mathrm{w}} u_y^{\mathrm{w}}}{3c} + \frac{F_y^{\mathrm{w}} \Delta t}{6c}, \\ f_7 &= f_5 + \frac{1}{2}(f_1 - f_3) - \frac{\rho_{\mathrm{w}} u_x^{\mathrm{w}}}{2c} - \frac{\rho_{\mathrm{w}} u_y^{\mathrm{w}}}{6c} + \frac{F_x^{\mathrm{w}} \Delta t}{4c} + \frac{F_y^{\mathrm{w}} \Delta t}{6c}, \\ f_8 &= f_6 - \frac{1}{2}(f_1 - f_3) + \frac{\rho_{\mathrm{w}} u_x^{\mathrm{w}}}{2c} - \frac{\rho_{\mathrm{w}} u_y^{\mathrm{w}}}{6c} - \frac{F_x^{\mathrm{w}} \Delta t}{4c} + \frac{F_y^{\mathrm{w}} \Delta t}{6c}. \end{aligned} \tag{6.58}$$

The extension of (6.58) to other boundary orientations is straightforward (cf. Exercise 6.4).

The necessity of including force corrections in the NEBB method has been recognised in a number of works, e.g. [53–55]. These terms prevent the appearance of discrete lattice artefacts in the macroscopic laws of wet boundary nodes. However, those errors terms are proportional to $\nabla \cdot \boldsymbol{F}$. Hence, they will only be macroscopically visible for *spatially varying* force fields. We will demonstrate this numerically in Sect. 6.7.

Exercise 6.4 Show that the Dirichlet velocity condition prescribed with the NEBB method at a left boundary takes the following form in the presence of a force $\boldsymbol{F} = (F_x, F_y)$:

$$\begin{aligned}
\rho &= \frac{c}{c-u_x^{\mathrm{w}}}\left(f_0+f_2+f_4+2\left(f_3+f_6+f_7\right)-\frac{F_x^{\mathrm{w}}\Delta t}{2c}\right),\\
f_1 &= f_3+\frac{2\rho_{\mathrm{w}}u_x^{\mathrm{w}}}{3c}-\frac{F_x^{\mathrm{w}}\Delta t}{6c},\\
f_5 &= f_7-\frac{1}{2}\left(f_2-f_4\right)+\frac{\rho_{\mathrm{w}}u_y^{\mathrm{w}}}{2c}+\frac{\rho_{\mathrm{w}}u_x^{\mathrm{w}}}{6c}-\frac{F_x^{\mathrm{w}}\Delta t}{6c}-\frac{F_y^{\mathrm{w}}\Delta t}{4c},\\
f_8 &= f_6+\frac{1}{2}\left(f_2-f_4\right)-\frac{\rho_{\mathrm{w}}u_y^{\mathrm{w}}}{2c}+\frac{\rho_{\mathrm{w}}u_x^{\mathrm{w}}}{6c}-\frac{F_x^{\mathrm{w}}\Delta t}{6c}+\frac{F_y^{\mathrm{w}}\Delta t}{4c}.
\end{aligned} \tag{6.59}$$

6.7 Benchmark Problems

So far we have limited the discussion about forces in the LBE to theoretical analyses. While this helps us understanding basic features underlying LB forcing schemes, we have yet to see actual effects on LB simulations. The goal of this section, therefore, is to illustrate the true impact of the force inclusion, particularly when an incorrect force model is adopted. We will compare four possible forcing strategies (summarised in Table 6.2).

The alternative forcing schemes presented in Sect. 6.4 can be considered equivalent to scheme IV in Table 6.2. Although they behave differently at higher orders, these differences are not relevant for the examples that will follow.

