Thermal fluctuations and boundary conditions in the lattice Boltzmann method

Dissertation
zur Erlangung des Grades
"Doktor der Naturwissenschaften"
am Fachbereich Physik, Mathematik
und Informatik
der Johannes Gutenberg-Universität
in Mainz

Ulf Daniel Schiller
geboren in Georgsmarienhütte

Mainz
November 2008

Datum der mündlichen Prüfung: 11. Dezember 2008

To my family

“I have not failed. I’ve just found 10,000 ways that won’t work.”

THOMAS ALVA EDISON

Zusammenfassung

Die Lattice-Boltzmann-Methode ist ein verbreitetes Verfahren zur Simulation hydrodynamischer Wechselwirkungen in weicher Materie und komplexen Flüssigkeiten. Dabei wird das Lösungsmittel durch ein räumliches Gitter repräsentiert, auf dem Teilchenpopulationen entlang der diskreten Kanten zwischen den Gitterpunkten propagieren und lokal miteinander kollidieren. Diese Mikrodynamik führt auf großen Skalen zu einem hydrodynamischen Strömungsfeld, wie es die Navier-Stokes-Gleichung beschreibt. In der vorliegenden Arbeit werden verschiedene methodische Erweiterungen der Lattice-Boltzmann-Methode entwickelt.

In komplexen Flüssigkeiten, z. B. Suspensionen, ist die Brownsche Molekularbewegung von zentraler Bedeutung. Sie kann jedoch mit der klassischen Lattice-Boltzmann-Methode nicht simuliert werden, da die Dynamik vollständig deterministisch ist. Es ist jedoch möglich, zusätzliche thermische Schwankungen einzuführen, mit denen fluktuierende Hydrodynamik reproduziert werden kann. In der Arbeit wird mit Hilfe eines verallgemeinerten Gitter-Gas-Modells eine systematische Herleitung der Gleichgewichtsverteilung aus Prinzipien der Statistischen Mechanik präsentiert. Der stochastische Anteil der Dynamik wird als Monte-Carlo-Prozess betrachtet, der dem Prinzip des detaillierten Gleichgewichts genügen muss. Hieraus lässt sich eine Bedingung für die thermischen Fluktuationen ableiten, die insbesondere besagt, dass alle Freiheitsgrade des Systems einschließlich der kinetischen Moden thermalisiert werden müssen. Der entwickelte Formalismus stellt sicher, dass die verbesserte fluktuierende Lattice-Boltzmann-Methode sowohl die fluktuierende Hydrodynamik reproduziert als auch konsistent auf der Statistischen Mechanik aufbaut. Dies könnte die Grundlage für zukünftige Erweiterungen der Methode sein, z. B. im Hinblick auf Multi-Phasen-Systeme oder Thermo-Hydrodynamik.

Ein wichtiges Anwendungsgebiet der Lattice-Boltzmann-Methode ist die Mikrofluidik. Simulationen leisten hier neben Theorie und Experiment einen wichtigen Beitrag auf dem Weg zum “Labor auf dem Chip”. Mikrofluidik-Systeme zeichnen sich durch ein hohes Verhältnis von Oberflächen zu Volumen aus. Besonderes Augenmerk muss daher auf die Randbedingungen gelegt werden, wobei im Mikrobereich die in der Hydrodynamik übliche Haftbedingung an der Oberfläche durch eine Gleitbedingung zu ersetzen ist.

In dieser Arbeit wird eine Randbedingung für die Lattice-Boltzmann-Methode konstruiert, die die Einstellung der Gleitlänge über einen entsprechenden Modellparameter ermöglicht. Es wird weiterhin ein neuer Ansatz zur Konstruktion von Randbedingungen untersucht. Ausgangspunkt ist dabei die explizite Berücksichtigung der gebrochenen Symmetrie an einer Oberfläche innerhalb des Gittermodells. Die Lattice-Boltzmann-Methode wird systematisch auf die gebrochene Symmetrie verallgemeinert. Am Beispiel einer Poiseuille-Strömung wird gezeigt, dass eine spezielle Wahl des Kollisionsoperators an der Wand erforderlich ist, damit das Strömungsprofil korrekt reproduziert wird. Die systematische Vorgehensweise führt dabei zu einem erweiterten Verständnis von Randbedingungen in der Lattice-Boltzmann-Methode, das nicht nur bei der Interpretation von Simulationsergebnissen hilfreich ist, sondern auch zu zukünftigen Verbesserungen der Methode führen könnte.

Abstract

The lattice Boltzmann method is a popular approach for simulating hydrodynamic interactions in soft matter and complex fluids. The solvent is represented on a discrete lattice whose nodes are populated by particle distributions that propagate on the discrete links between the nodes and undergo local collisions. On large length and time scales, the microdynamics leads to a hydrodynamic flow field that satisfies the Navier-Stokes equation. In this thesis, several extensions to the lattice Boltzmann method are developed.

In complex fluids, for example suspensions, Brownian motion of the solutes is of paramount importance. However, it can not be simulated with the original lattice Boltzmann method because the dynamics is completely deterministic. It is possible, though, to introduce thermal fluctuations in order to reproduce the equations of fluctuating hydrodynamics. In this work, a generalized lattice gas model is used to systematically derive the fluctuating lattice Boltzmann equation from statistical mechanics principles. The stochastic part of the dynamics is interpreted as a Monte Carlo process, which is then required to satisfy the condition of detailed balance. This leads to an expression for the thermal fluctuations which implies that it is essential to thermalize all degrees of freedom of the system, including the kinetic modes. The new formalism guarantees that the fluctuating lattice Boltzmann equation is simultaneously consistent with both fluctuating hydrodynamics and statistical mechanics. This establishes a foundation for future extensions, such as the treatment of multi-phase and thermal flows.

An important range of applications for the lattice Boltzmann method is formed by microfluidics. Fostered by the "lab-on-a-chip" paradigm, there is an increasing need for computer simulations which are able to complement the achievements of theory and experiment. Microfluidic systems are characterized by a large surface-to-volume ratio and, therefore, boundary conditions are of special relevance. On the microscale, the standard no-slip boundary condition used in hydrodynamics has to be replaced by a slip boundary condition. In this work, a boundary condition for lattice Boltzmann is constructed that allows the slip length to be tuned by a single model parameter. Furthermore, a conceptually new approach for constructing boundary conditions is explored, where the reduced symmetry at the boundary is explicitly incorporated into the lattice model. The lattice Boltzmann method is systematically extended to the reduced symmetry model. In the case of a Poiseuille flow in a plane channel, it is shown that a special choice of the collision operator is required to reproduce the correct flow profile. This systematic approach sheds light on the consequences of the reduced symmetry at the boundary and leads to a deeper understanding of boundary conditions in the lattice Boltzmann method. This can help to develop improved boundary conditions that lead to more accurate simulation results.

Contents

List of symbols and notations

A variety of symbols and notations is used in this thesis. The following table shall help the reader to find the way through. Many formulas in this work can be written in more than one notation, the most frequent option probably being the choice of vector or index notation. The choices made in the text are an attempt to present the formulas in a way that is at the same time simple and intuitive, with the aim to guide the reader along the lines of thought. Therefore, we sometimes deliberately switch between different notations "on the fly". The table below lists the elementary symbols, some of which can be indexed in several ways and combinations of indexes can be used. We have not listed all possible combinations here. The meaning of combinations of symbols and notations should always become clear in the respective context.

Symbol/Notation	**Meaning**
Miscellanea	
$\mathbf{a} \cdot \mathbf{b}$	scalar product of two vectors $\mathbf{a}$ and $\mathbf{b}$
$\mathbf{a} \otimes \mathbf{b},\ \mathbf{ab}$	tensor product of two vectors $\mathbf{a}$ and $\mathbf{b}$ (cf. footnote 6 on page 15)
$\mathsf{a} : \mathsf{b}$	full contraction of rank-two tensors a and b
$\langle f \vert g \rangle$	scalar product of two functions f and g
$\frac{\partial}{\partial \mathbf{r}}$	gradient or divergence (nabla operator)
$\frac{\delta}{\delta f_i}$	functional derivative with respect to f_i
$(\partial f)_{\text{coll}}$	general collision operator in kinetic theory
$\mathbb{1}$	unit matrix

Dimensionless numbers	
Bo	Boltzmann number
Kn	Knudsen number
Ma	Mach number

Latin letters	
$A, B, C, \ldots$	coefficients in the discrete equilibrium distribution
$\tilde{A}, \tilde{B}, \tilde{C}, \ldots$	coefficients in the discrete boundary equilibrium distribution
$A, B_1, B_2, C_1, \ldots$	coefficients of the lattice tensors in the reduced symmetry
a	lattice spacing

Continued on next page

Symbol/Notation	Meaning

Continued from previous page

$\mathsf{a}^{(n)},\ \mathsf{a}^{(n)}_{\alpha_1 \ldots \alpha_n}$	tensor coefficients in the Hermite expansion
$\mathbf{b},\ b_k$	lengths of the basis vectors e_{ki} of mode space
c_s	speed of sound
$\mathbf{c}_i,\ c_{i\alpha}$	discrete velocity vectors
$\hat{\mathbf{c}}_i,\ \hat{c}_{i\alpha}$	dimensionless form of velocity vector, $\mathbf{c}_i = \hat{\mathbf{c}}_i\, a/\tau$
$E^n_{D,d}$	quadrature of degree d in D dimensions using n nodes
$\mathsf{E}^{(2)},\ \mathsf{E}^{(3)}$	tensors occurring in the Chapman-Enskog expansion
e_{ki}	basis vectors of mode space
$\mathbf{F}$	external force in the Boltzmann equation
$f(\mathbf{r}, \mathbf{v}, t)$	one-particle distribution function in kinetic theory
$f^N(\mathbf{r}, \mathbf{v}, t)$	truncated Hermite expansion of $f(\mathbf{r}, \mathbf{v}, t)$
$f^{\text{eq}}(\rho, \mathbf{u})$	local equilibrium distribution function, Maxwell-Boltzmann distribution
$f_i(\mathbf{r}, t)$	discrete velocity population in lattice models
$f_i^{\text{eq}}(\rho, \mathbf{u})$	local equilibrium population of discrete velocities
$\mathbf{f},\ f_i$	population numbers in general
f_i^*	post-collisional value of population number
$f_i^{\text{eq}},\ f_i^{\text{neq}}$	equilibrium and non-equilibrium part of discrete velocity population
$f_i^{(0)},\ f_i^{(1)},\ f_i^{(2)}$	different orders of the Chapman-Enskog expansion of the populations
$f^{B,\text{eq}}(\rho, \mathbf{u})$	local equilibrium population at a boundary
f_{ext}	external force
$\mathsf{G},\ G_{\alpha\beta}$	tensor in the forcing term of the lattice Boltzmann equation
$\mathbf{g},\ g_\alpha$	volumetric force
$\mathbf{j},\ j_\alpha$	hydrodynamic momentum density
k_B	Boltzmann's constant
l_{mfp}	molecular mean free path
L	hydrodynamic length scale, typically the width of a channel
m	mass
m_p	(fictitious) mass of a lattice Boltzmann particle
M	transformation matrix from populations to modes
M^B	transformation matrix from populations to modes at a boundary
$\mathbf{m},\ m_k$	hydrodynamic modes or moments
m_k^{eq}	equilibrium value of the moment m_k
m_k^*	post-collisional value of the moment m_k
$\mathbf{n},\ n_\alpha$	normal vector of a boundary surface
P	projector on the boundary normal $\mathsf{P} = \mathbf{n}\mathbf{n}^T$
p	hydrostatic pressure

Continued on next page

Symbol/Notation	Meaning

Continued from previous page

$p(\mathbf{r}, \mathbf{v}, t)$	polynomial in $\mathbf{v}$
$Q_{\alpha\beta\gamma\delta}$	correlations of the fluctuations of the stress tensor s
q	fraction of a cut link that lies outside a boundary
$\mathbf{r},\ r_\alpha$	position vector
$\mathbf{r}_1,\ r_{1\alpha}$	coarse-grained position vector $\mathbf{r}_1 = \epsilon\mathbf{r}$
$S(\{\nu_i\})$	associated entropy in the generalized lattice gas model
$\mathcal{S}(\{\nu_i\}, \chi, \boldsymbol{\lambda})$	entropy functional with Lagrange multipliers
$\mathsf{s},\ s_{\alpha\beta}$	fluctuating stress in Landau-Lifshitz fluctuating hydrodynamics
T	temperature
$\mathsf{T}^{(n)},\ T^{(n)}_{\alpha_1\ldots\alpha_n}$	lattice tensor or lattice sum (includes the weights w_i)
t	time variable
t_1	coarse-grained time variable, convective scale $t_1 = \epsilon t$
t_2	coarse-grained time variable, diffusive scale $t_2 = \epsilon^2 t$
$\mathbf{u},\ u_\alpha$	hydrodynamic flow velocity
$\mathbf{u}_{\text{slip}}$	slip velocity at a boundary
$\mathbf{v}$	molecular velocity in kinetic theory
$w_i,\ w_q$	weights in the lattice Boltzmann model, i indexes individual $\mathbf{c}_i$ while q indexes a subshell with the same absolute velocity

Greek letters	
α	Maxwell's accommodation coefficient
β	slip coefficient
γ	friction parameter in the hydrodynamic slip boundary condition
γ_k	relaxation parameter of the k-th moment m_k
$\delta_{\alpha\beta}$	Kronecker delta
$\delta_{\alpha\beta\gamma\delta}$	rank-4 cubic anisotropy, $\delta_{\alpha\beta\gamma\delta} = \delta_{\alpha\beta}\delta_{\beta\gamma}\delta_{\gamma\delta}$ (no summation over double indices)
$\boldsymbol{\delta}^{(n)},\ \delta^{(n)}_{\alpha_1\ldots\alpha_n}$	n-th rank isotropic tensor, sum of all $(2n-1)!!$ products of Kronecker deltas
Δ_i	discrete collision term in the lattice Boltzmann equation
Δ'_i	stochastic collision term
Δ^g_i	forcing term
$\Delta^{(0)}_i,\ \Delta^{(1)}_i,\ \Delta^{(2)}_i$	different orders of Chapman-Enskog expansion of the collision term
δ_B	slip length at a boundary
ϵ	expansion parameter in the Chapman-Enskog expansion
ζ	friction parameter in the wall friction model
$\eta_{\alpha\beta\gamma\delta}$	viscosity tensor

Continued on next page

Symbol/Notation	Meaning

Continued from previous page

η_b	bulk viscosity
η_s	shear viscosity
θ	reduced (dimensionless) temperature
κ_4	coefficient of the anisotropic term of the rank-4 lattice sum
λ	eigenvalue of the BGK collision operator (single-relaxation-time approximation)
λ_b	eigenvalue of the bulk stress modes, related to bulk viscosity
λ_g	eigenvalue of the kinetic (ghost) modes
λ_k	eigenvalues of the k-th mode m_k of the collision operator
λ_s	eigenvalue of the shear stress modes, related to shear viscosity
$\lambda_\rho,\ \lambda_{\mathsf{j}},\ \lambda_{\mathsf{\Pi}}$	Lagrange multipliers
$\boldsymbol{\lambda},\ \lambda_\alpha$	Lagrange multiplier
$\lambda_\alpha^{(1)},\ \lambda_\alpha^{(2)}$	low velocity expansion of the Lagrange multiplier
$\boldsymbol{\lambda}^B$	Lagrange multiplier at a boundary
$\lambda_\alpha^{B,(1)},\ \lambda_\alpha^{B,(2)}$	low velocity expansion of the Lagrange multiplier at a boundary
μ	mass density parameter
$\boldsymbol{\nu},\ \nu_i$	integer population numbers in the generalized lattice gas model
ν_i^*	post-collisional integer population number
$\mathsf{\Pi},\ \Pi_{\alpha\beta}$	pressure or stress tensor
$\mathsf{\Pi}^*,\ \Pi_{\alpha\beta}^*$	post-collisional stress tensor
$\mathsf{\Pi}^{\text{eq}},\ \Pi_{\alpha\beta}^{\text{eq}}$	equilibrium (Euler) stress tensor
$\mathsf{\Pi}^{\text{neq}},\ \Pi_{\alpha\beta}^{\text{neq}}$	non-equilibrium (Newtonian) stress tensor
$\mathsf{\Pi}^{(0)}, \mathsf{\Pi}^{(1)}$	different orders of the Chapman-Enskog expansion of the stress tensor
ρ	density
$\sigma,\ \sigma_{\alpha\beta},\ \overline{\sigma_{\alpha\beta}}$	Navier-Stokes viscous stress tensor (overline denotes traceless part)
$\sigma^r,\ \sigma_{\alpha\beta}^r,\ \overline{\sigma}_{\alpha\beta}^r$	random fluctuating stress in the lattice Boltzmann model (overline denotes traceless part)
σ_2	coefficient of the rank-2 lattice sum
σ_4	coefficient of the isotropic term of the rank-4 lattice sum
τ	lattice Boltzmann time step
$\mathsf{\Phi}^{\text{eq}},\ \Phi_{\alpha\beta\gamma}^{\text{eq}}$	equilibrium value of the third-moment
$\mathsf{\Phi}^{(0)},\ \Phi_{\alpha\beta\gamma}^{(0)}$	lowest order Chapman-Enskog expansion of the third moment
φ_k	amplitude of the random noise for the k-th moment m_k
χ_i	indicator variable
χ	Lagrange multiplier
$\chi^{(1)},\ \chi^{(2)}$	low velocity expansion of the Lagrange multiplier
χ^B	Lagrange multiplier at a boundary

Continued on next page

Symbol/Notation	**Meaning**

Continued from previous page

$\chi^{B,(1)},\ \chi^{B,(2)}$	low velocity expansion of the Lagrange multiplier at a boundary
$\omega(\mathbf{v})$	weight function in kinetic theory
$\omega(m \to m^*)$	transition probability from m to m^* for stochastic collisions

Calligraphic letters	
$\mathcal{B}(\mathbf{v}' \to \mathbf{v})$	boundary scattering kernel in kinetic theory
$\mathcal{B}_{ij}$	discrete boundary scattering kernel
$\mathcal{C}$	linear collision operator in kinetic theory
$\mathcal{C}_{BGK}$	BGK collision operator in kinetic theory
$\mathcal{F}(\{f_i\})$	functional of the population numbers
$\mathcal{F}_B(\{f_i\})$	functional at a boundary
$\mathcal{H}^{(n)},\ \mathcal{H}^{(n)}_{\alpha_1 \ldots \alpha_n}$	n-th order Hermite tensor polynomial
$\mathcal{L},\ \mathcal{L}_{ij}$	discrete linear collision operator
$\mathcal{L}_M$	collision operator in mode space
$\mathcal{N}$	normalization factor

1 Introduction

Soft matter systems and in particular complex fluids are receiving ongoing research interest from both theoretical and experimental perspectives. Typical examples are colloidal dispersions and polymer solutions. More recently, evolved systems such as membranes, vesicles and cells have attracted growing interest. A decisive property of such systems is the presence of a hierarchy of time and length scales, ranging from the "atomistic" scale of the molecular interactions to the "macroscopic" scale of continuum hydrodynamics. The length scales in between are called "mesoscopic" and are pivotal for the rich diversity of phenomena that can be observed in soft matter. The additional mesoscopic length scale provided by particles dispersed in a solvent can lead to a qualitative change of the behavior in complex fluids compared to the pure solvent. For example, the addition of a small amount of polymer to water can lead to a significant reduction of drag when the solution is pumped through tubes or channels. Such phenomena are induced by the physics that takes place on the mesoscopic scale, which therefore can not be neglected in a theoretical description of complex fluids.

The behavior of soft matter is usually governed by a variety of different physical mechanisms, e.g., thermodynamics and phase transitions, electrokinetic and rheological effects, nonlinear phenomena and instabilities. A complete analytical treatment including all these mechanisms is certainly out of reach, and available theoretical predictions are generally based on more or less crude approximations. For this reason, soft matter and complex fluids belong to the realm of computer simulations. Simulations can serve as a powerful tool to investigate the principles underlying the experimental observations. They allow one to study simple model systems extensively and under well-defined conditions, where the influence of different interactions can be systematically identified and isolated. In this way, computer simulations form another pillar besides analytical theory and experiment, that is indispensable for gaining a better understanding of soft matter. The latter is clearly of paramount importance for practical applications, ranging from engineering processes to medical diagnosis and therapy. The development of successful and efficient computer simulation methods is therefore an important branch of contemporary physical research.

Efficient simulation models for complex fluids are mostly based on the concept of coarse-graining. The number of degrees of freedom in a complex fluid is so huge that an explicit treatment of all of them is impracticable. Moreover, the dynamics of the dispersed particles takes place on much longer time scales than the solvent dynamics, so that following every solvent molecule's trajectory is unnecessary and wastes a lot of computing time. This latter observation, which is the so-called separation of scales, is at the heart of the improvements achieved through coarse-graining. Since the molecular details of the solvent are irrelevant on the scales of the dispersed particles, it can be treated as a continuous hydrodynamic medium that is governed by the Navier-Stokes equation. Complex fluids can typically be described

in the incompressible and creeping flow limit, i.e., the Mach number and the Reynolds number are small. In this limit, the effect of the solvent on the dispersed particles can be modeled in terms of a Stokes friction. One of the simplest approaches to include solvent effects is Langevin dynamics, where each dispersed particle is subject to a friction force proportional to its velocity. The diffusion of the particles in Langevin dynamics is consistent with the Einstein relation. However, the simple friction force can not capture the momentum transport through the solvent. If a dispersed particle moves, the flow field of the solvent is changed. This perturbation propagates through the solvent and affects the motion of other particles. The correlations mediated by this effect are called *hydrodynamic interactions*. They are of a long-ranged nature and their efficient implementation in computer simulations is a topic still undergoing active development.

Among the techniques for simulating hydrodynamic interactions, the so-called mesoscopic methods have proven particularly useful for soft matter applications. They are based on representing the solvent degrees explicitly but via simplistic models on a mesoscopic scale. Particle based methods, such as dissipative particle dynamics and multi-particle collision dynamics, use explicit solvent particles that represent a collective lump of fluid. Besides particle methods, the lattice Boltzmann method has become a popular approach to simulate complex fluids. Lattice Boltzmann is built on a special space-time discretization of kinetic theory, where the solvent is modeled in terms of particle distributions on a regular lattice. Although it was originally devised as an alternative tool for computational fluid dynamics, it has successfully been applied to a variety of soft-matter systems. The reason for the success of the lattice Boltzmann method is its versatility originating from the possibility to pad the plain solvent dynamics with specific details describing the structure of complex fluids. Numerous examples show that lattice Boltzmann is a very flexible simulation method for many different kinds of systems, ranging from suspensions over reactive and multi-phase flows to turbulence.

Besides hydrodynamic interactions, *thermal fluctuations* play an important role in many soft matter systems. They arise from the underlying microscopic dynamics of the solvent molecules and are responsible for the observed Brownian motion of dispersed particles. On the hydrodynamic level, thermal fluctuations in the hydrodynamic variables can be described within the framework of fluctuating hydrodynamics. A simulation method for complex fluids should therefore be capable of reproducing the fluctuating hydrodynamic equations and provide a means for adjusting the strength of the thermal fluctuations at will. In the lattice Boltzmann method, the evolution of the hydrodynamic variables is completely deterministic, hence Brownian motion is not automatically reproduced. It is possible though, to equip the lattice Boltzmann method with thermal noise in a way that is consistent with fluctuating hydrodynamics. However, this approach lacks a rigorous justification in terms of the underlying statistical mechanics. In fact, some criticism of the original fluctuating lattice Boltzmann was put forward and a modified approach was suggested by Adhikari et al. [1]. The ultimate clarification of this debate was only achieved recently during the course of this thesis [2].

Another more recent development in soft matter research is an increasing trend of turning towards microfluidics. Modern experimental methods make it possible to fabricate microfluidic devices with dimensions on the micro- or even the nanoscale. The development of the first MEMS (micro-electromechanical systems) and μTAS (micro total analysis systems) has led to an ever-growing interest in the so-called lab-on-a-chip paradigm. Microchips are designed to displace, transport, manipulate and separate very small fluid volumes. The aim is to be able to conduct all typical elements of physico-chemical processing – mixing, reacting, analyzing and so on – on a microscopic scale. Potential applications emerge from the rapid developments in molecular biotechnology and include, for example, bioanalytic devices that can be used for medical diagnosis.

With the reduced spatial dimensions of microfluidic devices the surface-to-volume ratio is significantly increased. Therefore the flow through microchips is largely dominated by surface and interface effects and fluid-solid interactions play an important role. With regard to simulation methods this means that special emphasis has to be put on the *boundary conditions*. There is ample experimental evidence that the classical no-slip boundary condition becomes inappropriate in microchannels when the Knudsen number exceeds a value of roughly 0.1. The fluid does not stick to the boundary any more, but an effective slip velocity is observed that is determined by surface roughness and the physico-chemical properties of the boundary. While the no-slip boundary condition is the standard boundary condition in lattice Boltzmann simulations, apparent slippage has been rarely addressed. In general, boundary conditions in lattice Boltzmann simulations are often based on heuristic arguments, and a systematic and unified framework is still lacking.

The aim of this thesis is to advance the development of the lattice Boltzmann method in order to make it applicable to the flow of complex fluids in microfluidic devices. For this purpose, two major points shall be addressed. First, the fluctuating lattice Boltzmann equation shall be revisited in order to restore the statistical mechanics origin of the thermal fluctuations. This will make the method consistent with both statistical mechanics and continuum hydrodynamics at the same time. The second major topic shall be boundary conditions for the lattice Boltzmann model. A novel way of implementing a tunable slip boundary condition based on a mesoscopic wall friction model will be devised. Furthermore, a conceptually new approach to boundary conditions shall be explored that systematically treats the reduced symmetry at the wall. The purpose of the latter point is twofold: On one hand, we aim at a general local scheme for boundary conditions that can efficiently be implemented in a parallel computing environment. On the other hand we seek a better understanding of boundary conditions with respect to the symmetry breaking induced by the boundary. For both thermal fluctuations and boundary conditions, another goal is to develop an efficient and versatile implementation of the methods for use in the ESPResSo software package [3, 4].

The remainder of this work is structured as follows. In chapter 2, the lattice Boltzmann method is reviewed from a modern perspective. Instead of following the historical route starting out with lattice gas automata, we base the presentation on the kinetic nature of the lattice Boltzmann equation. In chapter 3, the statistical mechanics of the lattice Boltzmann equation will be developed. The connection between the lattice Boltzmann equation and

macroscopic hydrodynamics is established by asymptotic analysis in terms of the Chapman-Enskog expansion, which will be the subject of chapter 4. The following chapters are devoted to boundary conditions. In chapter 5, standard boundary conditions are reviewed, and a novel boundary condition for a tunable slip is devised. The conceptually new treatment of reduced symmetries at the boundary will be introduced and discussed in detail in chapter 6. Finally, chapter 7 closes with conclusions and discussion. The appendix contains details on the lattice Boltzmann implementation in the ESPResSo package. It also contains some of the more technical derivations which would have disturbed the flow of the main text.

2 The lattice Boltzmann method: a modern overview

This section introduces the theoretical background of the lattice Boltzmann method for fluid mechanics. In view of its development over the past twenty years and its popularity, it is not surprising that there is a vast corpus of existing literature on LB (see for example the book by Succi [5] or the reviews [6–8]). Nevertheless, some details in the understanding of the lattice Boltzmann method have only been worked out recently, e.g., the fluctuating lattice Boltzmann equation [2, 8], and some open questions still remain for example with respect to multiphase LB models. In the course of its evolution the theoretical insights into LB have led to a formal framework that is rather different from the original description related to lattice-gas automata. Therefore, it is useful to give an overview of the lattice Boltzmann theory from this modern perspective, which forms the basis for the further developments in this work. The material in this section is mainly a review of existing work. Nevertheless, some effort was put on a comprehensive and unified presentation of the lattice Boltzmann framework.

The remainder of this section is organized as follows: After some short historical remarks, the foundations of kinetic theory and its relation to continuum fluid mechanics will be introduced. Then it will be discussed how the Boltzmann equation can be systematically discretized, leading to an a-priori derivation of the lattice Boltzmann equation. Finally, the construction of lattice Boltzmann models will be outlined, and as a specific example, the D3Q19 model will be explained in detail.

2.1 Historical remarks

The historical roots of the lattice Boltzmann method lie in the lattice gas automata (LGA), a special class of cellular automata aiming at simulating fluid dynamics in terms of discrete microscopic models [9, 10]. They were inspired by the observation that the macroscopic flow behavior is similar for many fluids even when the microscopic structure is quite different. While the microscopic details may influence the dimensional values of transport coefficients, the form of the macroscopic hydrodynamic equations depends solely on symmetries and conservation laws. An early precursor of the LGA was introduced by Kadanoff and Swift [11] already in 1968. The first lattice gas model for fluids was proposed by Hardy, Pomeau and de Pazzis [12] and became known as the HPP model. It conserves mass and momentum and leads to sound waves, but it can not reproduce the Navier-Stokes equation because the underlying square lattice lacks sufficient rotational invariance. This issue was resolved in the FHP model by Frisch, Hasslacher and Pomeau [13] by using a triangular lattice with hexagonal symmetry. The FHP model was the first lattice gas automaton fluid

that successfully reproduces the Navier-Stokes equation, yet only in two dimensions. The development of three-dimensional lattice gas automaton fluids was first hindered by the fact that there is no lattice with sufficient symmetry which is at the same time space-filling [9]. Fortunately, the way out was found in four dimensions where the face-centered hypercubic (FCHC) lattice has sufficient symmetry [14]. The FCHC can be projected into three-dimensional space yielding a lattice gas automaton for three-dimensional Navier-Stokes hydrodynamics. The theoretical foundations of the LGA were presented, for example, by Frisch et al. [10] and Wolfram [15]. These works contain already essential insights that are also at the heart of the lattice Boltzmann method, namely, that besides conservation of mass and momentum, the isotropy of tensors up to fourth rank is required for Navier-Stokes behavior, which has to be guaranteed by a sufficient symmetry of the underlying lattice. Navier-Stokes behavior is obtained in the double limit of small Mach number and small Knudsen number [10, 16]. Concerning the symmetry properties, Hasslacher pointed out that instead of using high symmetry, one can also use multiple speeds corresponding to different neighbor shells of the lattice [see 9]. The isotropy of the lattice tensors in these model is achieved by introducing speed dependent weights for the different subshells. A first multi-speed model has been proposed by d'Humières, Lallemand and Frisch [14] as early as in 1986 – and turns out to be the LGA analogon of what is today used under the name D3Q19 model.

Lattice gas automata became very popular because, due to their Boolean nature, they provided an easy to implement and round-off free method to simulate fluid flows. They were, however, "plagued by several *diseases*" [9], the most severe of which being the lack of Galilean invariance and statistical noise [5]. To overcome the statistical noise inherent to the *Boolean occupation numbers*, one can introduce *ensemble-averaged populations*. This had already been done by Frisch et al. [10] to calculate the viscosity through linear response theory, which could be viewed as the first occurrence of a lattice Boltzmann equation. The first "real" lattice Boltzmann (LB) model, where the populations are used as dynamic variables, was put forward by McNamara and Zanetti [17]. In their model, the collision rules were derived from the microdynamics of the underlying LGA. A substantial simplification can be made by linearizing the collision operator around the equilibrium, as introduced by Higuera and Jimenez [18]. Not only did this reduce the complexity of the collision operator significantly, but it was also an important step in realizing the direct relation of lattice Boltzmann and continuum kinetic theory. An immediate follow-up is the single relaxation time approximation tantamount to the Bhatnagar-Gross-Krook (BGK) collision operator [19–21]. With the advent of the lattice BGK model, the lattice Boltzmann method had basically fledged into a self-standing form, the main ingredients of which are the local equilibrium distribution and the linear collision operator. In a 1992 seminal paper, Qian et al. [22] have presented a whole family of LB models and coined the now common nomenclature DnQm for n-dimensional models with m velocities.

Subsequently, many studies were conducted to gain a better understanding of the lattice Boltzmann method and to devise further refinements. One of them is the (re-)introduction of a multi-relaxation-time (MRT) collision operator to overcome the limitations of fixed Prandtl number and fixed ratio of bulk and shear viscosity in the lattice BGK model [23–

25]. The MRT model provides more flexibility to tune the macroscopic behavior of the lattice fluid and, on top of that, it is much more stable than lattice BGK [26]. In the view of this author, the MRT model is indeed the generic case of the lattice Boltzmann method, and lattice BGK is merely a special choice of relaxation parameters. Another important series of studies was concerned with the H-theorem in the lattice Boltzmann method [27–32]. It was shown by Wagner [27] that the usual polynomial form of the discrete local equilibrium distribution does not admit a H-theorem. Karlin et al. [28–30] have developed a framework to derive local equilibria from entropy functions in such a way that a discrete H-theorem can be proven. The approach has become known as *entropic lattice Boltzmann* and can be used to stabilize numerical algorithms [31]. More importantly, it has brought the conceptual advance that instead of prescribing an ansatz for the equilibrium distribution, it can be supported by an appropriate entropy function. Along a different line, Shan and Luo and their coworkers have established a systematic *a-priori* derivation of the lattice Boltzmann equation from continuum kinetic theory [33–35]. In this sense, the lattice Boltzmann equation can be seen as a fully discretized version of the continuum Boltzmann equation. This was a major finding that shows that the LB model is indeed more than just a plain Navier-Stokes solver and potentially led to a "conceptual shift in devising models of complex fluid behavior" [5].

The mentioned shift was also supported by many practical developments. A major push in modeling complex fluids were the simulations of particle suspensions pioneered by Ladd [36–38]. In order to account for Brownian motion, Ladd introduced fluctuations in the lattice Boltzmann equation. This was an important step towards linking the hydrodynamic variables to the statistical fluctuations of the lattice Boltzmann populations. It constitutes the starting point for applications of LB to various soft matter systems, where it is meanwhile one of the standard methods for hydrodynamic interactions [8]. A lot of interest is attracted to simulations of multiphase flows. The first multicomponent LB method was developed by Gunstensen et al. based on color-component lattice gas models [9]. In the Shan-Chen model, explicit interaction potentials are introduced to model interface forces between the phases [39, 40]. An improvement over these phenomenological approaches are the free energy models introduced by Swift et al. [41, 42]. They try to devise a local equilibrium distribution that is consistent with the thermodynamics of the interface. A fully consistent multiphase approach in terms of statistical mechanics is to date not available. Another active topic are thermal fluids where heat transport has to be modeled. It is well known that a quadratic equilibrium can not reproduce the heat transport equation. To this end, third-order terms were included by Alexander et al. [43], and Qian and Chen have used larger velocity sets to devise thermal LB models [44, 45]. However, these models suffer from the lack of a rigorous justification in terms of statistical mechanics.

Nowadays, it is well perceived that lattice Boltzmann is more than a Navier-Stokes solver. As a discrete kinetic scheme, it is capable of simulating behavior in the non-hydrodynamic regime up to Knudsen numbers as high as $Kn \sim \mathcal{O}(1)$ [46]. It is realized that the systematic derivation from the Boltzmann equation in terms of quadratures gives rise to a whole hierarchy of lattice Boltzmann models. Shan et al. have recently presented a systematic account of higher approximations to the Boltzmann equation beyond the Navier-Stokes level [47].

The framework can be used to devise LB models for thermohydrodynamics, which requires to employ higher level expansions and leads to larger velocity sets. Numerical results obtained by Ansumali et al. [48] within the entropic lattice Boltzmann method give strong support that the LB hierarchy is indeed capable of representing kinetic theory beyond the hydrodynamic regime.

In the following, we will pick up the spirit of these latest developments in the following and outline the lattice Boltzmann method as a fully discretized approximation to kinetic theory.

2.2 Kinetic theory and continuum fluid mechanics

Hydrodynamic interactions in a complex fluid are mediated by the flow field of the solvent, the time evolution of which can be described by the continuity equation and the Navier-Stokes equation[1] [49]

$$\begin{aligned} &\frac{\partial}{\partial t}\rho + \frac{\partial}{\partial \mathbf{r}} \cdot (\rho \mathbf{u}) = 0, \\ &\frac{\partial}{\partial t}(\rho \mathbf{u}) + \frac{\partial}{\partial \mathbf{r}} \cdot (\rho \mathbf{u} \otimes \mathbf{u}) = -\frac{\partial}{\partial \mathbf{r}} p + \frac{\partial}{\partial \mathbf{r}} \cdot \sigma + \mathbf{g}, \end{aligned} \tag{2.1}$$

where ρ is the mass density, $\rho\mathbf{u} = \mathbf{j}$ the momentum density, p the scalar pressure, σ the deviatoric (viscous) stress tensor, and $\mathbf{g}$ an external volumetric force. The deviatoric stress has the form

$$\sigma_{\alpha\beta} = \eta_{\alpha\beta\gamma\delta} \frac{\partial u_\gamma}{\partial r_\delta}, \tag{2.2}$$

where the viscosity tensor is given by

$$\eta_{\alpha\beta\gamma\delta} = \eta_s \left(\delta_{\alpha\gamma}\delta_{\beta\delta} + \delta_{\alpha\delta}\delta_{\beta\gamma}\right) + \left(\eta_b - \frac{2}{3}\eta_s\right) \delta_{\alpha\beta}\delta_{\gamma\delta} \tag{2.3}$$

with η_s and η_b being the shear and bulk viscosity of the fluid, respectively.

These equations describe the fluid as a continuum in the hydrodynamic limit, that is, on large length and time scales. Therefore we will refer to equations (2.1) as the *macroscopic* description of the fluid dynamics. The mass density ρ and the momentum density $\mathbf{j}$ describe the state of the fluid and are termed the macroscopic or hydrodynamic fields.

An important feature of the Navier-Stokes equation is its universality, i.e., it applies to a whole class of fluids and its structure is independent of the *microscopic* interactions that can be quite different, for example, in a liquid compared to a gas. The microscopic details of the dynamics are subsumed in the transport coefficients, in other words, the irrelevant degrees of freedom have been "projected out" and do not show up in the structure of the

[1] The full Navier-Stokes-Fourier description of continuum fluid mechanics includes also the heat transport equation, which we deliberately omit here because we will concentrate on an isothermal fluid.

macroscopic equations any more. This is an essential feature of continuum fluid mechanics because it means that, in order to simulate a solvent, we can use a wide range of micro-models[2], as long as they give rise to the Navier-Stokes equation on the macroscopic level and reproduce the correct values for the transport coefficients. However, such models have to be devised with care since it is not a-priori clear which are the basic physical properties they have to retain. The development of models that reproduce the relevant physics while ignoring irrelevant details is called *coarse-graining* and forms a field of vivid research in computational physics and chemistry.

A common approach in coarse-graining is to follow a "bottom-up" strategy: starting with the microscopic equations, simplified mesoscopic models are derived that reproduce the macroscopic dynamics. The microscopic dynamics of the particles (atoms or molecules) of a fluid is described by Newton's equation of motion. In principle, these equations can be solved for given initial conditions, for example by a molecular dynamics (MD) algorithm. However, this is usually not feasible for a fluid since the number of particles whose trajectories one has to compute is on the order of the Avogadro number $Av \sim 10^{23}$. Another problem is that such a system falls into the regime of Lyapunov instability, such that it is impossible to compute a deterministic trajectory due to the omnipresent round-off errors.[3] On the other hand, the observables used to describe the state of the fluid also depend on a large number of particles. The macroscopic properties are quite stable and insensitive to deviations in the initial conditions. This suggests that the global observables can be described as statistical averages over a large number of particle trajectories. The exact knowledge of the individual trajectories is not required any more, hence we can use a probabilistic picture to describe the motion of the particles itself. These assumptions are the basis of kinetic theory [50].

2.2.1 The Boltzmann equation

A central quantity in kinetic theory is the (single-particle) distribution function $f(\mathbf{r}, \mathbf{v}, t)$ depicting the probability to find a particle with velocity $\mathbf{v}$ around the point $\mathbf{r}$ at time t. That is, the quantity

$$f(\mathbf{r}, \mathbf{v}, t)\, d\mathbf{r}\, d\mathbf{v} \tag{2.4}$$

represents the mean number of molecules in the phase-space volume $d\mathbf{r}d\mathbf{v}$. The distribution function f is linked to the macroscopic observables by its moments, for example the mass, momentum and energy densities are given by

$$\begin{aligned} m \int f\, d\mathbf{v} &= \rho(\mathbf{r}, t), \\ m \int \mathbf{v} f\, d\mathbf{v} &= \rho \mathbf{u}(\mathbf{r}, t), \\ m \int \frac{\mathbf{v}^2}{2} f\, d\mathbf{v} &= \rho e(\mathbf{r}, t). \end{aligned} \tag{2.5}$$

[2] From this viewpoint, the micro-model is in principle arbitrary, but it will turn out that there are a number of prerequisites that have to be fulfilled in order to obtain Navier-Stokes behavior.

[3] In fact, it is impossible to specify the initial conditions precisely, and the computed trajectories always diverge exponentially from the exact solution.

The time evolution of the distribution function f is governed by the equation [50, 51]

$$\left(\frac{\partial}{\partial t} + \mathbf{v}\cdot\frac{\partial}{\partial \mathbf{r}} + \frac{\mathbf{F}}{m}\cdot\frac{\partial}{\partial \mathbf{v}}\right) f(\mathbf{r},\mathbf{v},t) = (\partial f)_{\text{coll}}\,, \tag{2.6}$$

where m is the mass of the particles and $\mathbf{F}$ an external force acting on them. The left hand side describes the streaming of the particles along their trajectories. The term $(\partial f)_{\text{coll}}$ on the right hand side describes the change of the distribution due to collisions between the particles. It is a shorthand notation and contains the information about the microscopic interactions between the particles. In principle, it contains the two-particle distribution function which itself is governed by a dynamical equation involving the three-particle distribution function and so on. This infinite hierarchy of equations is called the BBGKY hierarchy and stems from the Liouville equation for the full phase space probability density [50]. A closed form of equation (2.6) can be obtained when point particles are considered and the collisions between them are assumed to be binary and uncorrelated. This is the so-called *molecular chaos assumption* (or *Stosszahlansatz*) that leads to the celebrated Boltzmann equation

$$\begin{aligned}\left(\frac{\partial}{\partial t} + \mathbf{v}\cdot\frac{\partial}{\partial \mathbf{r}} + \frac{\mathbf{F}}{m}\cdot\frac{\partial}{\partial \mathbf{v}}\right) f(\mathbf{r},\mathbf{v},t) = \int d\mathbf{v}_1 \int d\Omega\, \sigma(\|\mathbf{v}_{\text{rel}}\|,\Omega)\,\|\mathbf{v}_{\text{rel}}\| \\ \times\left[f(\mathbf{r},\mathbf{v}',t) f(\mathbf{r},\mathbf{v}_1',t)) - f(\mathbf{r},\mathbf{v},t) f(\mathbf{r},\mathbf{v}_1,t)\right],\end{aligned} \tag{2.7}$$

where $\mathbf{v}_{\text{rel}} = \mathbf{v}_1 - \mathbf{v}$ is the relative velocity before the binary collision, $\sigma(v_{\text{rel}},\Omega)$ is the scattering cross section, and $\mathbf{v}'$ and $\mathbf{v}_1'$ are the post-collisional velocities characterized by the scattering angle Ω [see Ref. 50 for details]. In spite of the closed form, the integro-differential equation (2.7) is in general complicated to solve.