6.7.1 Problem Description

We consider a 2D Poiseuille channel flow driven by a combined pressure gradient $\partial p/\partial x$ and body force F_x:

$$\rho\nu\frac{\partial u_x}{\partial y} = \frac{\partial p}{\partial x} - F_x. \tag{6.60}$$

Table 6.2 LB forcing schemes tested in Sect. 6.7. They have different velocity or space-time discretisation orders

Scheme	Velocity order	Space-time order	Examples of references
I	1st [(6.16)]	1st [(6.20)]	[43, 56–60]
II	2st [(6.14)]	1st [(6.20)]	[16, 18, 20, 44, 48, 61]
III	1st [(6.16)]	2st [(6.25)]	[4, 23, 37, 45, 46, 62]
IV	2st [(6.14)]	2st [(6.25)]	[15, 25, 28, 51, 63, 64]

The velocity solution is

$$u_x(y) = \frac{1}{2\rho\nu}\left(\frac{\partial p}{\partial x} - F_x\right)\left[y^2 - \left(\frac{H}{2}\right)^2\right] \tag{6.61}$$

where the no-slip condition ($u_x = 0$) holds at bottom/top walls ($y = \pm H/2$) as shown in Fig. 1.1b.

6.7.2 Numerical Procedure

For the bulk nodes we use the BGK collision operator with the incompressible equilibrium from Sect. 4.3.2. We will make some comments about the application of the standard (compressible) equilibrium later. We consider and individually discuss two different wall boundary schemes: the bounce-back and the non-equilibrium bounce-back (NEBB) methods.

The simulations are initialised by setting $f_i(\boldsymbol{x}, t = 0) = f_i^{\text{eq}}(\rho = 1, \boldsymbol{u} = \boldsymbol{0})$ as explained in Sect. 6.6.1; they are stopped when the velocity u_x reaches the steady-state criterion $L_2 \leq 10^{-10}$ between 100 consecutive time steps (cf. Sect. 4.5.2). The channel domain is discretised using $N_x \times N_y = 5 \times 5$ grid nodes. We evaluate the LB results for each of the four strategies presented in Table 6.2 and compare them with the analytical solution in (6.61) through the L_2 error norm.

6.7.3 Constant Force

Let us start by considering the simplest case: a purely force-driven Poiseuille flow ($\partial p/\partial x = 0$). We use periodic boundary conditions for the inlet and outlet (cf. Sect. 5.3.1) and the force magnitude is $F_x = 10^{-3}$ (in simulation units).

Since the force is uniform, any possible *bulk error* caused by an incorrect forcing scheme vanishes (cf. Sect. 6.5.2). However, boundaries can still lead to errors (cf. Sect. 6.6.2).

6.7.3.1 Bounce-Back

The errors for the LBGK model with bounce-back walls for several τ values are summarised in Table 6.3. The velocity discretisation of the force plays no role in this case. Differences exist in the space-time discretisation, though. While both strategies are able to reproduce the parabolic solution exactly, this happens at different values of τ. The reason is that spatial discretisation errors are cancelled for

Table 6.3 L_2 errors for Poiseuille flow with constant force and bounce-back at the walls (LBGK, grid resolution of $N_x \times N_y = 5 \times 5$). Results are identical for standard and incompressible equilibria

	ϵ_u [%]	
$\tau/\Delta t$	Schemes I and II	Schemes III and IV
0.6	5.91	5.18
0.8	5.04	2.85
$\sqrt{3/16} + 1/2 = 0.933$	3.16	2.04×10^{-12}
1.0	1.82	1.82
$\sqrt{13/64} + 5/8 = 1.076$	1.42×10^{-12}	4.20
1.2	3.72	8.83
1.4	11.60	18.17

specific values of τ, depending on the discretisation order of the force scheme [20, 43, 62, 65].

6.7.3.2 Non-equilibrium Bounce-Back

The NEBB method reproduces the dynamical rules of the bulk solution at boundary nodes. Consequently, for a constant force, the errors discussed in Sect. 6.5.2 vanish. This makes the NEBB method exact for the parabolic velocity solution in (6.61), regardless the forcing scheme employed.