2.2.2 The Maxwell-Boltzmann equilibrium distribution

A pivotal role in kinetic theory is played by the local equilibrium distribution, that is, a solution of the Boltzmann equation which, in the absence of external forces, is independent of $\mathbf{r}$ and t, or equivalently, is a collisional invariant satisfying $(\partial f^{\text{eq}})_{\text{coll}} = 0$. From the collision term in the Boltzmann equation (2.7) we can deduce the condition of *detailed balance*

$$f(\mathbf{r},\mathbf{v}_1',t) f(\mathbf{r},\mathbf{v}_2',t) = f(\mathbf{r},\mathbf{v}_1,t) f(\mathbf{r},\mathbf{v}_2,t). \tag{2.8}$$

Detailed balance implies that the logarithm of f is an additive invariant, hence in thermodynamic equilibrium, $\ln f$ must be a linear combination of the collisional invariants

$$\ln f = \gamma_0 + \boldsymbol{\gamma}\cdot\mathbf{v} + \gamma_4 \mathbf{v}^2. \tag{2.9}$$

The parameters γ_i can be expressed in terms of the hydrodynamic fields which leads to the Maxwell-Boltzmann equilibrium distribution [50–52]

$$f^{\text{eq}}(\mathbf{v}) = \left(\frac{m}{2\pi k_B T}\right)^{\frac{3}{2}} \frac{\rho}{m}\,\exp\left[-\frac{m(\mathbf{v}-\mathbf{u})^2}{2k_B T}\right], \tag{2.10}$$

where k_B is the Boltzmann constant.

2.2.3 The linearized Boltzmann equation

In practice, it is usually suitable to assume that the actual distribution function deviates only slightly from the local equilibrium distribution. In this case, we have

$$f = f^{\text{eq}} + f^{\text{neq}}, \qquad \frac{f^{\text{neq}}}{f^{\text{eq}}} \ll 1, \tag{2.11}$$

and the collision term can be linearized around the equilibrium. Thus we get the linearized Boltzmann equation

$$\left(\frac{\partial}{\partial t} + \mathbf{v} \cdot \frac{\partial}{\partial \mathbf{r}} + \frac{\mathbf{F}}{m} \cdot \frac{\partial}{\partial \mathbf{v}} \right) f(\mathbf{r}, \mathbf{v}, t) = \mathcal{C} f^{neq}, \tag{2.12}$$

where $\mathcal{C}$ is a linear operator. It can be shown that the operator $\mathcal{C}$ is self-adjoint with respect to the scalar product

$$\langle g | h \rangle = \int \frac{1}{f^{\text{eq}}(v)} g^*(\mathbf{v}) h(\mathbf{v}) \, d\mathbf{v}, \tag{2.13}$$

and its eigenvalues are negative or equal to zero. It is clear from the definition that $\mathcal{C}$ has the degenerate eigenvalue zero corresponding to the collisional invariants.

The simplest form of the linear collision operator is the BGK approximation (after Bhatnagar, Gross and Krook [19]), which assumes a collision frequency λ such that during the time interval dt a fraction $\lambda \, dt$ of particles is relaxed to equilibrium. The collision operator then becomes

$$\mathcal{C}_{\text{BGK}} f^{\text{neq}} = -\lambda f^{\text{neq}}. \tag{2.14}$$

This expression is much easier to treat in analytical calculations and many lattice Boltzmann models are based on the BGK approximation. We will use the BGK approximation in some of the derivations that follow in order to keep the formal presentation simple. In most cases, a general linear operator does not introduce additional complications and can be treated along the same lines.

2.2.4 Hydrodynamic fields and macroscopic equations

We have already seen that the connection between the kinetic level and the hydrodynamic level is obtained by calculating moments of the distribution function f. In general, a moment is given by

$$m_\psi(\mathbf{r}, t) = \int \psi(\mathbf{v}) f(\mathbf{r}, \mathbf{v}, t) \, d\mathbf{v}, \tag{2.15}$$

where $\psi(\mathbf{v})$ is a polynomial in the components of $\mathbf{v}$. The integration over velocity space is essentially an averaging process which reflects the statistical nature underlying the kinetic theory picture. The equations of motion for these averages, i.e., the macroscopic dynamics,

are the local conservation laws obtained by multiplying the Boltzmann equation (2.7) with the collisional invariants. This yields

$$\begin{aligned} \frac{\partial \rho}{\partial t} + \frac{\partial \mathbf{j}}{\partial \mathbf{r}} &= 0, \\ \frac{\partial \mathbf{j}}{\partial t} + \frac{\partial \Pi}{\partial \mathbf{r}} &= \frac{\rho}{m}\mathbf{F}, \end{aligned} \tag{2.16}$$

where $\mathbf{j} = \rho\mathbf{u}$ and the pressure tensor

$$\Pi(\mathbf{r}, t) = \int (\mathbf{v} \otimes \mathbf{v}) f(\mathbf{r}, \mathbf{v}, t)\, d\mathbf{v} \tag{2.17}$$

was introduced. This tensor describes the flow of macroscopic momentum due to microscopic motion of the particles. Its equilibrium value can be computed from the Maxwell-Boltzmann distribution (2.10)

$$\Pi^{\text{eq}} = \int (\mathbf{v} \otimes \mathbf{v}) f^{\text{eq}}\, d\mathbf{v} = p\mathbb{1} + \rho\mathbf{u} \otimes \mathbf{u}, \tag{2.18}$$

where $p = \rho\frac{k_B T}{m}$ is the scalar thermodynamic pressure. Since $\mathbf{v} \otimes \mathbf{v}$ is not a collisional invariant, the pressure tensor has a non-equilibrium contribution which has to be determined from the distribution function f. This means that without explicit knowledge of the distribution function we can not obtain a closed form of the equation system (2.16) in terms of ρ and $\mathbf{u}$. Comparison with the Navier-Stokes equation (2.1) shows that we have to require

$$\Pi = \Pi^{\text{eq}} + \Pi^{\text{neq}} = p\mathbb{1} + \rho\mathbf{u} \otimes \mathbf{u} - \sigma. \tag{2.19}$$

The missing link is an explicit expression for the non-equilibrium pressure tensor Π^{neq} which has to match the viscous stress tensor $-\sigma$. What remains to be done to come full circle is to find a closure for the equation system (2.16). This is usually done with certain approximations. In the Chapman-Enskog method, a closure is obtained by expressing the distribution function f and the higher moments in terms of ρ and $\mathbf{u}$ and their gradients. It is based on the assumption that these macroscopic variables vary on scales much larger then the characteristic microscopic scales (limit of small Knudsen number). The Chapman-Enskog method will be explained in detail in chapter 4. An alternative approach is to expand the distribution function in Hermite polynomials. Such an expansion was used by Grad [53] to obtain partial differential equations for the 13 hydrodynamically significant moments. Since there is a close connection between the lattice Boltzmann method and the Hermite expansion, we sketch the procedure in the following paragraphs. A systematic non-perturbative procedure was presented by Levermore [54] which leads to a whole hierarchy of closed systems. However, since we are only interested in the Navier-Stokes behavior, we will not discuss this approach further.

2.2.5 Dimensionless formulation

For what follows, it will be useful to non-dimensionalize the Boltzmann equation. The absence of physical units is also needed for the implementation of a computer algorithm.

In order to remove the units, we introduce a length scale l_0 and a time scale t_0. We choose them in such a way that $l_0/t_0 = c_0$ is a characteristic velocity. The form of the Maxwell-Boltzmann distribution suggests to use $c_0 = \sqrt{k_B T_0/m}$ which is the speed of sound in the fluid at a characteristic temperature T_0. The dimensionless ratio of the actual temperature T and T_0 will be denoted by θ. With these definitions, the Maxwell-Boltzmann distribution comes as

$$f^{\mathrm{eq}}(\mathbf{v}) = \frac{\rho}{(2\pi\theta)^{3/2}} \exp\left[-\frac{(\mathbf{v}-\mathbf{u})^2}{2\theta}\right] \tag{2.20}$$

where $\mathbf{v}$ and $\mathbf{u}$ have been made dimensionless by scaling with c_0, and ρ by scaling with m/l_0^3, respectively. With the above choice of the characteristic scales the Boltzmann-BGK equation keeps the same form when λ is understood as a dimensionless relaxation frequency. For the rest of this section, we will stick to the dimensionless formulation which has the advantage that the moments have the same units.

2.2.6 Hermite-Expansion

According to Grad [53], the distribution function f can be expanded in the basis of the Hermite polynomials as

$$f(\mathbf{r},\mathbf{v},t) = \omega(\mathbf{v}) \sum_{n=0}^{\infty} \frac{1}{n!} \mathsf{a}^{(n)}(\mathbf{r},t) \mathcal{H}^{(n)}(\mathbf{v}), \tag{2.21}$$

where the weight function $\omega(\mathbf{v})$ is given by

$$\omega(\mathbf{v}) = (2\pi)^{-\frac{3}{2}} \exp\left[-\frac{\mathbf{v}^2}{2}\right]. \tag{2.22}$$

Here, both the Hermite polynomials $\mathcal{H}^{(n)}$ (cf. appendix B.1) and the coefficients $\mathsf{a}^{(n)}$ are tensors of order n. The latter are given by

$$\mathsf{a}^{(n)}(\mathbf{r},t) = \int \mathcal{H}^{(n)}(\mathbf{v}) f(\mathbf{r},\mathbf{v},t)\, d\mathbf{v}. \tag{2.23}$$

The motivation behind this expansion is that the coefficients $\mathsf{a}^{(n)}(\mathbf{r},t)$ are linear combinations of the moments, i.e., the lower order expansion coefficients are directly related to the hydrodynamic variables by the following identities:

$$\begin{aligned} \mathsf{a}^{(0)} &= \int f\, d\mathbf{v} = \rho, \\ \mathsf{a}^{(1)} &= \int \mathbf{v} f\, d\mathbf{v} = \rho\mathbf{u}, \\ \mathsf{a}^{(2)} &= \int (\mathbf{v}\otimes\mathbf{v} - \mathbb{1})\, f\, d\mathbf{v} = \Pi - \rho\mathbb{1}. \end{aligned} \tag{2.24}$$

The hydrodynamic variables of the Navier-Stokes level are completely determined by the first coefficients. Consequently, the macroscopic equations can be represented by partial differential equations for the coefficients $\mathsf{a}^{(n)}$. From equation (2.21) and (2.23) it follows that a truncation of the Hermite expansion at a certain order does not change the expansion coefficients up to that order, because the Hermite polynomials are mutually orthogonal. This means that we can approximate the distribution function by the first N Hermite polynomials

$$f^N(\mathbf{r}, \mathbf{v}, t) = \omega(\mathbf{v}) \sum_{n=0}^{N} \frac{1}{n!} \mathsf{a}^{(n)}(\mathbf{r}, t) \mathcal{H}^{(n)}(\mathbf{v}) \tag{2.25}$$

without changing the moments up to order N, i.e., the conservation equations for the lower order moments are not affected by the truncation.[4] The gain of the truncation is that the system of partial differential equations for the $\mathsf{a}^{(n)}$ is now determined and can be used to obtain a closed set of hydrodynamic equations. In this way, Grad used the third order approximation to obtain his 13-moment system.[5] Because of its special properties, the Hermite expansion is also particularly well suited for further discretizations. This will be used in the next section to derive the lattice Boltzmann equation from the linear Boltzmann equation.

2.3 Discretization of the Boltzmann equation

Having achieved a closure of the moment equation system, we now turn to the problem of discretizing the Boltzmann equation in order to make it accessible for a computer simulation. For the sake of simplicity, we focus on the Boltzmann-BGK equation without a forcing term in this section

$$\frac{\partial}{\partial t} f + \mathbf{v} \cdot \frac{\partial}{\partial \mathbf{r}} f = -\lambda \left(f - f^{\mathrm{eq}} \right). \tag{2.26}$$

We seek a discrete representation of phase space and time $(\mathbf{r}, \mathbf{v}, t)$ such that (2.26) can be turned into a finite-difference scheme where all quantities are evaluated at discrete points. We will do this stepwise by first discretizing velocity space by means of a Gauss-Hermite quadrature. This yields a so-called discrete velocity model (DVM). If chosen appropriately, the abscissae of the quadrature naturally lead to a discretization of configuration space in form of a regular lattice. Finally, the derivatives in the Boltzmann equation are replaced by finite differences. This route of discretizing the Boltzmann equation follows the work of Shan and He [35], Shan et al. [47]. The Gauss-Hermite quadrature was already used earlier by He and Luo [33, 34] to derive the lattice Boltzmann equation in a slightly different way than the one presented here.

[4] The truncated terms may, however, affect the dynamics of the hydrodynamic variables. Therefore the truncation is really an approximation whose validity will be justified later.

[5] In principle, there are 20 moments for the third order approximation, but only thirteen are considered hydrodynamically significant in Grad's moment system.

2.3.1 Discretization of velocity space

In the following, we assume that the distribution function can be approximated by its truncated Hermite expansion f^N. We will see later that this approximation corresponds to the limit where the typical hydrodynamic flow velocity u is small compared to the speed of sound c_s, that is, the *Mach number* $Ma = u/c_s$ is small. The expansion coefficients are then given by

$$\mathsf{a}^{(n)} = \int \mathcal{H}^{(n)}(\mathbf{v}) f^N(\mathbf{r}, \mathbf{v}, t)\, d\mathbf{v} = \int \omega(\mathbf{v})\, p(\mathbf{r}, \mathbf{v}, t)\, d\mathbf{v}. \tag{2.27}$$

Since $p = \frac{\mathcal{H}^{(n)} f^N}{\omega}$ is a polynomial of degree less than or equal to $2N$, we can use the Gauss-Hermite quadrature explained in appendix B.1 to calculate the integral in (2.27) using the values of p at a set of discrete velocities $\mathbf{c}_i$

$$\mathsf{a}^{(n)} = \sum_i w_i\, p(\mathbf{r}, \mathbf{c}_i, t) = \sum_i w_i\, \frac{\mathcal{H}^{(n)}(\mathbf{c}_i) f^N(\mathbf{r}, \mathbf{c}_i, t)}{\omega(\mathbf{c}_i)}. \tag{2.28}$$

The nodes $\mathbf{c}_i$ and the weights w_i are given by the quadrature and depend on the chosen degree of the quadrature. We now introduce

$$f_i(\mathbf{r}, t) = \frac{w_i f^N(\mathbf{r}, \mathbf{c}_i, t)}{\omega(\mathbf{c}_i)} \tag{2.29}$$

which are functions of space and time only. It is important to note that the truncated distribution function f^N is completely determined by the f_i. Therefore, without approximation the hydrodynamic variables can be written as

$$\begin{aligned} \rho = \mathsf{a}^{(0)} &= \sum_i f_i, \\ \rho\mathbf{u} = \mathsf{a}^{(1)} &= \sum_i f_i \mathbf{c}_i, \\ \Pi = \mathsf{a}^{(2)} + \mathsf{a}^{(0)}\mathbb{1} &= \sum_i f_i \mathbf{c}_i \otimes \mathbf{c}_i \end{aligned} \tag{2.30}$$

This is already the form of the hydrodynamic fields that will be used in the lattice Boltzmann method.[6]

The Maxwell-Boltzmann equilibrium distribution can be expanded in Hermite polynomials in the same way. This is necessary because f^{eq} has non-zero Hermite coefficients at all orders, such that the conservation laws for the collisional invariants hold exactly only when f^{eq} is truncated similarly to f [35]. This corresponds to a projection into the subspace spanned by the Hermite polynomials up to the respective order. Replacing f^N by f^{eq} in

[6] From now on, we will skip the symbol $\otimes$ in the tensor product. In the opinion of the author, this makes the structure of the formulas more visible to the reader's eye. For the more mathematically inclined readers, please accept this apology for the lack of notational rigor.

equation (2.27) and inserting into (2.21) we get the Hermite expansion of the Maxwellian. Up to second order it reads

$$f^{\mathrm{eq},(2)}(\mathbf{v}) \approx \omega(\mathbf{v})\rho\left[1 + \mathbf{u}\cdot\mathbf{v} + \frac{1}{2}\mathbf{uu} : (\mathbf{vv} - \mathbb{1}) + \frac{1}{2}(\theta - 1)(\mathbf{v}^2 - D)\right]. \tag{2.31}$$

For an isothermal system $\theta = 1$ and the last term in the brackets vanishes. Then this expression is equal to the Taylor expansion of f^{eq} up to terms of order $\mathbf{u}^2$. The latter has been used by He and Luo [33, 34] to derive the lattice Boltzmann equation on a slightly different route. In analogy to (2.29), we introduce the equilibrium distribution for the discrete velocities

$$f_i^{\mathrm{eq}} = \frac{w_i f^{\mathrm{eq},(2)}(\mathbf{c}_i)}{\omega(\mathbf{c}_i)} = w_i\rho\left[1 + \mathbf{u}\cdot\mathbf{c}_i + \frac{1}{2}\mathbf{uu} : (\mathbf{c}_i\mathbf{c}_i - \mathbb{1})\right]. \tag{2.32}$$

Finally, we formulate the Boltzmann-BGK equation in terms of the discrete velocity distributions f_i. Taking equation (2.26) at $\mathbf{c}_i$ and multiplying again with $w_i/\omega(\mathbf{c}_i)$ we arrive at

$$\frac{\partial}{\partial t} f_i + \mathbf{c}_i \cdot \frac{\partial}{\partial \mathbf{r}} f_i = -\lambda(f_i - f_i^{\mathrm{eq}}). \tag{2.33}$$

This set of differential equations in space and time constitutes a discrete velocity model (DVM) and represents an approximation to the continuous Boltzmann-BGK equation [55, 56]. The moments ρ and $\mathbf{u}$ are preserved, an important feature if one aims at the hydrodynamic limit of the kinetic equations. It is also possible to preserve higher moments by going to higher order Hermite approximations. The relation of the DVM and the lattice Boltzmann equation has been emphasized by Luo [57], in particular with respect to some rigorous results concerning thermodynamics and the H-theorem. In this context it should be noted that the positivity of the Maxwell-Boltzmann distribution is sacrificed in the finite Hermite expansion. This is of relevance when the stability of the lattice Boltzmann method is concerned. Even more important, it has the consequence that no H-theorem exists for the truncated equilibrium distribution, i.e., it is not guaranteed that any initial distribution will converge to the equilibrium distribution. For this reason, the latter is sometimes also termed *pseudo-equilibrium*. The lack of an H-theorem has motivated alternative approaches which have led to the development of the entropic lattice Boltzmann models [28–30, 32]. The concept of entropy will be used later in this work when we discuss the statistical mechanics of the lattice Boltzmann equation, cf. chapter 3. We close this section here by quoting the unscaled discrete equilibrium distribution

$$f_i^{\mathrm{eq}} = w_i\rho\left[1 + \frac{\mathbf{u}\cdot\mathbf{c}_i}{c_s^2} + \frac{\mathbf{uu} : (\mathbf{c}_i\mathbf{c}_i - \mathbb{1})}{2c_s^4}\right], \tag{2.34}$$

where $c_s = \sqrt{k_B T/m}$ is the isothermal speed of sound. Note that we have deliberately made the transition to a mass density here, whereas before the distribution functions where number densities.

2.3.2 Discretization of configuration space and time

In order to discretize space and time, we rewrite the discrete velocity equation (2.33) as an ordinary differential equation

$$\frac{df_i}{dt} + \lambda f_i = \lambda f_i^{\mathrm{eq}}. \tag{2.35}$$

This can be formally integrated over a time τ to give

$$f_i(\mathbf{r} + \tau\mathbf{c}_i, t + \tau) = e^{-\lambda\tau} f_i(\mathbf{r}, t) + \lambda e^{-\lambda\tau} \int_0^\tau e^{\lambda t'} f_i^{\mathrm{eq}}(\mathbf{r} + t'\mathbf{c}_i, t + t')\, dt'. \tag{2.36}$$

For small τ, we can to a first approximation write

$$f_i^{\mathrm{eq}}(\mathbf{r} + t'\mathbf{c}_i, t + t') = f_i^{\mathrm{eq}}(\mathbf{r}, t) + t' \frac{f_i^{\mathrm{eq}}(\mathbf{r} + \tau\mathbf{c}_i, t + \tau) - f_i^{\mathrm{eq}}(\mathbf{r}, t)}{\tau} + \mathcal{O}(\tau^2). \tag{2.37}$$

Expanding the exponential as well and neglecting all terms of order $\mathcal{O}(\tau^2)$, we arrive at

$$f_i(\mathbf{r} + \tau\mathbf{c}_i, t + \tau) = f_i(\mathbf{r}, t) - \lambda \left[f_i(\mathbf{r}, t) - f_i^{\mathrm{eq}}(\mathbf{r}, t) \right]. \tag{2.38}$$

This equation is now fully discrete in phase space and time. It is to be noted that the use of the BGK approximation does not mean a loss of generality here. The result of the full discretization for the Boltzmann equation with a general linear operator reads

$$f_i(\mathbf{r} + \tau\mathbf{c}_i, t + \tau) = f_i(\mathbf{r}, t) + \mathcal{L}_{ij} \left[f_j(\mathbf{r}, t) - f_j^{\mathrm{eq}}(\mathbf{r}, t) \right]. \tag{2.39}$$

Equation (2.39) is nothing but the famous lattice Boltzmann equation (LBE). The derivation shows that it is merely a finite difference approximation to the continuous Boltzmann equation. In particular, the expression for the equilibrium distribution is a result of the projection onto the lower order Hermite polynomials and the weights w_i are *a priori* known through the choice of the quadrature. This is in contrast to the alternative approach where the weights are determined within the Chapman-Enskog expansion such that the correct hydrodynamic equations are obtained. What remains to be done at this stage in order to complete the development of the lattice Boltzmann method is to choose an appropriate quadrature to obtain the discrete velocities $\mathbf{c}_i$ and the corresponding weights w_i.

2.3.3 Choice of truncation and quadrature

As pointed out in the previous section, the truncated Hermite expansion f^N preserves the moments up to order N. In what follows, we will use the second order truncation which preserves ρ, $\mathbf{j}$ and Π. We will only be concerned with isothermal models, hence the heat flux contained in the third moment is not of primary interest. However, in the Chapman-Enskog analysis it turns out that the third moment enters the dynamics of the pressure tensor,

Quadrature	LB model	q	b_q	w_q	$\mathbf{c}_q$
$E^3_{1,5}$	D1Q3	0	1	$\frac{2}{3}$	0
		1	2	$\frac{1}{6}$	$\pm\sqrt{3}$
$E^9_{2,5}$	D2Q9	0	1	$\frac{4}{9}$	$(0,0)$
		1	4	$\frac{1}{9}$	$(\pm\sqrt{3},0),(0,\pm\sqrt{3})$
		2	4	$\frac{1}{36}$	$(\pm\sqrt{3},\pm\sqrt{3})$
$E^{15}_{3,5}$	D3Q15	0	1	$\frac{2}{9}$	$(0,0,0)$
		1	6	$\frac{1}{9}$	$(\pm\sqrt{3},0,0),(0,\pm\sqrt{3},0),(0,0,\sqrt{3})$
		3	8	$\frac{1}{72}$	$(\pm\sqrt{3},\pm\sqrt{3},\pm\sqrt{3})$
$E^{19}_{3,5}$	D3Q19	0	1	$\frac{1}{3}$	$(0,0,0)$
		1	6	$\frac{1}{18}$	$(\pm\sqrt{3},0,0),(0,\pm\sqrt{3},0),(0,0,\sqrt{3})$
		2	12	$\frac{1}{36}$	$(\pm\sqrt{3},\pm\sqrt{3},0),(\pm\sqrt{3},0,\pm\sqrt{3}),(0,\pm\sqrt{3},\pm\sqrt{3})$
$E^{27}_{3,5}$	D3Q27	0	1	$\frac{8}{27}$	$(0,0,0)$
		1	6	$\frac{2}{27}$	$(\pm\sqrt{3},0,0),(0,\pm\sqrt{3},0),(0,0,\sqrt{3})$
		2	12	$\frac{1}{54}$	$(\pm\sqrt{3},\pm\sqrt{3},0),(\pm\sqrt{3},0,\pm\sqrt{3}),(0,\pm\sqrt{3},\pm\sqrt{3})$
		3	8	$\frac{1}{216}$	$(\pm\sqrt{3},\pm\sqrt{3},\pm\sqrt{3})$

Table 2.1: Gauss-Hermite quadratures of degree 5 in different dimensions and the corresponding lattice Boltzmann models. Following Shan et al. [47], the naming convention $E^n_{D,d}$ denotes a degree-d quadrature in D dimensions with n abscissae. The vectors $\mathbf{c}_q$ with the same value of $q = \|\mathbf{c}_i\|^2/3$ form a symmetry class within which the weight w_q does not vary. By scaling the $\mathbf{c}_q$ with $\sqrt{3}$ sublattices of the standard cubic lattice are obtained.

such that in principle the second order truncation is not sufficient to reproduce the Navier-Stokes equation [47]. The error is, however, of order $\mathcal{O}(Ma^3)$ and can be neglected.[7] The calculation of the Hermite coefficients of the Nth order expansion involves polynomials up to degree $2N$. The quadrature therefore must have a degree $n \geq 2N$, i.e., for the second order approximation we need a quadrature of degree $n \geq 4$. In addition to the accurateness, the quadrature is required to induce a regular lattice, i.e., the nodes $\mathbf{c}_i$ should leave the spatial grid invariant under the transformation $\mathbf{r} \to \mathbf{r} + \tau\mathbf{c}_i$. For details on the production rules for three dimensional quadratures, we refer to appendix B.1. It turns out that some of the commonly used lattice Boltzmann models, e.g., the D2Q9, D3Q19 and D3Q27 models after the naming convention of Qian et al. [22], stem from a degree-5 quadrature. The corresponding quadratures are listed in table 2.1, where we follow the nomenclature of Shan et al. [47]. An important property of the quadratures is that they automatically imply isotropy of the lattice tensors of rank up to the degree of the quadrature [59]

$$T^{(n)} = \sum_i w_i \mathbf{c}_i \dots \mathbf{c}_i = \begin{cases} 0 & n \text{ odd} \\ \boldsymbol{\delta}^{(n)} & n \text{ even} \end{cases}, \quad \forall n \leq m. \tag{2.40}$$

This is an important requirement to obtain hydrodynamic behavior. If instead of a quadrature an ad-hoc ansatz for the lattice model is used, the weights w_i have to be determined such

[7] The second order truncation leads to errors in the viscosity of order $\mathcal{O}(\mathbf{u}^2)$ which are related to incomplete Galilean invariance of the higher moments [58, 59].

that isotropy of the lattice tensors up to a certain rank is satisfied. This approach is worked out in more detail in section 2.5 for the D3Q19 model. It should be remarked that a lattice Boltzmann model does not necessarily need to correspond to an exact quadrature. For example, the D3Q13 model does not stem from a quadrature and still obeys isotropy and Galilean invariance [24]. However, the models related to quadratures usually yield better accuracy and stability [60].

All the quadratures listed in table 2.1 yield regular cubic lattices because their velocities can be expressed by integer multiples of a common constant, i.e., $\sqrt{3}$. In general, the ratio of the velocities of a quadrature can be irrational such that they do not connect the nodes of a simple lattice. This holds in particular for most higher degree quadratures, e.g., $E^{27}_{3,7}$, which have to be considered for thermo-hydrodynamics, or the Burnett level momentum flow. In this case, the simple lattice Boltzmann method can not be used and one has to resort to more complex algorithms, for example the interpolation scheme by He et al. [61] or the volumetric scheme by Chen [62].

Having obtained an explicit quadrature, the lattice Boltzmann equation is fully specified and an algorithm can in principle be implemented. The derivation presented here is systematic and shows the underlying kinetic nature of the lattice Boltzmann method. However, when working on algorithms for computer simulations, it is useful to also have a more vivid understanding of the physics behind the equations. In the next section, we will therefore depict the lattice Boltzmann equation from a more algorithmic view in order to demonstrate how it works.

2.4 The lattice Boltzmann equation

As we have seen, the lattice Boltzmann equation is in essence a systematically discretized form of the continuous Boltzmann equation. However, to be algorithmically useful, the discretizations of phase space and time should be chosen coherently in such a manner that the discrete velocities form the links of a regular lattice. Then we can interpret the discrete distribution functions $f_i(\mathbf{r}, t)$ as quantities assigned to a lattice site $\mathbf{r}$ at time t. We will refer to the f_i as populations of the lattice site $\mathbf{r}$ in this context. The lattice Boltzmann equation describes the dynamic evolution of these populations, which can be written in two parts as

$$f_i(\mathbf{r} + \tau\mathbf{c}_i, t + \tau) = f_i^*(\mathbf{r}, t) = f_i(\mathbf{r}, t) + \Delta_i\left(\mathbf{f}(\mathbf{r}, t)\right) \tag{2.41}$$

These two parts can be illustrated as follows: the transition from f_i to f_i^* is an instantaneous local process where the populations are redistributed among the different velocity directions according to the operator Δ_i. This process is due to the collisions at the microscopic level and is therefore called the *collision step*. The second part of the lattice Boltzmann equation is the *streaming step* which assigns the post-collisional population f_i^* at $\mathbf{r}$ and time t to the new population at $\mathbf{r} + \tau\mathbf{c}_i$ and time $t + \tau$. One can think of the streaming as a whole population moving along a link $\mathbf{c}_i$ of the lattice, but this should not be confused with the

real motion of single particles. The information about the latter is not included in the kinetic picture and can not be recovered at this level. The lattice Boltzmann equation is thus a truly mesoscopic approach to fluid dynamics, as it contains more information than just the macroscopic Navier-Stokes description while at the same time not representing the full microscopic degrees of freedom. The combination of collisions and streaming leads to an update scheme for the whole lattice. Because of the simple structure and the locality of most of the operations, the scheme is very well suited for highly efficient implementations on parallel computing platforms. We will deal with implementing an efficient lattice Boltzmann kernel in some detail in appendix A.

The equilibrium distribution appearing in equation (2.39) was derived in the previous section from the standard Maxwell-Boltzmann equilibrium, based on the systematic discretization of velocity space. As pointed out by d'Humières [23], it is also admissible to consider the set of discrete velocities as a free choice and use the form of the equilibrium distribution as an *ansatz* with certain parameters that allow the model to be tuned for specific properties. Basically, this means that the type of lattice can be chosen with respect to certain constraints. The introduced free parameters in the equilibrium distribution correspond to choosing the equilibrium values of the conserved and non-conserved moments.[8] Similarly, the collision operator Δ_i provides some freedom through its eigenvalues which are unspecified parameters so far. Since we aim at constructing a mesoscopic model for fluid mechanics, the choice of these parameters is dictated by the intrinsic properties of the Navier-Stokes equation, that is, the symmetries and conservation laws of macroscopic fluid mechanics.

While the explicit link will be established within the Chapman-Enskog expansion, we anticipate the results of chapter 4 at this point and list the requirements that the construction of a lattice Boltzmann model has to meet:

1. The lower moments corresponding to hydrodynamic fields have to satisfy
   - mass conservation:
   $$\sum_i f_i = \sum_i f_i^{\mathrm{eq}} = \rho, \tag{2.42}$$
   - momentum conservation:
   $$\sum_i f_i \mathbf{c}_i = \sum_i f_i^{\mathrm{eq}} \mathbf{c}_i = \rho \mathbf{u}, \tag{2.43}$$
   - Euler form of equilibrium stress:
   $$\sum_i f_i^{\mathrm{eq}} \mathbf{c}_i \mathbf{c}_i = p \mathbb{1} + \rho \mathbf{u}\mathbf{u}. \tag{2.44}$$

[8] It can be shown that the form of the hydrodynamic equations is completely determined by the lower moments of the equilibrium distribution [59]. The number of free parameters depends on the order at which the Hermite expansion is truncated. This determines also the order to which the equilibrium moments agree with those of the Maxwell-Boltzmann distribution.

2. The collision operator Δ_i must obey
   - mass conservation:
   $$\sum_i \Delta_i = 0, \tag{2.45}$$
   - momentum conservation:
   $$\sum_i \Delta_i \mathbf{c}_i = 0, \tag{2.46}$$
   - Navier-Stokes form of the deviatoric momentum flux:
   $$\sum_i (f_i^{*\mathrm{neq}} + f_i^{\mathrm{neq}}) \mathbf{c}_i \mathbf{c}_i = -2\sigma. \tag{2.47}$$

Before we proceed to the systematic derivation of these requirements, we illustrate them in terms of the D3Q19 model, which is one of the most popular lattice Boltzmann models for simulating complex fluids.

2.5 The D3Q19 model

2.5.1 Equilibrium distribution

The D3Q19 model is based on a three-dimensional regular cubic lattice and the set of 19 discrete velocities consists of the zero velocity vector, the six nearest neighbors and twelve next-nearest neighbors on the cubic lattice, see figure 2.1. The corresponding dimensionless lattice vectors $\hat{\mathbf{c}}_i$ are the columns of the matrix

$$\mathsf{C} = \begin{pmatrix} 0 & 1 & -1 & 0 & 0 & 0 & 0 & 1 & -1 & 1 & -1 & 1 & -1 & 1 & -1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & -1 & 0 & 0 & 1 & -1 & -1 & 1 & 0 & 0 & 0 & 0 & 1 & -1 & 1 & -1 \\ 0 & 0 & 0 & 0 & 0 & 1 & -1 & 0 & 0 & 0 & 0 & 1 & -1 & -1 & 1 & 1 & -1 & -1 & 1 \end{pmatrix}$$

and the velocity vectors are $\mathbf{c}_i = \hat{\mathbf{c}}_i \, a/\tau$. The ansatz for the equilibrium distribution is the low Mach number expansion

$$f_i^{\mathrm{eq}}(\rho, \mathbf{u}) = w_i \rho \left(1 + A\mathbf{u}\cdot\mathbf{c}_i + B(\mathbf{u}\cdot\mathbf{c}_i)^2 + C\mathbf{u}^2\right). \tag{2.48}$$

We have to retain at least terms up to $\mathcal{O}(\mathbf{u}^2)$ in order to obtain the quadratic term in the Euler stress. Sometimes, the non-linear terms are neglected which corresponds to the creeping-flow limit [63, 64]. However, the $\mathbf{uu}$-terms ensure Galilean invariance of the Euler stress [65, 66] and will not be neglected here. For symmetry reasons, the w_i must be independent of the direction of the $\mathbf{c}_i$ and must only depend on the length $|\mathbf{c}_i|$. The other coefficients A, B and C are independent of $\mathbf{c}_i$. Since mass conservation is to be satisfied independently of the value of $\mathbf{u}$, we can use $\mathbf{u} = 0$ in (2.48) to obtain the normalization condition for the weights

$$\sum w_i = 1. \tag{2.49}$$

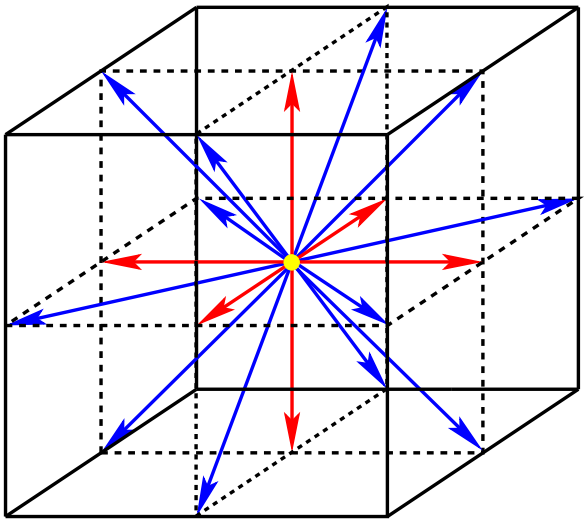


Figure 2.1: Illustration of the D3Q19 model. There are 19 populations on every lattice site: six that move to the nearest neighbors during streaming (red), twelve that move to to the next nearest neighbors (blue), and one population that stays on the same lattice site (yellow).

The remaining conditions on the moments on f_i^{eq} yield

$$\begin{aligned}
\sum f_i^{\mathrm{eq}} &= \rho\left(1 + AT_{\alpha}^{(1)}u_\alpha + BT_{\alpha\beta}^{(2)}u_\alpha u_\beta + Cu_\alpha u_\alpha\right), \\
\sum f_i^{\mathrm{eq}} c_{i\alpha} &= \rho\left(T_{\alpha}^{(1)} + AT_{\alpha\beta}^{(2)}u_\beta + BT_{\alpha\beta\gamma}^{(3)}u_\beta u_\gamma + T_{\alpha}^{(1)}u_\beta u_\beta\right), \\
\sum f_i^{\mathrm{eq}} c_{i\alpha}c_{i\beta} &= \rho\left(T_{\alpha\beta}^{(2)} + AT_{\alpha\beta\gamma}^{(3)}u_\gamma + BT_{\alpha\beta\gamma\delta}^{(4)}u_\gamma u_\delta + T_{\alpha\beta}^{(2)}u_\gamma u_\gamma\right),
\end{aligned} \tag{2.50}$$

where the lattice sums

$$T_{\alpha_1\ldots\alpha_n}^{(n)} = \sum_i w_i c_{i\alpha_1}\ldots c_{i\alpha_n}. \tag{2.51}$$

are involved. The lattice sums are treated in appendix B.2. Using (B.22), (B.24) and (B.26) in (2.50) we get

$$\begin{aligned}
\sum f_i^{\mathrm{eq}} &= \rho + B\sigma_2\rho u_\alpha u_\alpha + C\rho u_\alpha u_\alpha, \\
\sum f_i^{\mathrm{eq}} c_{i\alpha} &= A\sigma_2\rho u_\alpha, \\
\sum f_i^{\mathrm{eq}} c_{i\alpha}c_{i\beta} &= \rho\sigma_2\delta_{\alpha\beta} + B\kappa_4\delta_{\alpha\beta\gamma\delta}\rho u_\gamma u_\delta \\
&\quad + B\sigma_4\left(\delta_{\alpha\beta}\delta_{\gamma\delta} + \delta_{\alpha\gamma}\delta_{\beta\delta} + \delta_{\alpha\delta}\delta_{\beta\gamma}\right)\rho u_\gamma u_\delta + C\sigma_2\delta_{\alpha\beta}\rho u_\gamma u_\gamma.
\end{aligned} \tag{2.52}$$

By comparison with (2.42) to (2.44), we can find a unique solution for the coefficients of the equilibrium distribution and the lattice tensors:

$$\begin{aligned}
A &= \frac{1}{c_s^2}, & B &= \frac{1}{2c_s^4}, & C &= -\frac{1}{2c_s^2}, \\
\sigma_2 &= c_s^2, & \sigma_4 &= \sigma_2^2, & \kappa_4 &= 0.
\end{aligned} \tag{2.53}$$

The weights are then given by

$$\begin{aligned} w_i &= \frac{1}{3} \qquad \text{for } \hat{\mathbf{c}}_i^2 = 0 \quad \text{(zero velocity)}, \\ w_i &= \frac{1}{18} \qquad \text{for } \hat{\mathbf{c}}_i^2 = 1 \quad \text{(nearest neighbors)}, \\ w_i &= \frac{1}{36} \qquad \text{for } \hat{\mathbf{c}}_i^2 = 2 \quad \text{(next-nearest neighbors)}, \end{aligned} \tag{2.54}$$

and the speed of sound has the fixed value

$$c_s^2 = \frac{1}{3}\left(\frac{a}{\tau}\right)^2. \tag{2.55}$$

This solution corresponds to the case where the mass density is equally distributed among the different subshells. It was noted earlier that this is beneficial for the stability of the D3Q19 model, and the maximum entropy formalism provides a deeper explanation for this improved stability. This was already recognized by Karlin et al. [30], albeit the formalism was slightly different there. We can now write down the equilibrium distribution for the D3Q19 model:

$$f_i^{\text{eq}}(\rho, \mathbf{u}) = w_i \rho \left(1 + \frac{\mathbf{u}\cdot\mathbf{c}_i}{c_s^2} + \frac{(\mathbf{u}\cdot\mathbf{c}_i)^2}{2c_s^4} - \frac{\mathbf{u}^2}{2c_s^2}\right), \tag{2.56}$$

which is the same form as obtained by the truncated Hermite expansion in equation (2.34).

2.5.2 Collision operator

Having constructed the equilibrium distribution, we now turn to the linear collision operator. It has to satisfy mass and momentum conservation and must be compatible with the symmetries of the D3Q19 lattice. Furthermore, it has to yield the Navier-Stokes form of the deviatoric momentum flux, which indicates that the form of the collision operator is related to the viscosity of the fluid. The precise link will be established within the Chapman-Enskog expansion. In this subsection, we develop a representation for the general linear collision operator.

$$\Delta_i(\mathbf{f}(\mathbf{r}, t)) = \mathcal{L}_{ij}\left(f_j - f_j^{\text{eq}}\right) = \mathcal{L}_{ij} f_j^{\text{neq}}. \tag{2.57}$$

The simple BGK collision operator corresponds to the choice $\mathcal{L}_{ij} = -\lambda\delta_{ij}$, which has some drawbacks such as fixed ratio of bulk and shear viscosity. A more general collision operator is provided by the multi-relaxation time model (MRT) of d'Humières et al. [25], which uses a diagonal representation of $\mathcal{L}_{ij}$ in the so-called *mode space*.

The basis vectors $\mathbf{e}_k$ of mode space are constructed by orthogonalizing polynomials of the dimensionless velocity vectors $\hat{\mathbf{c}}_i$. The corresponding orthogonality relation is

$$\sum_i w_i e_{ki} e_{li} = b_k \delta_{kl}, \tag{2.58}$$

where the weights from the equilibrium distribution enter. This is not necessary in general, but it has the advantage that the kinetic moments have no projection on the equilibrium distribution. The backward relation is then

$$\sum_k b_k^{-1} e_{ki} e_{kj} = w_i^{-1} \delta_{ij}. \tag{2.59}$$

The b_k are normalization factors with respect to the scalar product

$$b_k = \sum_i w_i e_{ki}^2. \tag{2.60}$$

With this, we can apply Gram-Schmidt orthogonalization to polynomials of the $\hat{\mathbf{c}}_i$ in a carefully chosen order. The first basis vectors are given by

$$\begin{aligned} e_{0i} &= 1, \\ e_{1i} &= \hat{c}_{ix}, \\ e_{2i} &= \hat{c}_{iy}, \\ e_{3i} &= \hat{c}_{iz}. \end{aligned} \tag{2.61}$$

They correspond to the mass and the momentum, respectively. The next six basis vectors are obtained from quadratic polynomials

$$\begin{aligned} e_{4i} &= \hat{\mathbf{c}}_i^2 - 1, \\ e_{5i} &= 3\hat{c}_{ix}^2 - \hat{\mathbf{c}}_i^2, \\ e_{6i} &= \hat{c}_{iy}^2 - \hat{c}_{iz}^2, \\ e_{7i} &= \hat{c}_{ix}\hat{c}_{iy}, \\ e_{8i} &= \hat{c}_{ix}\hat{c}_{iz}, \\ e_{9i} &= \hat{c}_{iy}\hat{c}_{iz}. \end{aligned} \tag{2.62}$$

They correspond to bulk ($k = 4$) and shear ($k = 5 \ldots 9$) modes. Up to here, the polynomials are complete. The higher order polynomials are not complete due to degeneracies in the D3Q19 model, i.e., $\hat{c}_{i\alpha} = \hat{c}_{i\alpha}^3$. Nevertheless, by sorting out the degeneracies we can construct a complete basis. Orthogonalization of the non-degenerate higher order polynomials yields the remaining basis vectors

$$\begin{aligned} e_{10i} &= (3\hat{\mathbf{c}}_i^2 - 5)\hat{c}_{ix}, \\ e_{11i} &= (3\hat{\mathbf{c}}_i^2 - 5)\hat{c}_{iy}, \\ e_{12i} &= (3\hat{\mathbf{c}}_i^2 - 5)\hat{c}_{iz}, \\ e_{13i} &= (\hat{c}_{iy}^2 - \hat{c}_{iz}^2)\hat{c}_{ix}, \\ e_{14i} &= (\hat{c}_{iz}^2 - \hat{c}_{ix}^2)\hat{c}_{iy}, \\ e_{15i} &= (\hat{c}_{ix}^2 - \hat{c}_{iy}^2)\hat{c}_{iz}, \\ e_{16i} &= 3\hat{\mathbf{c}}_i^4 - 6\hat{\mathbf{c}}_i^2 + 1, \\ e_{17i} &= (2\hat{\mathbf{c}}_i^2 - 3)(3\hat{c}_{ix}^2 - \hat{\mathbf{c}}_i^2), \\ e_{18i} &= (2\hat{\mathbf{c}}_i^2 - 3)(\hat{c}_{iy}^2 - \hat{c}_{iz}^2). \end{aligned} \tag{2.63}$$