6.7.4 Constant Force and Pressure Gradient

Let us now increase the complexity of the previous exercise by considering the simultaneous presence of a constant force and pressure gradient. In terms of implementation, the only modification concerns the inlet and outlet boundaries which are now modelled with pressure periodic boundary conditions (cf. Sect. 5.3.2). The relative fraction of the pressure gradient and the force density has no impact on the velocity solution of an incompressible flow, providing their combined effect is kept fixed. Without loss of generality, the overall magnitude is $\left(F_x - \partial p/\partial x\right) = 2 \times 10^{-3}$ (in simulation units), where we consider a 50/50 contribution from each term.

Similarly to the previous case, a constant force leads to vanishing force errors in the bulk, regardless the forcing strategy adopted (cf. Sect. 6.5.2). Yet, the closure relations at boundaries established by the bounce-back rule can differ, depending on the forcing scheme adopted.

Table 6.4 L_2 errors for Poiseuille flow with constant force and pressure gradient, bounce-back at the walls and pressure periodic conditions at inlet/outlet (LBGK, grid resolution of $N_x \times N_y = 5 \times 5$, incompressible equilibrium)

	ϵ_u (%)	
$\tau/\Delta t$	Schemes I and II	Schemes III and IV
0.6	5.55	5.18
0.8	3.94	2.85
$\sqrt{3/16} + 1/2 = 0.933$	1.58	1.37×10^{-11}
1.0	5.78×10^{-13}	1.82
$\sqrt{13/64} + 5/8 = 1.076$	2.10	4.20
1.2	6.27	8.83
1.4	14.89	18.17

6.7.4.1 Bounce-Back

Table 6.4 summaries the errors obtained with bounce-back. While the velocity discretisation order plays no role, the space-time discretisation is important.

In fact, we see that only the second-order space-time discretisation guarantees that the force-driven solution is unchanged when adding the pressure gradient. This follows from comparing Table 6.3 and Table 6.4: solutions are exactly equivalent for any τ value in that case. From a physical point of view, this result is expected since a constant force and a constant pressure gradient are equivalent in incompressible hydrodynamics.

However, this physical equivalence can be violated numerically when a less accurate space-time force discretisation is adopted. According to Table 6.4, the value where the solution becomes exact, $\tau = \Delta t$, now differs from the pure force-driven case where $\tau = (\sqrt{13/64} + 5/8)\Delta t$ gives the exact solution.

6.7.4.2 Non-equilibrium Bounce-Back

Similarly to the purely force-driven case in Sect. 6.7.3, no force errors occur for a constant force. The explanation is the same as before.

6.7.5 Linear Force and Pressure Gradient

Finally, let us address the most interesting case in this exercise: the modelling of a spatially varying force. The force increases linearly along the streamwise direction, but the total contribution remains constant so that the overall magnitude remains locally $\left(F_x - \partial p/\partial x\right) = 2 \times 10^{-3}$ (in simulation units), with a 50/50 local contribution from each term. That means the slope of variation of each term is equal, but with different signs. More details about this test case are described in [23].

Compared to the last two cases, the key difference is that the body force is inhomogeneous now. Thus, the force bulk errors in Sect. 6.5.2 do not vanish any more. Both bulk and boundary errors can now interfere with the LB solution in case of an incorrect force implementation. This case allows us to identify the most accurate LB forcing scheme for *steady incompressible* flow problems.

6.7.5.1 Bounce-Back

According to Table 6.5, the first-order space-time discretisation never reaches the exact solution, regardless of the τ value. This is due to the non-vanishing bulk errors given by (6.44). However, also the velocity discretisation affects the bulk error. As outlined in Sect. 6.3 and Sect. 6.5.2, the correct modelling of steady incompressible hydrodynamics with a body force requires a *first-order* velocity discretisation of the forcing term. This is confirmed in Table 6.5 where only scheme III can reproduce the exact solution.

The reason for the exact solution only occurring for $\tau = (\sqrt{3/16} + 1/2)\Delta t$ is the τ-dependence of the bounce-back scheme. As explained in Sect. 5.3.3 and Sect. 5.4.1, only this value of τ locates the wall *exactly* halfway between nodes in a parabolic flow profile.