The normalization factors of these basis vectors are

$$\mathbf{b} = \left(1, \frac{1}{3}, \frac{1}{3}, \frac{1}{3}, \frac{2}{3}, \frac{4}{3}, \frac{4}{9}, \frac{1}{9}, \frac{1}{9}, \frac{1}{9}, \frac{2}{3}, \frac{2}{3}, \frac{2}{3}, \frac{2}{9}, \frac{2}{9}, \frac{2}{9}, 2, \frac{4}{9}, \frac{4}{3}\right)^T. \tag{2.64}$$

The basis vectors are used to calculate a complete set of moments, the so-called *modes*

$$m_k = \sum_i e_{ki} f_i. \tag{2.65}$$

The back transformation from mode space to the populations can be obtained using (2.59) and is given by

$$f_i = w_i \sum_k b_k^{-1} m_k e_{ki}. \tag{2.66}$$

By construction, the moments originating from polynomials up to quadratic order yield the hydrodynamic variables

$$\begin{aligned}
\rho &= m_0, \\
j_x &= m_1\, a/\tau, \\
j_y &= m_2\, a/\tau, \\
j_z &= m_3\, a/\tau, \\
\Pi_{\alpha\alpha} &= (m_0 + m_4)\,(a/\tau)^2, \\
\overline{\Pi}_{xx} &= \frac{1}{3} m_5 (a/\tau)^2, \\
\overline{\Pi}_{yy} &= -\frac{m_5 - 3m_6}{6} (a/\tau)^2, \\
\overline{\Pi}_{zz} &= -\frac{m_5 + 3m_6}{6} (a/\tau)^2, \\
\overline{\Pi}_{xy} &= m_7 (a/\tau)^2, \\
\overline{\Pi}_{xz} &= m_8 (a/\tau)^2, \\
\overline{\Pi}_{yz} &= m_9 (a/\tau)^2,
\end{aligned} \tag{2.67}$$

where we have decomposed the pressure tensor into its trace and the traceless part

$$\Pi_{\alpha\beta} = \overline{\Pi}_{\alpha\beta} + \frac{1}{3} \Pi_{\gamma\gamma} \delta_{\alpha\beta}. \tag{2.68}$$

The first moments m_0 to m_3 are the conserved hydrodynamic modes, i.e., mass density and momentum density. The equilibria of the non-conserved hydrodynamic modes can be

expressed as functions of the mass density and the momentum density:

$$\begin{aligned} m_4^{\text{eq}} &= \frac{j_x^2 + j_y^2 + j_z^2}{\rho}, \\ m_5^{\text{eq}} &= \frac{j_x^2 - j_y^2}{\rho}, \\ m_6^{\text{eq}} &= \frac{2j_x^2 - j_y^2 - j_z^2}{\rho}, \\ m_7^{\text{eq}} &= \frac{j_x j_y}{\rho}, \\ m_8^{\text{eq}} &= \frac{j_x j_z}{\rho}, \\ m_9^{\text{eq}} &= \frac{j_y j_z}{\rho}. \end{aligned} \tag{2.69}$$

The higher moments m_{10} to m_{18} are related to the additional degrees of freedom due to the kinetic representation. They will therefore be referred to as *kinetic modes*.[9] By construction, they have no equilibrium part. Although the kinetic modes do not influence the dynamics in the hydrodynamic limit, they can have an effect through the underlying kinetic model. This becomes important when adding fluctuations to the model and in the case of boundary conditions.

It should be remarked that in principle one can choose the equilibrium values of all the non-conserved moments at will, as long as they are compatible with the symmetry of the lattice. The chosen values then enter the MRT algorithm (cf. equation (2.71) below) instead of those obtained from the explicit expression for the equilibrium distribution. However, it was shown in Lallemand and Luo [26] that the additional parameters, if chosen to satisfy Galilean invariance and isotropy, reduce to the ones that come out of the equilibrium distribution. We will therefore use the expressions (2.69) without any additional parameters.

The transformation to mode space can be written in matrix-vector form

$$\mathbf{m} = \mathsf{M}\,\mathbf{f} \tag{2.70}$$

where the entries of the transformation matrix M are the components of the basis vectors e_{ij}. For the D3Q19 model, the matrix is given in figure 2.2. Since M represents a basis transformation, the collision operator can be represented in mode space as

$$\mathcal{L}\,\mathbf{f}^{\text{neq}} = \mathsf{M}^{-1}\left(\mathsf{M}\mathcal{L}\mathsf{M}^{-1}\right)\mathsf{M}\,\mathbf{f}^{\text{neq}} = \mathsf{M}^{-1}\mathcal{L}_M\,\mathbf{m}^{\text{neq}} \tag{2.71}$$

where $\mathcal{L}_M = \mathsf{M}\,\mathcal{L}\,\mathsf{M}^{-1}$. In mode space, we can choose the collision operator to be diagonal such that the collisions describe a linear relaxation of the non-equilibrium moments

$$m_k^{*\text{neq}} = (1 + \lambda_k)\, m_k^{\text{neq}}. \tag{2.72}$$

[9] The kinetic modes are sometimes also called *ghost modes*, because they have no relevance on the hydrodynamic level.

$$
\mathsf{M} = \begin{pmatrix}
1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\
0 & 1 & -1 & 0 & 0 & 0 & 1 & -1 & 1 & -1 & 1 & -1 & 1 & -1 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 1 & -1 & 0 & 1 & -1 & -1 & 1 & 0 & 0 & 0 & 0 & 1 & -1 & 1 & -1 & 1 \\
0 & 0 & 0 & 0 & 0 & 1 & -1 & 0 & 0 & 0 & 1 & -1 & -1 & 1 & 1 & -1 & -1 & 1 & -1 \\
-1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\
0 & 1 & 1 & -1 & -1 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & -1 & -1 & -1 & -1 & -1 \\
0 & 1 & 1 & 1 & 1 & -2 & -2 & 2 & 2 & 2 & 2 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & -1 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & -1 & -1 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & -1 & -1 \\
0 & -2 & 2 & 0 & 0 & 0 & 0 & 1 & -1 & 1 & -1 & 1 & -1 & 1 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & -2 & 2 & 0 & 0 & 1 & -1 & -1 & 1 & 0 & 0 & 0 & 1 & -1 & 1 & -1 & 1 \\
0 & 0 & 0 & 0 & 0 & -2 & 2 & 0 & 0 & 0 & 0 & 1 & -1 & 1 & 1 & -1 & -1 & 1 & -1 \\
0 & -0 & 0 & 0 & 0 & 0 & 0 & 1 & -1 & 1 & -1 & 1 & -1 & 1 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & -0 & 0 & 0 & 0 & 1 & -1 & -1 & 1 & 0 & 0 & 0 & -1 & 1 & -1 & 1 & 1 \\
0 & 0 & 0 & 0 & 0 & -0 & 0 & 0 & 0 & 0 & 0 & 1 & -1 & 1 & -1 & 1 & 1 & -1 & -1 \\
1 & -2 & -2 & -2 & -2 & -2 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\
0 & -1 & -1 & 1 & 1 & -0 & -0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 1 & -1 & -1 & -1 & -1 \\
0 & -1 & -1 & -1 & -1 & 2 & 2 & 2 & 2 & 2 & 2 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1
\end{pmatrix}.
$$

Figure 2.2: The transformation matrix M for the D3Q19 model for the basis vectors given in equations (2.61), (2.62) and (2.63). It is to be noted that this representation depends on the order of the $\mathbf{c}_i$.

The choice of the eigenvalues λ_k must reflect the symmetries of the lattice, that is, the modes that are related by a symmetry must have the same eigenvalue. For the conserved modes m_0 to m_3, the eigenvalue is irrelevant because $m_k^{\text{neq}} = 0$. The remaining modes can be grouped into six symmetry classes using the explicit expression for the basis vectors; the bulk mode m_4, the shear modes m_5 to m_9, and four groups of kinetic modes: the triplet m_{10} to m_{12}, the triplet m_{13} to m_{15}, the singlet m_{16} and the doublet m_{17} and m_{18}. From (2.67) we get that the eigenvalues λ_b and λ_s for the bulk and shear modes correspond to the relaxation of the trace and the traceless part of the pressure tensor

$$\begin{aligned} \Pi_{\alpha\alpha}^{*\text{neq}} &= (1+\lambda_b)\Pi_{\alpha\alpha}^{\text{neq}}, \\ \overline{\Pi}_{\alpha\beta}^{*\text{neq}} &= (1+\lambda_s)\overline{\Pi}_{\alpha\beta}^{\text{neq}}. \end{aligned} \tag{2.73}$$

Within the Chapman-Enskog expansion it will be shown that this relaxation process leads to the correct Newtonian viscous stress, where the viscosities are given in terms of the eigenvalues

$$\begin{aligned} \eta_s &= -\frac{\rho c_s^2 \tau}{2}\frac{2+\lambda_s}{\lambda_s}, \\ \eta_b &= -\frac{\rho c_s^2 \tau}{3}\frac{2+\lambda_b}{\lambda_b}. \end{aligned} \tag{2.74}$$

The eigenvalues for the kinetic modes are not related to any macroscopic transport coefficients and their value is irrelevant on the Navier-Stokes level. They do, however, influence the dynamics on the kinetic level. This is important in the case of boundary conditions where the kinetic eigenvalues can be tuned to improve the accuracy of the boundary condition. Moreover, the kinetic eigenvalues are related to the noise strength in the fluctuating lattice Boltzmann model, which will be discussed in section 3.3.4.

3 Statistical mechanics of the lattice Boltzmann equation

The lattice Boltzmann model presented in the previous section is completely deterministic and does not include fluctuations. This is a result of the coarse-graining procedure, where the averaging process for the single-particle distribution leads to a mean population number while the information about the variance is dropped. The connection to statistical mechanics of the microscopic degrees of freedom is thus lost. However, fluctuations can be essential at the mesoscopic level. Brownian motion, for example, is an immediate consequence of the fluctuations of the solvent molecules beyond the hydrodynamic scale. Besides Brownian motion, fluctuations are also relevant for non-linear phenomena and they have great impact on many critical phenomena. While other simulation methods for hydrodynamics, e.g., dissipative particle dynamics (DPD) [67–71] and multi-particle collision dynamics (MPCD) [7, 72–80], include those fluctuations automatically via the underlying particle representation of the solvent, the lattice Boltzmann method must be extended by suitable inclusion of thermal fluctuations. On the continuum level, this extension corresponds to describing the mesoscale dynamics of a fluid by fluctuating hydrodynamics [81].

In this section, we will introduce fluctuating hydrodynamics and the fluctuating lattice Boltzmann model. The original version of the fluctuating lattice Boltzmann equation was developed by Ladd [36–38] and connects the fluctuations of the populations to the fluctuating stress by solving a discrete Langevin equation. Although the derivation gives rise to the correct fluctuating hydrodynamics, it does not provide a direct link to the statistical mechanics of the model. Adhikari et al. [1] observed in practical simulations that in fact the variance of the fluctuating hydrodynamic quantities is not fully captured within Ladd's approach. They demonstrated that this can be improved by adding fluctuations to the non-hydrodynamic moments as well, but without a detailed theoretical clarification of the statistical mechanics. In the course of this work, we succeeded in finding a new derivation of the fluctuating lattice Boltzmann equation that provides a consistent link to statistical mechanics [2]. The presentation here will be based on the latter derivation, which also gives a clear quantitative meaning to the fluctuations of the populations.

3.1 Fluctuating hydrodynamics

We first present the equations of fluctuating hydrodynamics as put forward in [81]. The basic idea is that the structure of the hydrodynamic equations (2.1) remains unchanged in the presence of fluctuations because they are conservation equations for mass and momentum

that hold in general. Fluctuations in the fluid can only lead to local momentum fluxes[1] that vanish globally. Hence they must enter the equations as a divergence, which can be achieved by adding a fluctuating part s to the stress tensor σ

$$\sigma \rightarrow \sigma + \mathsf{s}. \tag{3.1}$$

The properties of the fluctuating stress s have to be consistent with the thermodynamics of the fluid. They can be derived within the theory of Gauss-Markov processes [82] or by using methods from Langevin analysis [83]. The fluctuating stress must have zero mean

$$\langle s_{\alpha\beta} \rangle = 0, \tag{3.2}$$

and must be delta-correlated in space and time

$$\langle s_{\alpha\beta}(\mathbf{r}_1, t_1)\, s_{\gamma\delta}(\mathbf{r}_2, t_2) \rangle = 2Q_{\alpha\beta\gamma\delta}\, \delta(\mathbf{r}_1 - \mathbf{r}_2)\delta(t_1 - t_2). \tag{3.3}$$

Another way of looking at the fluctuations is to see them as the balancing forces as opposed to the viscous dissipation in the fluid. From this point of view, the properties of the noise follow from the appropriate fluctuation dissipation relation. The non-dissipative part of hydrodynamics, i.e., an Euler fluid is described by the Hamiltonian

$$\mathcal{H}_{\text{EF}} = \int d\mathbf{r} \left(\frac{\rho}{2} \mathbf{u}^2 + \epsilon(\rho) \right), \tag{3.4}$$

where $\epsilon(\rho)$ is the internal energy density of the fluid, which is related to the pressure p by

$$\rho \frac{\partial^2 \epsilon}{\partial \rho^2} = \frac{\partial p}{\partial \rho}. \tag{3.5}$$

The latter equation can be used to construct $\epsilon(\rho)$. The Liouville operator for the Hamiltonian system is

$$\mathcal{L}_{\text{EF}} = \int d\mathbf{r} \left[\partial_\alpha j_\alpha \frac{\delta}{\delta \rho} + \partial_\beta \Pi^E_{\alpha\beta} \frac{\delta}{\delta j_\alpha} \right], \tag{3.6}$$

where $\frac{\delta}{\delta \rho}$ and $\frac{\delta}{\delta j_\alpha}$ denote functional derivatives with respect to mass density and momentum density, and $\Pi^E_{\alpha\beta} = p\delta_{\alpha\beta} + \rho u_\alpha u_\beta$ is the Euler stress. $\mathcal{L}_{\text{EF}}$ can be used to write the Fokker-Planck equation for the Euler fluid

$$\partial_t P(\{\rho\}, \{\mathbf{j}\}) = \mathcal{L}_{\text{EF}} P(\{\rho\}, \{\mathbf{j}\}). \tag{3.7}$$

This formulation can now easily be extended to include dissipative (viscous) and fluctuating (stochastic) parts by adding the appropriate terms to the Liouville operator for the Euler fluid, i.e., we replace

$$\mathcal{L}_{\text{EF}} \rightarrow \mathcal{L}_{\text{EF}} + \mathcal{L}_v + \mathcal{L}_s. \tag{3.8}$$

[1] We restrict the discussion to an isothermal fluid and the momentum equation. The heat transport equation can in principle be treated in the same way by introducing local fluctuating heat fluxes.

The viscous and fluctuating operators can be obtained from the Navier-Stokes equation by a Kramers-Moyal expansion [see Ref. 84] and are given by

$$\begin{aligned}\mathcal{L}_v &= -\eta_{\alpha\beta\gamma\delta}\int d\mathbf{r}\,\partial_\beta\partial_\gamma u_\delta\frac{\delta}{\delta j_\alpha},\\ \mathcal{L}_s &= -Q_{\alpha\beta\gamma\delta}\int d\mathbf{r}\,\partial_\beta\partial_\gamma\frac{\delta}{\delta j_\alpha}\frac{\delta}{\delta j_\delta}.\end{aligned} \tag{3.9}$$

The fluctuation dissipation relation requires that the Boltzmann factor $\exp(-\mathcal{H}_{\text{EF}}/k_BT)$ is a stationary solution of the Fokker-Planck equation. Therefore we have to require

$$\begin{aligned}0 &= (\mathcal{L}_v+\mathcal{L}_s)\exp\left(-\frac{\mathcal{H}_{\text{EF}}}{k_BT}\right)\\ &= -\left[\eta_{\alpha\beta\gamma\delta}\int d\mathbf{r}\,\partial_\beta\partial_\gamma u_\delta\frac{\delta}{\delta j_\alpha}+Q_{\alpha\beta\gamma\delta}\int d\mathbf{r}\,\partial_\beta\partial_\gamma\frac{\delta}{\delta j_\alpha}\frac{\delta}{\delta j_\delta}\right]\exp\left(-\frac{\mathcal{H}_{\text{EF}}}{k_BT}\right).\end{aligned} \tag{3.10}$$

This condition is satisfied if

$$\begin{aligned}0 &= \left[\eta_{\alpha\beta\gamma\delta}\partial_\beta\partial_\gamma u_\delta+Q_{\alpha\beta\gamma\delta}\partial_\beta\partial_\gamma\frac{\delta}{\delta j_\delta}\right]\exp\left(-\frac{\mathcal{H}_{\text{EF}}}{k_BT}\right)\\ &= \left[\eta_{\alpha\beta\gamma\delta}\partial_\beta\partial_\gamma u_\delta-\frac{1}{k_BT}Q_{\alpha\beta\gamma\delta}\partial_\beta\partial_\gamma u_\delta\right]\exp\left(-\frac{\mathcal{H}_{\text{EF}}}{k_BT}\right),\end{aligned} \tag{3.11}$$

which yields

$$Q_{\alpha\beta\gamma\delta} = k_BT\,\eta_{\alpha\beta\gamma\delta}. \tag{3.12}$$

Using this result in (3.3) we get the correlations for the traceless and trace parts of the fluctuating stress tensor

$$\begin{aligned}\langle\overline{s_{\alpha\beta}}(\mathbf{r}_1,t_1)\,\overline{s_{\gamma\delta}}(\mathbf{r}_2,t_2)\rangle &= 2\eta_s k_BT\left(\delta_{\alpha\gamma}\delta_{\beta\delta}+\delta_{\alpha\delta}\delta_{\beta\gamma}-\frac{2}{3}\delta_{\alpha\beta}\delta_{\gamma\delta}\right)\delta(\mathbf{r}_1-\mathbf{r}_2)\delta(t_1-t_2),\\ \langle s_{\alpha\alpha}(\mathbf{r}_1,t_1)\,s_{\beta\beta}(\mathbf{r}_2,t_2)\rangle &= 18\eta_b k_BT\,\delta(\mathbf{r}_1-\mathbf{r}_2)\delta(t_1-t_2),\\ \langle\overline{s_{\alpha\beta}}(\mathbf{r}_1,t_1)\,s_{\gamma\gamma}(\mathbf{r}_2,t_2)\rangle &= 0.\end{aligned} \tag{3.13}$$

These are the same expressions as originally put forward by Landau and Lifshitz [81]. They introduce fluctuations into the hydrodynamic equations in a thermodynamically consistent way. The derivation presented here points out that the stochastic momentum flux can be seen as the counterbalance to the viscous friction. The fluctuation dissipation relation between the dissipative and the fluctuating part assures that the correct isothermal ensemble is obtained. We will see later that such a balance is important for every degree of freedom in the system, even for those that are irrelevant on the hydrodynamic level.

3.2 The fluctuating lattice Boltzmann equation

Fluctuations can be incorporated into the lattice Boltzmann equation by adding a stochastic contribution to the collision term

$$\Delta_i = \mathcal{L}_{ij}f_j^{\text{neq}} + \Delta_i'. \tag{3.14}$$

Mass and momentum conservation imply that the zeroth and first moment of the stochastic contribution must vanish

$$\sum_i \Delta_i' = 0, \qquad \sum_i \Delta_i' \mathbf{c}_i = 0. \tag{3.15}$$

It is assumed that the fluctuations Δ_i' are uncorrelated in space and time such that the stochastic collision operator is still local. The mean of the fluctuations vanishes $\langle \Delta_i' \rangle = 0$, and the covariance matrix $\langle \Delta_i' \Delta_j' \rangle$ has to generate the correct fluctuations on the hydrodynamic level, cf. equation (3.13). The stochastic collision operator gives rise to a random contribution to the non-equilibrium stress

$$\sigma_{\alpha\beta}^r = \sum_i \Delta_i' c_{i\alpha} c_{i\beta}. \tag{3.16}$$

The modification of equations (2.73) for the stochastic collision operator reads

$$\begin{aligned} \Pi_{\alpha\alpha}^{*\mathrm{neq}} &= (1+\lambda_b)\Pi_{\alpha\alpha}^{\mathrm{neq}} + \sigma_{\alpha\alpha}^r, \\ \overline{\Pi}_{\alpha\beta}^{*\mathrm{neq}} &= (1+\lambda_s)\overline{\Pi}_{\alpha\beta}^{\mathrm{neq}} + \overline{\sigma}_{\alpha\beta}^r. \end{aligned} \tag{3.17}$$

It is to be noted that $\sigma_{\alpha\beta}^r$ corresponds to fluctuations on the lattice time scale τ, as opposed to the fluctuations $s_{\alpha\beta}$ of the hydrodynamic momentum flux on the kinetic time scale. The connection between the two scales will become explicit within the Chapman-Enskog expansion. The relation between the hydrodynamic fluctuations and the random stresses on the lattice level is

$$\begin{aligned} \sigma_{\alpha\alpha}^r &= \lambda_b s_{\alpha\alpha}, \\ \overline{\sigma}_{\alpha\beta}^r &= \lambda_s \overline{s_{\alpha\beta}}, \end{aligned} \tag{3.18}$$

which leads to the following correlations for the random stresses

$$\begin{aligned} \left\langle \overline{\sigma}_{\alpha\beta}^r \overline{\sigma}_{\gamma\delta}^r \right\rangle &= \frac{2\eta_s k_B T \lambda_s^2}{a^3 \tau} \left(\delta_{\alpha\gamma}\delta_{\beta\delta} + \delta_{\alpha\delta}\delta_{\beta\gamma} - \frac{2}{3}\delta_{\alpha\beta}\delta_{\gamma\delta} \right), \\ \left\langle \sigma_{\alpha\alpha}^r \sigma_{\beta\beta}^r \right\rangle &= \frac{18\eta_b k_B T \lambda_b^2}{a^3 \tau}, \\ \left\langle \overline{\sigma}_{\alpha\beta}^r \sigma_{\gamma\gamma}^r \right\rangle &= 0. \end{aligned} \tag{3.19}$$

The delta distributions of equation (3.13) have been replaced here by the lattice units a^{-3} and τ^{-1} to reflect the locality of the stochastic collisions on the discrete lattice. These are the expressions for the random stresses as originally derived by Ladd [36, 37], which guarantee correct fluctuating hydrodynamics at the macroscopic level. They can be directly implemented in the MRT lattice Boltzmann model by adding the random contributions in mode space during the collision phase. In the original implementation of Ladd and various follow-ups by other authors [38, 85–88], the random contribution was only imposed on the stress modes, while the kinetic modes where not thermalized but projected out entirely during the collision phase. Although this is perfectly consistent with fluctuating hydrodynamics, it was

shown by Adhikari et al. [1] that the procedure leads to poor accuracy on shorter length scales. The reason is that the procedure neglects the kinetic nature of the lattice Boltzmann method, which includes more degrees of freedom than just the hydrodynamic ones, namely the higher order kinetic modes. These are non-conserved modes that take part in the dissipative processes in the fluid, which suggests that they should be thermalized as well. It was shown numerically that additional noise on the kinetic modes significantly improves the accuracy on short length-scales [1]. In the course of this thesis, this observation could be clarified theoretically from a statistical mechanics viewpoint, making use of a generalized lattice gas model [2]. The generalized lattice gas model and the derivation of the statistical mechanics of the fluctuating lattice Boltzmann model will be the subject of the following section.