6.7.5.2 Non-equilibrium Bounce-Back

The NEBB method leads to essentially the same results as the bounce-back scheme; compare Table 6.5 and Table 6.6. Once again, a bulk error can corrupt the LB solutions if the velocity and space-time discretisations are not properly handled. To reproduce steady incompressible hydrodynamics exactly, scheme III is necessary (cf. Table 6.6). Here the exact solution is reproduced for any value of τ because: (i) scheme III leads to a velocity solution in bulk free from errors (cf. Sect. 6.5.2)

Table 6.5 L_2 errors for Poiseuille flow with linear force and pressure gradient, bounce-back at the walls and pressure periodic conditions at inlet/outlet (LBGK, grid resolution of $N_x \times N_y = 5 \times 5$, incompressible equilibrium)

	ϵ_u (%)			
$\tau/\Delta t$	Scheme I	Scheme II	Scheme III	Scheme IV
0.6	6.10	5.44	5.18	5.07
0.8	4.12	3.84	2.85	2.74
$\sqrt{3/16} + 1/2 = 0.933$	1.70	1.48	8.46×10^{-15}	0.12
1.0	0.51	0.51	1.82	1.94
$\sqrt{13/64} + 5/8 = 1.076$	2.00	2.21	5.47	4.32
1.2	6.19	6.39	8.83	8.96
1.4	14.81	15.03	18.17	18.32

Table 6.6 L_2 errors for Poiseuille flow with linear force and pressure gradient, non-equilibrium bounce-back at the walls and pressure periodic conditions at inlet/outlet (LBGK, grid resolution of $N_x \times N_y = 5 \times 5$, incompressible equilibrium)

	ϵ_u (%)			
$\tau/\Delta t$	Scheme I	Scheme II	Scheme III	Scheme IV
0.6	0.19	0.27	8.60×10^{-14}	0.05
0.8	0.42	0.41	4.41×10^{-15}	0.07
$\sqrt{3/16}+1/2$	0.63	0.63	1.24×10^{-14}	0.07
1.0	0.74	0.74	1.32×10^{-14}	0.07
$\sqrt{13/64}+5/8$	0.87	0.87	1.24×10^{-14}	0.07
1.2	1.08	1.08	2.39×10^{-14}	0.07
1.4	1.42	1.42	1.55×10^{-14}	0.07

and (ii) the NEBB scheme accommodates this solution at the wall in an exact and τ-independent way (cf. Sect. 6.6.2).

6.7.6 *Role of Compressibility*

The previous exercises used the incompressible equilibrium that allows for the exact description of steady incompressible flows (cf. Sect. 4.3.2). This explains why all test cases could reach an exact solution, providing the correct forcing scheme is chosen.

On the other hand, the standard equilibrium recovers the compressible NSE which approximates incompressible hydrodynamics in the limit of slow flows and small density (pressure) variations [4]. Associated with this are *compressibility errors*, as discussed in Sect. 4.5.4.

Compared to the discrete lattice artefacts, coming from the incorrect force modelling, and/or the velocity slip, created by the bounce-back boundary scheme, the compressibility errors typically have a secondary impact [4]. Still, they always contaminate the solutions. In this case, they preclude exact results even when the above error sources are corrected. This issue will be illustrated below, by repeating the previous exercises with the standard (compressible) equilibrium. As we shall see, although compressibility errors may obscure the clear identification of the force discretisation artefacts, the trends of the incompressible equilibrium remain. But this time, the lowest minimum in L_2, for a spatially varying force, is found in the forcing scheme with the second-order discretisation in velocity space (cf. Sect. 6.5.2), although differences are very small.

6.7.6.1 Constant Force

The first test case was the purely force-driven Poiseuille flow. Since no pressure variations occur in this setup, we have identical results for both the incompressible and the standard equilibria.

6.7.6.2 Constant Force and Pressure Gradient

The second test case considered the simultaneous presence of a constant force and pressure gradient. As pressure varies here, the LB solution now contains compressibility errors.

The effect of the velocity discretisation order remains negligible, although machine accuracy is never reached. This is in contrast to the incompressible case. Once again, the space-time discretisation has the largest effect, as shown in Fig. 6.2.

Using bounce-back boundaries (Fig. 6.2a), the first-order space-time discretisation features the L_2 minimum at $\tau = \Delta t$, while for the second-order discretisation it is at $\tau = (\sqrt{13/64} + 5/8)\Delta t$. This behaviour is similar to the incompressible case, Table 6.4, except that now the minimum does not correspond to the exact solution. The same kind of qualitative results occur when the NEBB method is used, yet without showing any clear minimum (cf. Fig. 6.2b). Obviously the NEBB has superior accuracy when the compressible equilibrium is used.

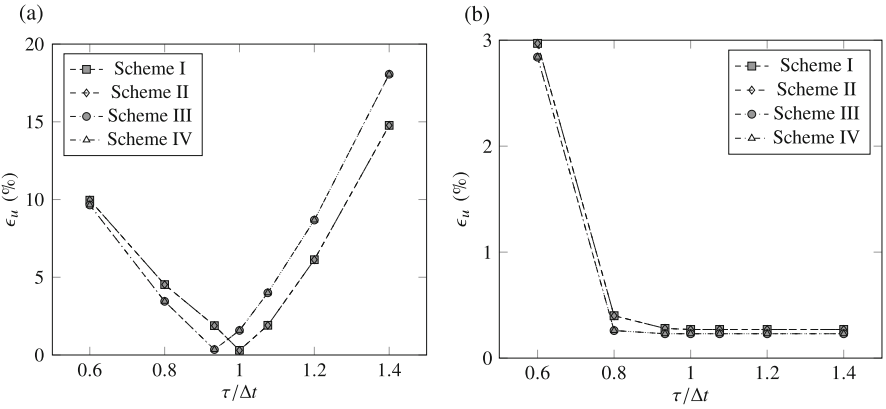

Fig. 6.2 L_2 errors for Poiseuille flow with constant force and pressure gradient, periodic conditions at inlet/outlet (LBGK, grid resolution of $N_x \times N_y = 5 \times 5$, compressible equilibrium). (**a**) Bounce-back. (**b**) Non-equilibrium bounce-back

6.7.6.3 Linear Force and Pressure Gradient

The third test case was a linearly varying body force and an according pressure gradient. In addition to the bulk errors caused by force artefacts, the pressure (density) variation also introduces compressibility errors.

While the order of the velocity discretisation has a slightly larger effect than in the previous problem, the LB solution is dominated by compressibility errors. In fact, using bounce-back, the second-order velocity discretisation is only more accurate for small values of τ (cf. Fig. 6.3a). With the NEBB method, the second-order velocity discretisation is more accurate for all values of τ (cf. Fig. 6.3b). Yet, the accuracy improvement due to a second-order velocity discretisation is marginal and not comparable to the incompressible case. Once again, the accuracy of the LB solution depends mostly on the space-time discretisation as shown in Fig. 6.3.

The conclusions are similar to those of the case with constant force and pressure gradient. The second-order space-time discretisation leads to minimum L_2 values. Still, due to the non-trivial interplay of force and compressibility errors, the second-order space-time discretisation does not perform better in the full range of τ, which is particularly noticeable for the bounce-back method. Generally, the NEBB method has smaller errors for the problem considered in this section.