3.3 The generalized lattice gas model (GLG)

One of the motives that drove the development of the lattice Boltzmann method was the aim to cope with the large statistical noise inherent to the preceding lattice gas automaton models. Since the dynamic quantities in the lattice gas automata are boolean variables, a considerable amount of sampling is needed to obtain good data statistics for the hydrodynamic fields. To circumvent this inefficiency, the lattice Boltzmann method replaces the boolean variables with their ensemble-averaged populations. This yields smooth dynamic variables that are not subject to noise any longer. On the other hand, the complete absence of fluctuations means that the connection to the underlying statistical properties of the populations is lost, which makes it necessary to reintroduce the fluctuations a posteriori. In the previous section we have shown how this can be done for fluctuating hydrodynamics. However, that approach fails to restore the connection to the statistics of the underlying micro-model. For simulations of soft matter systems it is of pivotal importance to sample a well defined statistical ensemble. In order to put the lattice Boltzmann method back onto the fundament of statistical mechanics, we have developed the conceptual model of a *generalized lattice gas* (GLG).

In the GLG model, the equilibrium distribution of the population numbers can be derived from fundamental statistical considerations. For this purpose, we introduce an ensemble of population numbers on the local lattice site where each velocity direction $\mathbf{c}_i$ can be occupied by an integer number ν_i of particles. The evolution equation for a single realization of the occupancies has the same form as in the lattice gas and lattice Boltzmann models

$$\nu_i(\mathbf{r} + \mathbf{c}_i h, t + h) = \nu_i^*(\mathbf{r}, t) = \nu_i(\mathbf{r}, t) + \Delta_i \left(\boldsymbol{\nu}(\mathbf{r}, t)\right). \tag{3.20}$$

The collision operator Δ_i of this model redistributes particles among the different velocity directions.The difference to the LG and LB models lies in the nature of the ν_i, for in a lattice gas the occupation variables are boolean whereas in lattice Boltzmann real-valued variables are used. The ensemble picture behind the occupancies allows to quantify the difference by

looking at the fluctuations of the variables ν_i. The dimensionless *Boltzmann number* (Bo) is defined by

$$Bo = \frac{\sqrt{\langle \nu_i^2 \rangle - \langle \nu_i \rangle^2}}{\langle \nu_i \rangle}, \tag{3.21}$$

where the angle brackets denote ensemble averages. In the LG models, $Bo \sim 1$, i.e., the fluctuations are on the same order as the mean which corresponds to a fully microscopic model. Conversely, erasing any fluctuations as in the deterministic LB models leads to $Bo = 0$. Our aim is to introduce thermal fluctuations in such a way that fluctuating hydrodynamics is obtained and at the same time statistical consistency is retained at the microscopic level.

The connection between the integer variables ν_i and the hydrodynamic variables can be established by introducing the mass density

$$\mu = \frac{m_p}{a^3}, \tag{3.22}$$

where m_p is the mass of a particle and a is the lattice spacing. The relation between the GLG occupancies ν_i and the LB mass densities f_i is then

$$f_i = \mu \nu_i, \tag{3.23}$$

and the hydrodynamic fields can be obtained as usual in lattice Boltzmann.

3.3.1 Statistics of the generalized lattice gas

The statistics on the microscopic level is governed by the probability distribution of the occupation numbers. In a homogeneous equilibrium state, we can consider the individual occupancies of the different velocity directions as independently sampled from a large reservoir. The probability distribution for an individual ν_i is then Poissonian

$$p(\nu_i) = \frac{\bar{\nu}_i^{\nu_i}}{\nu_i!} e^{-\bar{\nu}_i}, \tag{3.24}$$

which is in accordance with the phase-space occupancies in an ideal gas [89]. $\bar{\nu}_i = \langle \nu_i \rangle$ denotes the mean number of particles occupying $\mathbf{c}_i$. This mean occupation number $\bar{\nu}_i$ can be written in terms of the total number of particles $\bar{\nu}$ on the lattice site

$$\bar{\nu}_i = w_i \sum_i \bar{\nu}_i = w_i \bar{\nu}, \tag{3.25}$$

where, for symmetry reasons, the weights w_i depend on the absolute speed of $\mathbf{c}_i$ only and not on the direction. The total occupation is related to the mass density by $\mu\bar{\nu} = \rho$. From Poisson statistics it follows that $\langle \nu_i^2 \rangle - \langle \nu_i \rangle^2 = \langle \nu_i \rangle$ and the variance of the LB mass density $\langle f_i^2 \rangle - \langle f_i \rangle^2 = \mu \langle f_i \rangle = m_p/a^3 \langle f_i \rangle$ is controlled by the mass m_p of a particle. The latter can be related to the temperature through the ideal gas equation of state

$$m_p = \frac{k_B T}{c_s^2}. \tag{3.26}$$

This means that $\langle f_i^2 \rangle - \langle f_i \rangle^2 \sim T$ and the thermal fluctuations can thus be controlled by the temperature as desired.

The joint probability distribution P of the occupation numbers is the product of the individual Poissonians, but subject to the constraints that it has to correspond to a given total mass and momentum

$$P(\{\nu_i\}) \propto \left(\prod_i \frac{\bar{\nu}_i^{\nu_i}}{\nu_i!} e^{-\bar{\nu}_i} \right) \delta \left(\mu \sum_i \nu_i - \rho \right) \delta \left(\mu \sum_i \nu_i \mathbf{c}_i - \mathbf{j} \right). \tag{3.27}$$

Using Stirling's approximation for the factorial in (3.27) we can introduce an associated entropy for the occupation numbers

$$S\left(\{\nu_i\}\right) = -\sum_i \left(\nu_i \ln \nu_i - \nu_i - \nu_i \ln \bar{\nu}_i + \bar{\nu}_i \right), \tag{3.28}$$

and the probability P can be rewritten as[2]

$$P\left(\{\nu_i\}\right) \propto \exp\left[S\left(\{\nu_i\}\right)\right] \delta \left(\mu \sum_i \nu_i - \rho \right) \delta \left(\mu \sum_i \nu_i \mathbf{c}_i - \mathbf{j} \right). \tag{3.29}$$

3.3.2 Equilibrium distribution

We now take as the equilibrium distribution ν_i^{eq} of the GLG the most probable set of occupation numbers for given values of mass and momentum. It can be obtained by maximizing P, or equivalently, by maximizing the entropy S subject to the constraints. We take the constraints into account via Lagrange multipliers and maximize the functional

$$\mathcal{S}(\{\nu_i\}, \chi, \boldsymbol{\lambda}) = S(\{\nu_i\}) + \chi \left(\sum_i \nu_i - \frac{\rho}{\mu} \right) + \boldsymbol{\lambda} \cdot \left(\sum_i \nu_i \mathbf{c}_i - \frac{\mathbf{j}}{\mu} \right). \tag{3.30}$$

Differentiation with respect to ν_i, χ and $\boldsymbol{\lambda}$ results in the following equation system

$$\frac{\partial S}{\partial \nu_i} + \chi + \boldsymbol{\lambda} \cdot \mathbf{c}_i = 0, \tag{3.31a}$$

$$\mu \sum_i \nu_i - \rho = 0, \tag{3.31b}$$

$$\mu \sum_i \nu_i \mathbf{c}_i - \mathbf{j} = 0. \tag{3.31c}$$

[2] The associated entropy and equation (3.29) can also be derived by considering a Bernoulli experiment where particles are selected with probability p_0 such that $\bar{\nu} = N p_0$ particles are drawn in total:

$$p(\nu) = \frac{N!}{\nu!(N-\nu)!} \left(\frac{\bar{\nu}}{N} \right)^{\nu} \left(1 - \frac{\bar{\nu}}{N} \right)^{N-\nu}.$$

The entropy (3.28) then follows from $\ln p(\nu)$ in the limit $N \to \infty$ at fixed $\bar{\nu}$ [see also Ref. 8].

The formal solution of (3.31a) is

$$\nu_i^{\mathrm{eq}} = \bar{\nu}_i \, \exp\left(\chi + \boldsymbol{\lambda} \cdot \mathbf{c}_i\right), \tag{3.32}$$

where the Lagrange multipliers χ and $\boldsymbol{\lambda}$ are functions of mass and momentum and have to be determined from the constraints (3.31b) and (3.31c). Due to the nonlinearity of the equation system, the explicit solution is difficult to obtain. However, in analogy to the low Mach number approximation in the conventional lattice Boltzmann models, we seek a polynomial approximation for ν_i^{eq}. We expand the equilibrium distribution around the reference state where the fluid is at rest. For this case, we can write the equilibrium distribution explicitly

$$\nu_i^{(0)} = \nu_i^{\mathrm{eq}}(\mathbf{j} = 0) = w_i \bar{\nu}. \tag{3.33}$$

The solution for non-vanishing momentum is obtained by perturbing around $\mathbf{j} = 0$, that is, we expand the Lagrange multipliers as

$$\chi = \sum_{n=1}^{\infty} \tilde{\epsilon}^n \chi^{(n)}, \qquad \boldsymbol{\lambda} = \sum_{n=1}^{\infty} \tilde{\epsilon}^n \boldsymbol{\lambda}^{(n)}, \qquad \mathbf{j} = \tilde{\epsilon} \mathbf{j}^{(1)} \tag{3.34}$$

where $\tilde{\epsilon}$ is a formal parameter that will be set to one at the end. Expanding the equilibrium distribution up to second order in the Lagrange multipliers yields

$$\begin{aligned} \nu_i^{\mathrm{eq}} = w_i \, \bar{\nu} \Big[& 1 + \tilde{\epsilon} \chi^{(1)} + \tilde{\epsilon} \boldsymbol{\lambda}^{(1)} \cdot \mathbf{c}_i \\ & + \frac{1}{2} \tilde{\epsilon}^2 \left(\chi^{(1)} + \boldsymbol{\lambda}^{(1)} \cdot \mathbf{c}_i \right)^2 + \tilde{\epsilon}^2 \chi^{(2)} + \tilde{\epsilon}^2 \boldsymbol{\lambda}^{(2)} \cdot \mathbf{c}_i + \dots \Big]. \end{aligned} \tag{3.35}$$

The constraints for mass and momentum should be satisfied by this expansion at all orders. The zeroth order does this by construction. The higher orders can be solved recursively. On the first order, we have

$$\begin{aligned} \mu \bar{\nu} \sum_i w_i \left(\chi^{(1)} + \boldsymbol{\lambda}^{(1)} \cdot \mathbf{c}_i \right) &= 0, \\ \mu \bar{\nu} \sum_i w_i \left[(1 + \chi^{(1)}) \mathbf{c}_i + \boldsymbol{\lambda}^{(1)} \cdot \mathbf{c}_i \mathbf{c}_i \right] &= \mathbf{j}^{(1)}. \end{aligned} \tag{3.36}$$

For the evaluation, we use the symmetry properties of the lattice sums of a cubic lattice derived in section B.2. The result for the Lagrange multipliers on the first order is

$$\chi^{(1)} = 0, \qquad \boldsymbol{\lambda}^{(1)} = \frac{\mathbf{j}^{(1)}}{\mu \bar{\nu} \sigma_2}. \tag{3.37}$$

Inserting this result into the next order we obtain the equations for the second order

$$\begin{aligned} \mu \bar{\nu} \sum_i w_i \left[\chi^{(2)} + \frac{1}{2} \left(\frac{\mathbf{j}^{(1)} \cdot \mathbf{c}_i}{\mu \bar{\nu} \sigma_2} \right)^2 \right] &= 0, \\ \mu \bar{\nu} \sum_i w_i \left(\boldsymbol{\lambda}^{(2)} \cdot \mathbf{c}_i \mathbf{c}_i \right) &= 0, \end{aligned} \tag{3.38}$$

and using the lattice sums again we get the second order Lagrange multipliers

$$\chi^{(2)} = -\frac{1}{2\sigma_2}\left(\frac{\mathbf{j}^{(1)}}{\mu\bar{\nu}}\right)^2, \qquad \boldsymbol{\lambda}^{(2)} = 0. \tag{3.39}$$

The procedure can be systematically carried out to higher orders. Results up to 8th order are for example given by Ansumali [90]. Here, we shall be satisfied with the second order approximation. Combining (3.37) and (3.39) into (3.35) we get the explicit solution for the equilibrium distribution up to quadratic terms in $\mathbf{j}$

$$\nu_i^{\mathrm{eq}} = w_i\bar{\nu}\left[1 + \frac{\mathbf{j}\cdot\mathbf{c}_i}{\mu\bar{\nu}\sigma_2} + \frac{1}{2\sigma_2^2}\left(\frac{\mathbf{j}\cdot\mathbf{c}_i}{\mu\bar{\nu}}\right)^2 - \frac{1}{2\sigma_2}\left(\frac{\mathbf{j}}{\mu\bar{\nu}}\right)^2\right]. \tag{3.40}$$

Converting the GLG occupation numbers into LB mass densities, $f_i^{\mathrm{eq}} = \mu\nu_i^{\mathrm{eq}}$, we have exactly the same form of the equilibrium distribution as used in the standard lattice Boltzmann models, cf. equations (2.34) and (2.56).

It should be remarked that the procedure described here is very similar to the entropic lattice Boltzmann approaches of Karlin and Succi [29], Karlin et al. [30], Ansumali et al. [91]. In that framework, however, the aim is to obtain an H-theorem for lattice Boltzmann models, and the derivation starts from a convex H-functional. The resulting equilibrium has, in contrast to the derivation presented here, no connection to an underlying probability distribution and consequently the entropic lattice Boltzmann is focused on deterministic models without fluctuations.

3.3.3 Fluctuations around equilibrium

The equilibrium distribution is the most probable set of populations of a lattice site for given mass and momentum. The actual populations ν_i fluctuate around those mean values according to the probability distribution P. Using the Fourier representation of the δ-distributions we can rewrite P as

$$P\left(\{\nu_i\}\right) \propto \int dq \int d\mathbf{k}\, \exp\left[S(\{\nu_i\}) + iq\left(\mu\sum_i \nu_i - \rho\right) + i\mathbf{k}\cdot\left(\mu\sum_i \nu_i\mathbf{c}_i - \mathbf{j}\right)\right]. \tag{3.41}$$

The expression in the square brackets is identical to the functional $\mathcal{S}(\{\nu_i\}, iq, i\mathbf{k})$ where the Lagrange multipliers χ and $\boldsymbol{\lambda}$ have been replaced by iq and $i\mathbf{k}$, respectively. The solution $(\{\nu_i^{\mathrm{eq}}\}, iq_0, i\mathbf{k}_0)$ obtained from the equation system (3.31a) above is a saddle point, around which we can expand to second order

$$\begin{aligned}\mathcal{S}\left(\{\nu_i\}, iq, i\mathbf{k}\right) = \mathcal{S}\left(\{\nu_i^{\mathrm{eq}}\}, iq_0, i\mathbf{k}_0\right) - \sum_i \frac{\left(\nu_i - \nu_i^{\mathrm{eq}}\right)^2}{2\nu_i^{\mathrm{eq}}} \\ + i\mu\left(q - q_0\right)\sum_i\left(\nu_i - \nu_i^{\mathrm{eq}}\right) + i\mu\left(\mathbf{k} - \mathbf{k}_0\right)\cdot\sum_i \mathbf{c}_i\left(\nu_i - \nu_i^{\mathrm{eq}}\right),\end{aligned} \tag{3.42}$$

where we have used the explicit form (3.28) of the entropy. This yields

$$\begin{aligned}\left.\frac{\partial^2 S}{\partial\nu_i\partial\nu_j}\right|_{\{\nu_i^{\mathrm{eq}}\}} &= -\frac{1}{\nu_i^{\mathrm{eq}}}\delta_{ij},\\ \frac{\partial^2 S}{\partial\nu_i\partial q} &= i\mu,\\ \frac{\partial^2 S}{\partial\nu_i\partial\mathbf{k}} &= i\mu\mathbf{c}_i.\end{aligned} \tag{3.43}$$

Inserting the expansion into (3.41) we can write the probability distribution for $\nu_i^{\mathrm{neq}} = \nu_i - \nu_i^{\mathrm{eq}}$

$$\begin{aligned}P\left(\{\nu_i^{\mathrm{neq}}\}\right) \propto \int d(q-q_0)\int d(\mathbf{k}-\mathbf{k}_0)\\ \times\exp\left[-\sum_i\frac{\left(\nu_i^{\mathrm{neq}}\right)^2}{2\nu_i^{\mathrm{eq}}} + i\mu\left(q-q_0\right)\sum_i\nu_i^{\mathrm{neq}} + i\mu\left(\mathbf{k}-\mathbf{k}_0\right)\sum_i\nu_i^{\mathrm{neq}}\mathbf{c}_i\right].\end{aligned} \tag{3.44}$$

where we have transformed the variables of the Fourier integrals and absorbed all constant factors in the normalization. Finally, we reintroduce the δ-distributions and obtain

$$P\left(\{\nu_i^{\mathrm{neq}}\}\right) \propto \exp\left[-\sum_i\frac{\left(\nu_i^{\mathrm{neq}}\right)^2}{2\nu_i^{\mathrm{eq}}}\right]\delta\left(\mu\sum_i\nu_i^{\mathrm{neq}}\right)\delta\left(\mu\sum_i\nu_i^{\mathrm{neq}}\mathbf{c}_i\right). \tag{3.45}$$

This expression shows that the fluctuations around the equilibrium have a Gaussian distribution subject to constraints. The variance is ν_i^{eq} and depends on direction, which is a consequence of the broken Galilean invariance. However, since the non-equilibrium populations are small compared to the equilibrium value, we can approximate the latter by the limit of vanishing fluid velocity $\mathbf{u} = 0$. In this case, Galilean invariance is restored and the variance

$$\lim_{\mathbf{u}\to 0}\nu_i^{\mathrm{eq}} = w_i\bar{\nu} \tag{3.46}$$

becomes independent of direction.[3] The final result for the fluctuations, written in terms of the lattice Boltzmann populations f_i, is

$$P\left(\{f_i^{\mathrm{neq}}\}\right) \propto \exp\left[-\sum_i\frac{\left(f_i^{\mathrm{neq}}\right)^2}{2\mu w_i\rho}\right]\delta\left(\sum_i f_i^{\mathrm{neq}}\right)\delta\left(\sum_i f_i^{\mathrm{neq}}\mathbf{c}_i\right), \tag{3.47}$$

which shows again that the fluctuations are controlled by μ.

In order to look at the fluctuations of the hydrodynamic variables, we transform to modes according to equation (2.65). The probability distribution for the non-equilibrium parts in

[3] Consequently, the weights w_i do not depend on direction. The approximation can be justified within the Chapman-Enskog expansion. It turns out that up to second order, the macroscopic dynamics is not changed by the approximation.

mode space is

$$P\left(\{m_k^{\text{neq}}\}\right) \propto \exp\left[-\sum_k \frac{\left(m_k^{\text{neq}}\right)^2}{2b_k\mu\rho}\right] \prod_{k\leq 3} \delta\left(m_k^{\text{neq}}\right) \propto \prod_{k>3} \exp\left[-\frac{\left(m_k^{\text{neq}}\right)^2}{2\mu b_k\rho}\right]. \tag{3.48}$$

Here, the constraints have been eliminated because the conserved moments m_k, $k = 0\ldots3$ do not fluctuate and hence do not contribute to P. The fluctuations in mode space are independent and Gaussian with variance $\mu b_k \rho$.