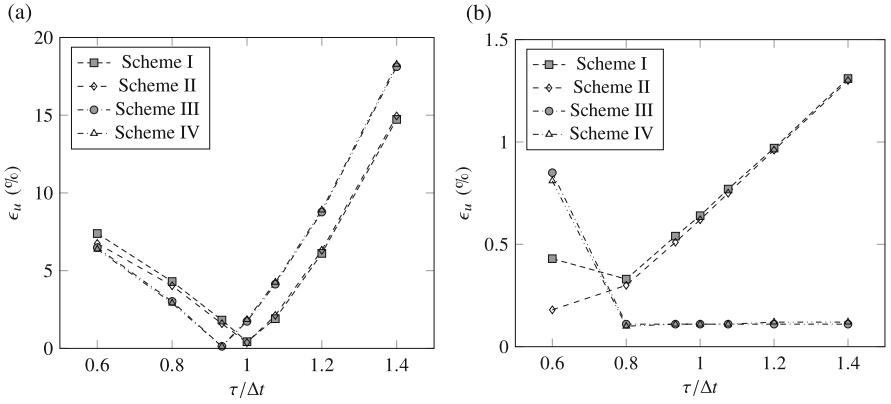

Fig. 6.3 L_2 errors for Poiseuille flow with linear force and pressure gradient, pressure periodic conditions at inlet/outlet (LBGK, grid resolution of $N_x \times N_y = 5 \times 5$, compressible equilibrium). (**a**) Bounce-back. (**b**) Non-equilibrium bounce-back

References

1. E. Bodenschatz, W. Pesch, G.Ahlers, Annu. Rev. Fluid Mech. **32**, 709 (2010)
2. S.I. Abarzhi, Phil. Trans. R. Soc. A **368**, 1809 (2010)
3. J. Lighthill, *Waves in Fluids*, 6th edn. (Cambridge University Press, Cambridge, 1979)
4. J.M. Buick, C.A. Greated, Phys. Rev. E **61**(5), 5307 (2000)
5. G. Silva, V. Semiao, J. Fluid Mech. **698**, 282 (2012)
6. P.J. Dellar, Comput. Math. Appl. **65**(2), 129 (2013)
7. R. Salmon, J. Mar. Res. **57**(3), 847 (1999)
8. J. Wang, M. Wang, Z. Li, J. Colloid Interf. Sci. **296**, 729 (2006)
9. M. Wang, Q. Kang, J. Comput. Phys. **229**, 728 (2010)
10. T.Y. Lin, C.L. Chen, Appl. Math. Model. **37**, 2816 (2013)
11. O. Shardt, S.K. Mitra, J.J. Derksen, Chem. Eng. J. **302**, 314 (2016)
12. S.H. Kim, H. Pitsch, Phys. Fluids **19**, 108101 (2007)
13. J. Zhang, D.Y. Kwok, Phys. Rev. E **73**, 047702 (2006)
14. L. Talon, D. Bauer, D. Gland, H. Auradou, I. Ginzburg, Water Resour. Res. **48**, W04526 (2012)
15. Z. Guo, C. Zheng, B. Shi, Phys. Rev. E **65**, 46308 (2002)
16. N.S. Martys, X. Shan, H. Chen, Phys. Rev. E **58**(5), 6855 (1998)
17. X. Shan, X.F. Yuan, H. Chen, J. Fluid Mech. **550**, 413 (2006)
18. L.S. Luo, Phys. Rev. E **62**(4), 4982 (2000)
19. P. Dellar, Phys. Rev. E **64**(3) (2001)
20. A.J.C. Ladd, R. Verberg, J. Stat. Phys. **104**(5–6), 1191 (2001)
21. S. Ansumali, I.V. Karlin, H.C. Öttinger, Phys. Rev. Lett. **94**, 080602 (2005)
22. J.R. Clausen, Phys. Rev. E **87**, 013309 (2013)
23. G. Silva, V. Semiao, Physica A **390**(6), 1085 (2011)
24. Z. Guo, C. Zheng, B. Shi, T. Zhao, Phys. Rev. E **75**(036704), 1 (2007)
25. R.W. Nash, R. Adhikari, M.E. Cates, Phys. Rev. E **77**(2), 026709 (2008)
26. S. Ubertini, P. Asinari, S. Succi, Phys. Rev. E **81**(1), 016311 (2010)
27. X. He, S. Chen, G.D. Doolen, J. Comput. Phys. **146**(1), 282 (1998)
28. A. Kuzmin, Z. Guo, A. Mohamad, Phil. Trans. Royal Soc. A **369**, 2219 (2011)
29. R.G.M. Van der Sman, Phys. Rev. E **74**, 026705 (2006)
30. S.D.C. Walsh, H. Burwinkle, M.O. Saar, Comput. Geosci. **35**(6), 1186 (2009)
31. Z. Guo, T.S. Zhao, Phys. Rev. E **66**, 036304 (2002)
32. X. Nie, N.S. Martys, Phys. Fluids **19**, 011702 (2007)
33. I. Ginzburg, Phys. Rev. E **77**, 066704 (2008)
34. I. Ginzburg, G. Silva, L. Talon, Phys. Rev. E **91**, 023307 (2015)
35. R. Salmon, J. Mar. Res. **57**(3), 503 (1999)
36. H. Huang, M. Krafczyk, X. Lu, Phys. Rev. E **84**(4), 046710 (2011)
37. I. Ginzburg, F. Verhaeghe, D. d'Humières, Commun. Comput. Phys. **3**, 427 (2008)
38. X. Shan, H. Chen, Phys. Rev. E **47**(3), 1815 (1993)
39. X. He, X. Shan, G. Doolen, Phys. Rev. E. Rapid Comm. **57**(1), 13 (1998)
40. A. Kupershtokh, D. Medvedev, D. Karpov, Comput. Math. Appl. **58**(5), 965 (2009)
41. D. Lycett-Brown, K.H. Luo, Phys. Rev. E **91**, 023305 (2015)
42. A. Kupershtokh, in *Proc. 5th International EHD Workshop, University of Poitiers, Poitiers, France* (2004), p. 241–246
43. X. He, Q. Zou, L.S. Luo, M. Dembo, J. Stat. Phys. **87**(1–2), 115 (1997)
44. L.S. Luo, Phys. Rev. Lett. **81**(8), 1618 (1998)
45. I. Ginzbourg, P.M. Adler, J. Phys. II France **4**(2), 191 (1994)
46. A.J.C. Ladd, J. Fluid Mech. **271**, 285 (1994)
47. Z. Guo, C. Zheng, B. Shi, Phys. Rev. E **83**, 036707 (2011)
48. I. Halliday, L.A. Hammond, C.M. Care, K. Good, A. Stevens, Phys. Rev. E **64**, 011208 (2001)
49. T. Reis, T.N. Phillips, Phys. Rev. E **75**, 056703 (2007)
50. I. Ginzburg, F. Verhaeghe, D. d'Humières, Commun. Comput. Phys. **3**, 519 (2008)

51. M. Gross, N. Moradi, G. Zikos, F. Varnik, Phys. Rev. E **83**(1), 017701 (2011)
52. Q. Zou, X. He, Phys. Fluids **9**, 1591 (1997)
53. A. D'Orazio, S. Succi, Future Generation Comput. Syst. **20**, 935 (2004)
54. A. Markus, G. Hazi, Phys. Rev. E **83**, 046705 (2011)
55. A. Karimipour, A.H. Nezhad, A. D'Orazio, E. Shirani, J. Theor. Appl. Mech. **51**, 447 (2013)
56. D.R. Noble, Chen, J.G. Georgiadis, R.O. Buckius, Phys. Fluids **7**, 203 (1995)
57. M. Bouzidi, M. Firdaouss, P. Lallemand, Phys. Fluids **13**, 3452 (2001)
58. D.A. Wolf-Gladrow, *Lattice-Gas Cellular Automata and Lattice Boltzmann Models* (Springer, New York, 2005)
59. M. Junk, A. Klar, L.S. Luo, J. Comput. Phys. **210**, 676 (2005)
60. T. Krüger, F. Varnik, D. Raabe, Phys. Rev. E **79**(4), 046704 (2009)
61. C.M. Pooley, K. Furtado, Phys. Rev. E **77**, 046702 (2008)
62. I. Ginzburg, D. d'Humières, Phys. Rev. E **68**, 066614 (2003)
63. K. Premnath, J. Abraham, J. Comput. Phys. **224**, 539 (2007)
64. J. Latt, Hydrodynamic limit of lattice Boltzmann equations. Ph.D. thesis, University of Geneva (2007)
65. M. Rohde, D. Kandhai, J.J. Derksen, H.E.A. Van den Akker, Phys. Rev. **67**, 066703 (2003)