3.3.4 Stochastic collision operator and detailed balance

Having obtained the probability distribution for the statistical fluctuations of the non-equilibrium moments explicitly, we turn to the implementation in the lattice Boltzmann equation. Similar to section 3.2, we introduce a stochastic collision operator that adds noise to the deterministic dynamics.We extend the update rule for the moments by an additional random noise in the following way

$$m_k^{*\text{neq}} = \gamma_k m_k^{\text{neq}} + \varphi_k r_k, \tag{3.49}$$

where $\gamma_k = 1 + \lambda_k$ and r_k is a Gaussian random number with zero mean and unit variance. The amplitude φ_k of the random noise for the k-th mode remains to be determined. This can be achieved by interpreting the update rule (3.49) as a Monte-Carlo process: the fluctuations of the modes are sampled by random moves. Such a process has to satisfy detailed balance to generate the correct distribution. The condition of detailed balance is

$$\omega\left(m_k^{\text{neq}} \to m_k^{*\text{neq}}\right) \exp\left[-\frac{\left(m_k^{\text{neq}}\right)^2}{2\mu b_k\rho}\right] = \omega\left(m_k^{*\text{neq}} \to m_k^{\text{neq}}\right) \exp\left[-\frac{\left(m_k^{*\text{neq}}\right)^2}{2\mu b_k\rho}\right]. \tag{3.50}$$

The probability for a move from the pre-collision moment m_k^{neq} to the post-collision moment $m_k^{*\text{neq}}$ is equal to the probability of generating the Gaussian random noise $\varphi_k r_k = m_k^{*\text{neq}} - \gamma_k m_k^{\text{neq}}$

$$\omega\left(m_k^{\text{neq}} \to m_k^{*\text{neq}}\right) = \sqrt{\frac{1}{2\pi\varphi_k^2}} \exp\left[-\frac{\left(m_k^{*\text{neq}} - \gamma_k m_k^{\text{neq}}\right)^2}{2\varphi_k^2}\right]. \tag{3.51}$$

The probability of the reverse transition is obtained analogously. Combining (3.50) and (3.51) we get

$$\frac{\omega\left(m_k^{\text{neq}} \to m_k^{*\text{neq}}\right)}{\omega\left(m_k^{*\text{neq}} \to m_k^{\text{neq}}\right)} = \frac{\exp\left[-\left(m_k^{*\text{neq}}\right)^2/(2\mu b_k\rho)\right]}{\exp\left[-\left(m_k^{\text{neq}}\right)^2/(2\mu b_k\rho)\right]} = \frac{\exp\left[-\left(m_k^{*\text{neq}} - \gamma_k m_k^{\text{neq}}\right)^2/(2\varphi_k^2)\right]}{\exp\left[-\left(m_k^{\text{neq}} - \gamma_k m_k^{*\text{neq}}\right)^2/(2\varphi_k^2)\right]}. \tag{3.52}$$

Taking the logarithm yields

$$\frac{\left(m_k^{\text{neq}}\right)^2 - \left(m_k^{*\text{neq}}\right)^2}{2\mu b_k\rho} = \frac{\left(1-\gamma_k^2\right)\left[\left(m_k^{\text{neq}}\right)^2 - \left(m_k^{*\text{neq}}\right)^2\right]}{2\varphi_k^2}, \tag{3.53}$$

which is satisfied if and only if

$$\varphi_k^2 = \mu b_k \rho \left(1 - \gamma_k^2\right) = \frac{\rho k_B T}{c_s^2 a^3} b_k \left(1 - \gamma_k^2\right). \tag{3.54}$$

This relation holds for *all* modes in the system, where $\gamma_k = 1$ applies to conserved modes without fluctuations. The value $\gamma_k = 0$ projects out the deterministic part and makes the mode entirely random. It is important to note that the relation (3.54) ensures consistent sampling of the fluctuations on the microscopic level. Therefore, it guarantees that detailed balance is satisfied on all scales. This is in contrast to the previous fluctuating lattice Boltzmann, where only the stress modes are thermalized while the kinetic modes are entirely projected out, i.e., $\gamma_k = 0$ and $\varphi_k = 0$ at the same time. Our derivation shows that such a procedure violates detailed balance, because $\omega\left(m_k^{\text{neq}} \to 0\right) = 1$ and $\omega\left(0 \to m_k^{\text{neq}}\right) = 0$, and explains why it leads to the observed poor thermalization. The fluctuations of the kinetic modes are needed for detailed balance and proper thermalization beyond fluctuating hydrodynamics. The relation (3.54) has a general interpretation: every degree of freedom in the system that is subject to dissipation, i.e., $\gamma_k \neq 1$, needs random fluctuations to counterbalance dissipation. That is, the number of random variables needed to thermalize the system must be equal to the number of non-conserved degrees of freedom. This is necessary to ensure that for every trajectory the reverse trajectory can be generated as well, which is another formulation of detailed balance. Otherwise the system will not reach thermal equilibrium. Although this general argument seems rather trivial, the consequences for the lattice Boltzmann equation have long been overlooked. It is among the benefits of the development of the generalized lattice gas model that it makes a rigorous statistical mechanics derivation of these concepts possible in the framework of the fluctuating lattice Boltzmann equation.

It should also be noted that the expressions obtained in section 3.2 for the fluctuating stress remain valid. But they have now a bottom-up justification in terms of the statistical fluctuations on the microscopic level, whereas they were previously derived top-down by comparison with the macroscopic equations. For the connection between the microscopic level and the macroscopic hydrodynamics we once again refer to the Chapman-Enskog expansion described in chapter 4.

4 Asymptotic analysis and the Chapman-Enskog expansion

The lattice Boltzmann method is based on the idea that the mesoscale kinetic description of the system gives rise to hydrodynamic behavior on the macroscale. The formal connection between the different levels of description can be achieved within an asymptotic analysis of the lattice Boltzmann equation. In particular, such an analysis makes the relation between the parameters of the lattice Boltzmann model and the macroscopic transport coefficients explicit.

There exist different approaches to link the microscopic dynamics of a system to a reduced description in terms of macroscopic variables. Most of them are multiscale methods that are based on the separation of scales, i.e., different physical mechanisms can be distinguished according to the time and length scales they are governed by. The mechanisms do not interfere dynamically and can thus be treated separately. Formally, this separation can be treated by multiscale expansion techniques [10, 65, 92]. The commonly used method for asymptotic analysis of the lattice Boltzmann equation is the Chapman-Enskog expansion, which will be described in the following sections.

4.1 Asymptotic analysis and scaling

The separation of scales for the kinetic description manifests itself in different transport phenomena. The hierarchy of time scales in a fluid ranges from the time between particle collisions over the time needed for a fluid element to travel a typical distance up to the diffusion time scale. Similarly, the length scales range from the molecular mean free path l_{mfp} to the typical macroscopic length scale L. The ratio of these two length scales defines the *Knudsen number* $Kn = l_{\text{mfp}}/L$. In the lattice Boltzmann method, the mean free path corresponds to the lattice spacing a. To distinguish between different phenomena, the relation between the time and length scales characterizing the transport processes is important, the so-called *scaling*. In a fluid, we can expect two types of transport processes to be relevant. First wave-like phenomena, that obey *convective scaling* $\Delta t \sim \Delta x$, i.e., the time scale Δt is linearly related to the length scale Δx. And second diffusive phenomena with *diffusive scaling* $\Delta t \sim (\Delta x)^2$ where the time scale is quadratically related to the length scale. This suggests to consider three different time scales within the analysis of the lattice Boltzmann method: the lattice time scale τ, the convective time scale t_1 and the diffusive time scale t_2. The separation of length scales is guaranteed by a small Knudsen number. This is an important premise which, in addition to the low Mach number assumption, sets the limits within which the lattice Boltzmann method can reproduce hydrodynamic behavior.

4.2 Chapman-Enskog expansion

The lattice Boltzmann equation describes the system on microscopic lattice scales. In order to analyze the dynamics on hydrodynamic scales, we have to coarse-grain time and space. We introduce a small dimensionless scaling parameter ϵ, where the above suggests that $\epsilon = Kn$ is a natural choice.[1] A coarse-grained length scale is introduced by writing

$$\mathbf{r}_1 = \epsilon\,\mathbf{r}, \tag{4.1}$$

which corresponds to measuring positions with a coarse-grained ruler, e.g., instead of nanometers we can only resolve the position up to micrometers.

Further, we introduce the convective time scale t_1 and the diffusive time scale t_2 by

$$t_1 = \epsilon t, \qquad\qquad t_2 = \epsilon^2 t. \tag{4.2}$$

The time-scales can be interpreted as the different hands of a clock: The lattice time t is the sweep hand counting every clock-tick, t_1 is the minute hand, whereas t_2 is the hour hand. In the course of the LB algorithm, the hands advance according to the scaling (4.2). One time step corresponds to $t \to t + \tau$, while $t_1 \to t_1 + \epsilon\tau$ and $t_2 \to t_2 + \epsilon\tau^2$. Measuring a coarse-grained time corresponds to reading-off the minute and hour hand (t_1, t_2).

To analyze the dynamics on the coarse-grained scales, we write the lattice Boltzmann variables f_i as functions of $\mathbf{r}_1$, t_1 and t_2. The deterministic lattice Boltzmann equation is then

$$f_i(\mathbf{r}_1 + \epsilon\tau\mathbf{c}_i, t_1 + \epsilon\tau, t_2 + \epsilon^2\tau) = f_i(\mathbf{r}_1, t_1, t_2) + \Delta_i\left(\mathbf{f}(\mathbf{r}_1, t_1, t_2)\right). \tag{4.3}$$

While on the lattice scale $\mathbf{r}$ and t are discrete variables, the coarse-grained variables can be considered as continuous because ϵ is assumed to be very small. The lattice Boltzmann equation written in terms of the coarse-grained variables can therefore be Taylor-expanded. Up to order $\mathcal{O}(\epsilon^2)$, we get

$$\begin{aligned} f_i(\mathbf{r}_1 + \epsilon\tau\mathbf{c}_i, t_1 + \epsilon\tau, t_2 + \epsilon\tau^2) = f_i(\mathbf{r}_1, t_1, t_2) &+ \epsilon\tau\left(\frac{\partial}{\partial t_1} + \mathbf{c}_i\cdot\frac{\partial}{\partial \mathbf{r}_1}\right) f_i(\mathbf{r}_1, t_1, t_2) \\ &+ \epsilon^2\tau\left[\frac{\partial}{\partial t_2} + \frac{\tau}{2}\left(\frac{\partial}{\partial t_1} + \mathbf{c}_i\cdot\frac{\partial}{\partial \mathbf{r}_1}\right)^2\right] f_i(\mathbf{r}_1, t_1, t_2). \end{aligned} \tag{4.4}$$

Similarly to the space-time variables, also the LB populations and the collision operator are expanded in powers of the scaling parameter ϵ

$$f_i = f_i^{(0)} + \epsilon f_i^{(1)} + \epsilon^2 f_i^{(2)} + \mathcal{O}(\epsilon^3), \tag{4.5a}$$

$$\Delta_i = \Delta_i^{(0)} + \epsilon\Delta_i^{(1)} + \epsilon^2\Delta_i^{(2)} + \mathcal{O}(\epsilon^3). \tag{4.5b}$$

[1] In other words, the Chapman-Enskog expansion is a perturbation expansion in the Knudsen number.

Because the collision operator is a function of the LB populations, we can also write [5, 93]

$$\begin{aligned}\Delta_i\left(\mathbf{f}\right) = \Delta_i(\mathbf{f}^{(0)}) + \epsilon \sum_j \left.\frac{\partial \Delta_i}{\partial f_j}\right|_{\mathbf{f}^{(0)}} f_j^{(1)} \\ + \epsilon^2 \left(\sum_j \left.\frac{\partial \Delta_i}{\partial f_j}\right|_{\mathbf{f}^{(0)}} f_j^{(2)} + \sum_{j,k} \left.\frac{\partial^2 \Delta_i}{\partial f_j \partial f_k}\right|_{\mathbf{f}^{(0)}} f_j^{(1)} f_k^{(1)} \right) + \mathcal{O}(\epsilon^3).\end{aligned} \tag{4.6}$$

Since the conservation laws hold on all scales, i.e., independently of ϵ, the collision operator must satisfy mass and momentum conservation at all orders,

$$\sum_i \Delta_i^{(k)} = 0, \qquad \sum_i \Delta_i^{(k)} \mathbf{c}_i = 0, \tag{4.7}$$

for all k.

Inserting the expansions (4.4), (4.5a) and (4.5b) into (4.3) we get the (quasi-)continuous and scale separated version of the lattice Boltzmann equation

$$\begin{aligned}\epsilon \left(\frac{\partial}{\partial t_1} + \mathbf{c}_i \cdot \frac{\partial}{\partial \mathbf{r}_1} \right) f_i^{(0)} + \epsilon^2 \left[\frac{\partial}{\partial t_2} + \frac{\tau}{2} \left(\frac{\partial}{\partial t_1} + \mathbf{c}_i \cdot \frac{\partial}{\partial \mathbf{r}_1} \right)^2 \right] f_i^{(0)} \\ + \epsilon^2 \left(\frac{\partial}{\partial t_1} + \mathbf{c}_i \cdot \frac{\partial}{\partial \mathbf{r}_1} \right) f_i^{(1)} = \frac{1}{\tau} \left(\Delta_i^{(0)} + \epsilon \Delta_i^{(1)} + \epsilon^2 \Delta_i^{(2)} \right),\end{aligned} \tag{4.8}$$

where we have neglected all terms of order $\mathcal{O}(\epsilon^3)$. The different orders in (4.8) can be treated separately and we get a hierarchy of equations at different powers of ϵ:

$$\mathcal{O}(\epsilon^0): \quad \Delta_i^{(0)} = 0, \tag{4.9a}$$

$$\mathcal{O}(\epsilon^1): \quad \left(\frac{\partial}{\partial t_1} + \mathbf{c}_i \cdot \frac{\partial}{\partial \mathbf{r}_1} \right) f_i^{(0)} = \frac{1}{\tau} \Delta_i^{(1)}, \tag{4.9b}$$

$$\mathcal{O}(\epsilon^2): \quad \left[\frac{\partial}{\partial t_2} + \frac{\tau}{2} \left(\frac{\partial}{\partial t_1} + \mathbf{c}_i \cdot \frac{\partial}{\partial \mathbf{r}_1} \right)^2 \right] f_i^{(0)} + \left(\frac{\partial}{\partial t_1} + \mathbf{c}_i \cdot \frac{\partial}{\partial \mathbf{r}_1} \right) f_i^{(1)} = \frac{1}{\tau} \Delta_i^{(2)}. \tag{4.9c}$$

In the following, we will investigate these equations by constructing the moments on the different scales.

4.2.1 Zeroth order

On the zeroth order, the collision operator $\Delta_i^{(0)}$ vanishes and from (4.9a) it follows that $\mathbf{f}^{(0)}$ is a collisional invariant. The latter can hence be identified with the equilibrium distribution $\mathbf{f}^{\text{eq}}$. The local conserved variables must be moments of the equilibrium distribution

only, that is, mass density ρ and momentum density $\mathbf{j}$ can be written as

$$\begin{aligned} \sum_i f_i^{\mathrm{eq}} &= \rho, \\ \sum_i f_i^{\mathrm{eq}} \mathbf{c}_i &= \mathbf{j}. \end{aligned} \tag{4.10}$$

The vanishing zeroth order allows to simplify the collision operator. Neglecting all terms of order $\mathcal{O}(\epsilon^2)$ we obtain

$$\Delta_i = \epsilon \Delta_i^{(1)} = \epsilon \sum_j \left. \frac{\partial \Delta_i}{\partial f_j} \right|_{\mathbf{f}^{(0)}} f_j^{(1)} = \sum_j \mathcal{L}_{ij} \left(f_j - f_j^{\mathrm{eq}} \right), \tag{4.11}$$

where $\mathcal{L}_{ij} = \left. \frac{\partial \Delta_i}{\partial f_j} \right|_{\mathbf{f}^{(0)}}$. This justifies the use of a linear collision operator in the lattice Boltzmann method [18, 93].

4.2.2 First order

The zeroth and first moment of the ϵ-order equation (4.9b) are

$$\frac{\partial}{\partial t_1} \rho + \frac{\partial}{\partial \mathbf{r}_1} \cdot \mathbf{j} = 0, \tag{4.12a}$$

$$\frac{\partial}{\partial t_1} \mathbf{j} + \frac{\partial}{\partial \mathbf{r}_1} \cdot \Pi^{(0)} = 0, \tag{4.12b}$$

where $\Pi^{(0)} = \sum_i f_i^{(0)} \mathbf{c}_i \mathbf{c}_i$ is the equilibrium momentum flux. These are exactly the inviscid fluid equations when the equilibrium momentum flux is equal to the Euler stress

$$\Pi^{(0)} = p \mathbb{1} + \rho \mathbf{u}\mathbf{u}, \tag{4.13}$$

where p is the scalar fluid pressure. For further reference, we calculate the second moment equation on this order which yields

$$\frac{\partial}{\partial t_1} \Pi^{(0)} + \frac{\partial}{\partial \mathbf{r}_1} \cdot \Phi^{(0)} = \frac{1}{\tau} \sum_i \Delta_i^{(1)} \mathbf{c}_i \mathbf{c}_i = \frac{1}{\tau} \left(\Pi^{*(1)} - \Pi^{(1)} \right). \tag{4.14}$$

$\Phi^{(0)} = \sum_i f_i^{(0)} \mathbf{c}_i \mathbf{c}_i \mathbf{c}_i$ is the equilibrium third moment and $\Pi^{*(1)}$ is the ϵ-order of the post-collisional momentum flux.

4.2.3 Second order

Before we construct the moments on the ϵ^2-order, we rewrite equation (4.9c) by inserting the ϵ-order which eliminates the second derivatives

$$\frac{\partial}{\partial t_2} f_i^{(0)} + \frac{1}{2}\left(\frac{\partial}{\partial t_1} + \mathbf{c}_i \cdot \frac{\partial}{\partial \mathbf{r}_1}\right)\left(f_i^{*(1)} + f_i^{(1)}\right) = \frac{1}{\tau}\Delta_i^{(2)}. \tag{4.15}$$

We have written $f_i^{*(1)} = f_i^{(1)} + \Delta_i^{(1)}$ for the $\mathcal{O}(\epsilon)$ post-collisional population. The equations for the zeroth and first moment then come as

$$\frac{\partial}{\partial t_2}\rho = 0, \tag{4.16a}$$

$$\frac{\partial}{\partial t_2}\mathbf{j} + \frac{1}{2}\frac{\partial}{\partial \mathbf{r}_1} \cdot \left(\Pi^{*(1)} + \Pi^{(1)}\right) = 0. \tag{4.16b}$$

Here we have used that the zeroth and first moment of $\mathbf{f}^{(1)}$ vanish. In the following, we will merge the moment equations of the different orders to obtain a single equation in the variables $\mathbf{r}$ and t.

4.2.4 Merging orders

The macroscopic fields depend on the coarse-grained variables $\mathbf{r}_1$, t_1 and t_2, and thus indirectly on the lattice length and time $\mathbf{r}$ and t. The derivatives with respect to the lattice variables come as

$$\begin{aligned} \frac{\partial}{\partial \mathbf{r}} &= \epsilon \frac{\partial}{\partial \mathbf{r}_1}, \\ \frac{\partial}{\partial t} &= \epsilon \frac{\partial}{\partial t_1} + \epsilon^2 \frac{\partial}{\partial t_2}. \end{aligned} \tag{4.17}$$

Up to the order $\mathcal{O}(\epsilon^2)$ of the Chapman-Enskog expansion, the second-order of the populations $f_i^{(2)}$ and the collision operator $\Delta_i^{(2)}$ do not show up in the zeroth and first moment equations, therefore we can set

$$\begin{aligned} f_i^{\text{eq}} &= f_i^{(0)}, \\ f_i^{\text{neq}} &= \epsilon f_i^{(1)}, \\ \Delta_i &= \epsilon \Delta_i^{(1)}. \end{aligned} \tag{4.18}$$

We merge (4.12a) and (4.12b) with (4.16a) and (4.16b) and obtain the combined equations for mass and momentum

$$\frac{\partial}{\partial t}\rho + \frac{\partial}{\partial \mathbf{r}} \cdot \mathbf{j} = 0, \tag{4.19a}$$

$$\frac{\partial}{\partial t}\mathbf{j} + \frac{\partial}{\partial \mathbf{r}}\Pi^{\text{eq}} + \frac{1}{2}\frac{\partial}{\partial \mathbf{r}}\left(\Pi^{*\text{neq}} + \Pi^{\text{neq}}\right) = 0, \tag{4.19b}$$

where $\Pi^{\text{eq}} = \Pi^{(0)}$ and $\Pi^{\text{neq}} = \epsilon\Pi^{(1)}$. The mass equation is the continuity equation which is automatically satisfied by the lattice Boltzmann equation. The momentum equation resembles the the Navier-Stokes equation if Π^{eq} equals the Euler stress and the pre- and post-collisional non-equilibrium stresses are related to the Newtonian viscous stress σ , cf. equation (2.19)

$$\Pi^{*\text{neq}} + \Pi^{\text{neq}} = -2\sigma \tag{4.20}$$

In order to evaluate this expression explicitly, we need to use the specific equilibrium distribution and the collision operator of the LB model. Below we will use the D3Q19 model introduced in section 2.5 to finalize the Chapman-Enskog expansion.

4.2.5 Closing the Chapman-Enskog expansion

Using the $\mathcal{O}(\mathbf{u}^2)$ polynomial expansion of the equilibrium distribution, we can calculate the equilibrium third moment

$$\Phi^{(0)}_{\alpha\beta\gamma} = \sum_i f_i^{\text{eq}} c_{i\alpha}c_{i\beta}c_{i\gamma} = \frac{\rho u_\delta}{c_s^2}\sum_i w_i c_{i\alpha}c_{i\beta}c_{i\gamma}c_{i\delta} = \rho c_s^2\left(u_\alpha\delta_{\beta\gamma} + u_\beta\delta_{\alpha\gamma} + u_\gamma\delta_{\alpha\beta}\right), \tag{4.21}$$

where we have exploited that only the even-rank lattice sums contribute. Equation (4.14) then becomes

$$\Pi^{*(1)}_{\alpha\beta} - \Pi^{(1)}_{\alpha\beta} = \tau\frac{\partial}{\partial t_1}\left(\rho c_s^2\delta_{\alpha\beta} + \rho u_\alpha u_\beta\right) + \tau c_s^2\frac{\partial}{\partial r_{1\gamma}}\left(\rho u_\alpha\delta_{\beta\gamma} + \rho u_\beta\delta_{\alpha\gamma} + \rho u_\gamma\delta_{\alpha\beta}\right). \tag{4.22}$$

The time derivative of the Euler stress can be expressed with the help of ϵ-order moment equations (4.12a) and (4.12b)

$$\begin{aligned}
&\frac{\partial}{\partial t_1}\left(\rho c_s^2\delta_{\alpha\beta} + \rho u_\alpha u_\beta\right)\\
&\quad= \rho c_s^2\delta_{\alpha\beta}\frac{\partial}{\partial t_1}\rho + u_\beta\left(\frac{\partial}{\partial t_1}\rho u_\alpha\right) + u_\alpha\left(\frac{\partial}{\partial t_1}\rho u_\beta\right) - u_\alpha u_\beta\frac{\partial}{\partial t_1}\rho\\
&\quad= -\rho c_s^2\delta_{\alpha\beta}\left(\frac{\partial}{\partial r_{1\gamma}}\rho u_\gamma\right) - c_s^2 u_\beta\delta_{\alpha\gamma}\left(\frac{\partial}{\partial r_{1\gamma}}\rho\right) - c_s^2 u_\alpha\delta_{\beta\gamma}\left(\frac{\partial}{\partial r_{1\gamma}}\rho\right) + \mathcal{O}(u^3),
\end{aligned} \tag{4.23}$$

where we have neglected terms of $\mathcal{O}(u^3)$. Inserting this into (4.22) yields

$$\Pi^{*(1)}_{\alpha\beta} - \Pi^{(1)}_{\alpha\beta} = \rho c_s^2\tau\left(\frac{\partial}{\partial r_{1\alpha}}u_\beta + \frac{\partial}{\partial r_{1\beta}}u_\alpha\right). \tag{4.24}$$

Using (4.17) we can write this in unscaled form

$$\Pi^{*\text{neq}}_{\alpha\beta} - \Pi^{\text{neq}}_{\alpha\beta} = \rho c_s^2\tau\left(\frac{\partial}{\partial r_\alpha}u_\beta + \frac{\partial}{\partial r_\beta}u_\alpha\right). \tag{4.25}$$

With the linear collision operator $\mathcal{L}_{ij}$, alternative expressions for the non-equilibrium stresses were obtained in (2.73)

$$\begin{aligned}\overline{\Pi}^{*\text{neq}}_{\alpha\beta} &= (1+\lambda_s)\,\overline{\Pi}^{\text{neq}}_{\alpha\beta},\\ \Pi^{*\text{neq}}_{\alpha\alpha} &= (1+\lambda_b)\,\Pi^{\text{neq}}_{\alpha\alpha}.\end{aligned} \tag{4.26}$$

From this equation system we obtain a solution for the non-equilibrium stresses in terms of velocity gradients

$$\begin{aligned}\overline{\Pi}^{\text{neq}}_{\alpha\beta} &= \frac{\rho c_s^2\tau}{\lambda_s}\left(\frac{\partial}{\partial r_\alpha}u_\beta + \frac{\partial}{\partial r_\beta}u_\alpha - \frac{2}{3}\frac{\partial}{\partial r_\gamma}u_\gamma\delta_{\alpha\beta}\right),\\ \Pi^{\text{neq}}_{\alpha\alpha} &= \frac{2\rho c_s^2\tau}{\lambda_b}\frac{\partial}{\partial r_\alpha}u_\alpha.\end{aligned} \tag{4.27}$$

This yields the viscous stresses

$$\begin{aligned}\overline{\sigma}_{\alpha\beta} &= -\frac{1}{2}\left(\overline{\Pi}^{*\text{neq}}_{\alpha\beta} + \overline{\Pi}^{\text{neq}}_{\alpha\beta}\right) = -\frac{\rho c_s^2\tau}{2}\frac{2+\lambda_s}{\lambda_s}\left(\frac{\partial}{\partial r_\alpha}u_\beta + \frac{\partial}{\partial r_\beta}u_\alpha - \frac{2}{3}\frac{\partial}{\partial r_\gamma}u_\gamma\delta_{\alpha\beta}\right),\\ \sigma_{\alpha\alpha} &= -\frac{1}{2}\left(\Pi^{*\text{neq}}_{\alpha\alpha} + \Pi^{\text{neq}}_{\alpha\alpha}\right) = -\rho c_s^2\tau\frac{2+\lambda_b}{\lambda_b}\frac{\partial}{\partial r_\alpha}u_\alpha.\end{aligned} \tag{4.28}$$

By comparing with the Newtonian form we find the relation between the eigenvalues λ_s and λ_b and the shear and bulk viscosities

$$\eta_s = -\frac{\rho c_s^2\tau}{2}\frac{2+\lambda_s}{\lambda_s} = -\rho c_s^2\tau\left(\frac{1}{\lambda_s}+\frac{1}{2}\right), \tag{4.29}$$

$$\eta_b = -\frac{\rho c_s^2\tau}{3}\frac{2+\lambda_b}{\lambda_b} = -\frac{2}{3}\rho c_s^2\tau\left(\frac{1}{\lambda_b}+\frac{1}{2}\right). \tag{4.30}$$

This result closes the Chapman-Enskog expansion of the lattice Boltzmann equation. The additional $\frac{1}{2}$ in the brackets is a lattice correction originating at the ϵ^2-scale. This correction is the reason that, despite the underlying lattice structure, Galilean invariance is restored at the macroscopic level. Strictly speaking, there comes another correction term due to the $\mathcal{O}(u^3)$ terms in (4.23). This would be of the form $\frac{\partial}{\partial r_\beta}\frac{\partial}{\partial r_\gamma}\rho u_\alpha u_\beta u_\gamma$, and hence there are $\mathcal{O}(u^2)$ corrections to the viscosity [58]. The resulting inaccuracy in the momentum equation can be compensated by including higher orders in the Hermite expansion corresponding to third order velocity terms in the equilibrium distribution [47]. However, for nearly incompressible flows at low Mach number the corrections are very small and can be neglected completely.

The higher order terms of the populations $f_i^{(2)}$ and $\Delta_i^{(2)}$ do not contribute to the momentum equation up to $\mathcal{O}(\epsilon^2)$. Therefore the eigenvalues of the collision operator corresponding to the kinetic modes are irrelevant on the Navier-Stokes level. The reason is that the Navier-Stokes equation contains only gradients of the velocity field. The second derivatives of the flow field only contribute to the kinetic modes, which do not appear on the macroscopic level. However, the second derivatives are important for accurate boundary conditions, as

shown by Ginzburg and d'Humières [64]. In that case, the kinetic eigenvalues have to be tuned to specific values, sometimes called "magic values", that yield second-order accurate boundary conditions. More down-to-earth, they can be obtained from the second-order Chapman-Enskog solution. We will therefore derive explicit expressions for $f_i^{(1)}$ and $f_i^{(2)}$ in the following.

4.2.6 Explicit expressions for $\mathrm{f}^{(1)}$ and $\mathrm{f}^{(2)}$

Instead of using the third equilibrium moment $\Pi^{(0)}$, we can also evaluate the ϵ^1-scale equation (4.9b) directly. Using the polynomial expansion for the equilibrium distribution and inserting the mass and momentum equation we get

$$\begin{aligned}
\left(\frac{\partial}{\partial t_1}+c_{i\alpha}\frac{\partial}{\partial r_{1\alpha}}\right) f_i^{(0)} &= \left(\frac{\partial}{\partial t_1}+c_{i\alpha}\frac{\partial}{\partial r_{1\alpha}}\right)\frac{w_i}{c_s^2}\left[\rho c_s^2+\rho u_\beta c_{i\beta}+\frac{\rho u_\beta u_\gamma}{2c_s^2}\left(c_{i\beta}c_{i\gamma}-c_s^2\delta_{\beta\gamma}\right)\right] \\
&= \frac{w_i}{c_s^2}\left\{\frac{\partial}{\partial r_{1\alpha}}\left[\rho c_s^2+\rho u_\beta c_{i\beta}+\frac{\rho u_\beta u_\gamma}{2c_s^2}\left(c_{i\beta}c_{i\gamma}-c_s^2\delta_{\beta\gamma}\right)\right]c_{i\alpha}\right. \\
&\qquad -\frac{\partial}{\partial r_{1\alpha}}\left[c_s^2\rho u_\alpha+\left(\rho c_s^2\delta_{\alpha\beta}+\rho u_\alpha u_\beta\right)c_{i\beta}\right] \\
&\qquad -\frac{u_\beta}{2c_s^2}\left(c_{i\beta}c_{i\gamma}-c_s^2\delta_{\beta\gamma}\right)\frac{\partial}{\partial r_{1\alpha}}\left(\rho c_s^2\delta_{\alpha\gamma}+\rho u_\alpha u_\gamma\right) \\
&\qquad -\frac{u_\gamma}{2c_s^2}\left(c_{i\beta}c_{i\gamma}-c_s^2\delta_{\beta\gamma}\right)\frac{\partial}{\partial r_{1\alpha}}\left(\rho c_s^2\delta_{\alpha\beta}+\rho u_\alpha u_\beta\right) \\
&\qquad \left.+u_\beta u_\gamma\left(c_{i\beta}c_{i\gamma}-c_s^2\delta_{\beta\gamma}\right)\frac{\partial}{\partial r_{1\alpha}}\left(\rho u_\alpha\right)\right\} \\
&= \frac{w_i}{c_s^2}\left\{\frac{\partial}{\partial r_{1\alpha}}\rho u_\beta\left(c_{i\alpha}c_{i\beta}-c_s^2\delta_{\alpha\beta}\right)\right. \\
&\qquad +\frac{\partial}{\partial r_{1\alpha}}\frac{\rho u_\beta u_\gamma}{2c_s^2}\left(c_{i\alpha}c_{i\beta}c_{i\gamma}-c_s^2c_{i\alpha}\delta_{\beta\gamma}-c_s^2c_{i\beta}\delta_{\alpha\gamma}-c_s^2c_{i\gamma}\delta_{\alpha\beta}\right) \\
&\qquad \left.-u_\beta\frac{\partial}{\partial r_{1\alpha}}\rho\left(c_{i\alpha}c_{i\beta}-c_s^2\delta_{\alpha\beta}\right)\right\}+\mathcal{O}(u^3) \\
&= \rho\frac{\partial}{\partial r_{1\alpha}}u_\beta E^{(2)}_{i\alpha\beta}+\frac{\partial}{\partial r_{1\alpha}}\frac{\rho u_\beta u_\gamma}{2c_s}E^{(3)}_{i\alpha\beta\gamma}+\mathcal{O}(u^3),
\end{aligned} \tag{4.31}$$

where we have introduced tensors $\mathsf{E}^{(2)}$ and $\mathsf{E}^{(3)}$ as shorthand notations

$$\begin{aligned}
E^{(2)}_{i\alpha\beta} &= \frac{w_i}{c_s^2}\left(c_{i\alpha}c_{i\beta}-c_s^2\delta_{\alpha\beta}\right), \\
E^{(3)}_{i\alpha\beta\gamma} &= \frac{w_i}{c_s^3}\left(c_{i\alpha}c_{i\beta}c_{i\gamma}-c_s^2c_{i\alpha}\delta_{\beta\gamma}-c_s^2c_{i\beta}\delta_{\alpha\gamma}-c_s^2c_{i\gamma}\delta_{\alpha\beta}\right).
\end{aligned} \tag{4.32}$$

Note that these tensors are dimensionless and have no projection onto the conserved modes. The time derivatives in (4.31) have been expressed in terms of spatial derivatives with the

help of the moment equations, and terms of $\mathcal{O}(u^3)$ have been neglected according to the low Mach number approximation. With the linear collision operator, the ϵ^1-scale equation now comes as

$$\frac{1}{\tau}\sum_j \mathcal{L}_{ij} f_j^{(1)} = \rho \frac{\partial}{\partial r_{1\alpha}} u_\beta \overline{E_{i\alpha\beta}^{(2)}} + \frac{\rho}{3}\frac{\partial}{\partial r_{1\alpha}} u_\beta E_{i\gamma\gamma}^{(2)} \delta_{\alpha\beta} + \frac{\partial}{\partial r_{1\alpha}} \frac{\rho u_\beta u_\gamma}{2c_s} E_{i\alpha\beta\gamma}^{(3)} \tag{4.33}$$

where we have decomposed the tensor $E_{i\alpha\beta}^{(2)}$ into its trace and traceless part. Now we use the specific form of the collision operator in mode space. By projecting equation (4.33) onto the modes we can assign the terms to the different symmetry classes: $\overline{E_{i\alpha\beta}^{(2)}}$ has a projection onto the shear stress modes only, $E_{i\gamma\gamma}^{(2)}\delta_{\alpha\beta}$ has a projection onto the bulk stress modes only, while $E_{i\alpha\beta\gamma}^{(3)}$ has projections onto the kinetic modes only. We can therefore use the eigenvalues of the collision operator to write

$$f_i^{(1)} = \frac{\rho\tau}{\lambda_s}\frac{\partial}{\partial r_{1\alpha}} u_\beta \overline{E_{i\alpha\beta}^{(2)}} + \frac{\rho\tau}{3\lambda_b}\frac{\partial}{\partial r_{1\alpha}} u_\beta E_{i\gamma\gamma}^{(2)}\delta_{\alpha\beta} + \frac{\tau}{\lambda_g}\frac{\partial}{\partial r_{1\alpha}}\frac{\rho u_\beta u_\gamma}{2c_s} E_{i\alpha\beta\gamma}^{(3)}, \tag{4.34}$$

where we have assumed that all kinetic (ghost) modes have the same eigenvalue λ_g. From this, we can determine the non-equilibrium stresses

$$\begin{aligned}
\overline{\Pi}_{\alpha\beta}^{(1)} &= \sum_i f_i^{(1)} c_{i\alpha} c_{i\beta} = \frac{\rho\tau}{\lambda_s}\frac{\partial}{\partial r_{1\gamma}} u_\delta \sum_i \overline{E_{i\gamma\delta}^{(2)}} c_{i\alpha} c_{i\beta} \\
&= \frac{\rho c_s^2 \tau}{\lambda_s}\frac{\partial}{\partial r_{1\gamma}} u_\delta \left(\delta_{\alpha\gamma}\delta_{\beta\delta} + \delta_{\alpha\delta}\delta_{\beta\gamma} - \frac{2}{3}\delta_{\alpha\beta}\delta_{\gamma\delta} \right), \\
\Pi_{\alpha\alpha}^{(1)} &= \sum_i f_i^{(1)} c_{i\alpha} c_{i\alpha} = \frac{\rho\tau}{3\lambda_b}\frac{\partial}{\partial r_{1\beta}} u_\beta \sum_i E_{i\gamma\gamma}^{(2)} c_{i\alpha} c_{i\alpha} \\
&= \frac{2\rho c_s^2 \tau}{\lambda_b}\frac{\partial}{\partial r_{1\beta}} u_\beta,
\end{aligned} \tag{4.35}$$

in accordance with (4.27). We know already, that there are no contributions from $f_i^{(2)}$ and $\Delta_i^{(2)}$ to the Navier-Stokes equation. However, here we are interested in the full Chapman-Enskog solution of the populations f_i up to second order. We therefore continue with the ϵ^2-scale equation (4.9c),

$$\begin{aligned}
\frac{1}{\tau}\sum_j \mathcal{L}_{ij} f_j^{(2)} &= \left[\frac{\partial}{\partial t_2} + \frac{\tau}{2}\left(\frac{\partial}{\partial t_1} + c_{i\alpha}\frac{\partial}{\partial r_{1\alpha}} \right)^2 \right] f_i^{(0)} + \left(\frac{\partial}{\partial t_1} + c_{i\alpha}\frac{\partial}{\partial r_{1\alpha}} \right) f_i^{(1)} \\
&= \frac{\partial}{\partial t_2} f_i^{(0)} + \left(\frac{\partial}{\partial t_1} + c_{i\alpha}\frac{\partial}{\partial r_{1\alpha}} \right) \left(\frac{1}{2}\sum_j \mathcal{L}_{ij} f_j^{(1)} + f_i^{(1)} \right).
\end{aligned} \tag{4.36}$$

We are interested in the solution $f_i^{(2)}$ up to second derivatives in the velocity field and ignore all higher order terms. That is, in the following we will neglect all third and higher derivatives of the velocity, and all second and higher derivatives of the mass and the nonlinear terms.

First we evaluate the t_2 time derivative of the equilibrium distribution

$$\frac{\partial}{\partial t_2} f_i^{(0)} = \frac{\partial}{\partial t_2} \frac{w_i}{c_s^2} \left[\rho c_s^2 + \rho u_\beta c_{i\beta} + \frac{\rho u_\beta u_\gamma}{2c_s^2} \left(c_{i\beta} c_{i\gamma} - c_s^2 \delta_{\beta\gamma} \right) \right]. \tag{4.37}$$

On the t_2-scale, the fluid is incompressible $\frac{\partial \rho}{\partial t_2} = 0$. The time derivative of the momentum can be replaced by spatial derivatives using the $\mathcal{O}(\epsilon^2)$ momentum equation

$$\begin{aligned} \frac{\partial}{\partial t_2} \rho u_\alpha &= -\frac{1}{2} \frac{\partial}{\partial r_{1\beta}} \left(\Pi_{\alpha\beta}^{*(1)} + \Pi_{\alpha\beta}^{(1)} \right) \\ &= \eta_s \frac{\partial^2}{\partial r_{1\beta} \partial r_{1\gamma}} u_\delta \left(\delta_{\alpha\gamma}\delta_{\beta\delta} + \delta_{\alpha\delta}\delta_{\beta\gamma} - \frac{2}{3}\delta_{\alpha\beta}\delta_{\gamma\delta} \right) + \eta_b \frac{\partial^2}{\partial r_{1\beta} \partial r_{1\gamma}} u_\delta \delta_{\alpha\beta}\delta_{\gamma\delta}. \end{aligned} \tag{4.38}$$

The t_2 derivative of the nonlinear term produces terms of the form $u_\alpha \frac{\partial}{\partial r_{1\beta}} \frac{\partial}{\partial r_{1\gamma}} u_\delta$ which are of order $\epsilon^2 u^2$ and can also be neglected.

Next we turn to the t_1 time derivative of the first-order solution $f_i^{(1)}$. From equation (4.34) we see that it produces only terms that are at least of order $\epsilon^2 u^2$, hence $\frac{\partial}{\partial t_1} f_i^{(1)}$ can be neglected. Dropping also the second derivative of the nonlinear term, equation (4.36) comes as

$$\begin{aligned} \frac{1}{\tau} \sum_j \mathcal{L}_{ij} f_j^{(2)} &= \frac{w_i}{c_s^2} \frac{\partial}{\partial t_2} \rho u_\beta c_{i\beta} + c_{i\alpha} \frac{\partial}{\partial r_{1\alpha}} \left(\frac{1}{2} \sum_j \mathcal{L}_{ij} f_j^{(1)} + f_i^{(1)} \right) \\ &= \frac{w_i \eta_s}{c_s^2} \frac{\partial^2}{\partial r_{1\alpha} \partial r_{1\gamma}} u_\delta \left(c_{i\delta}\delta_{\alpha\gamma} + c_{i\gamma}\delta_{\alpha\delta} - \frac{2}{3} c_{i\alpha}\delta_{\gamma\delta} \right) \\ &\quad + \frac{w_i \eta_b}{c_s^2} \frac{\partial^2}{\partial r_{1\alpha} \partial r_{1\gamma}} u_\delta c_{i\alpha}\delta_{\gamma\delta} \\ &\quad + c_{i\alpha} \frac{\partial}{\partial r_{1\alpha}} \left[\left(\frac{1}{2} + \frac{1}{\lambda_s} \right) \rho\tau \frac{\partial}{\partial r_{1\gamma}} u_\beta \overline{E_{i\beta\gamma}^{(2)}} \right. \\ &\qquad \left. + \left(\frac{1}{2} + \frac{1}{\lambda_b} \right) \frac{\rho\tau}{3} \frac{\partial}{\partial r_{1\gamma}} u_\beta E_{i\gamma\gamma}^{(2)} \delta_{\beta\gamma} \right] \qquad (4.39a) \\ &= \frac{w_i \eta_s}{c_s^2} \frac{\partial^2}{\partial r_{1\alpha} \partial r_{1\gamma}} u_\delta \left(c_{i\delta}\delta_{\alpha\gamma} + c_{i\gamma}\delta_{\alpha\delta} - \frac{2}{3} c_{i\alpha}\delta_{\gamma\delta} \right) \\ &\quad + \frac{w_i \eta_b}{c_s^2} \frac{\partial^2}{\partial r_{1\alpha} \partial r_{1\gamma}} u_\delta c_{i\alpha}\delta_{\gamma\delta} \\ &\quad - c_{i\alpha} \frac{\partial}{\partial r_{1\alpha}} \left[\frac{w_i \eta_s}{c_s^4} \frac{\partial}{\partial r_{1\gamma}} u_\beta \left(c_{i\beta} c_{i\gamma} - \frac{1}{3} c_{i\delta} c_{i\delta} \delta_{\beta\gamma} \right) \right. \\ &\qquad \left. + \frac{w_i \eta_b}{2c_s^4} \frac{\partial}{\partial r_{1\gamma}} u_\beta \left(c_{i\delta} c_{i\delta} - 3c_s^2 \right) \delta_{\beta\gamma} \right] \end{aligned}$$

$$
\begin{aligned}
&= -\eta_s \frac{\partial^2}{\partial r_{1\alpha}\partial r_{1\gamma}} u_\beta \frac{w_i}{c_s^2}\left(\frac{1}{c_s^2}c_{i\alpha}c_{i\beta}c_{i\gamma} - c_{i\beta}\delta_{\alpha\gamma} - c_{i\gamma}\delta_{\alpha\beta}\right.\\
&\qquad\qquad\left. + \frac{2}{3}c_{i\alpha}\delta_{\beta\gamma} - \frac{1}{3c_s^2}c_{i\delta}c_{i\delta}c_{i\alpha}\delta_{\beta\gamma}\right)\\
&\quad - \frac{3\eta_b}{2}\frac{\partial^2}{\partial r_{1\alpha}\partial r_{1\gamma}} u_\beta \frac{w_i}{c_s^2}\left(\frac{1}{3c_s^2}c_{i\delta}c_{i\delta}c_{i\alpha}\delta_{\beta\gamma} - \frac{5}{3}c_{i\alpha}\delta_{\beta\gamma}\right)\\
&= -\frac{\eta_s}{c_s}\frac{\partial^2 u_\beta}{\partial r_{1\alpha}\partial r_{1\gamma}}\overline{E^{(3)}_{i\alpha\beta\gamma}} - \frac{\eta_b}{2c_s}\frac{\partial^2 u_\beta}{\partial r_{1\alpha}\partial r_{1\gamma}}E^{(3)}_{i\alpha\delta\delta}\delta_{\beta\gamma},
\end{aligned}
\tag{4.39b}
$$

where the traceless part of the third-rank tensor is to be understood in the following way:

$$
\overline{E^{(3)}_{i\alpha\beta\gamma}} = E^{(3)}_{i\alpha\beta\gamma} - \frac{1}{3}E^{(3)}_{i\alpha\delta\delta}\delta_{\beta\gamma}. \tag{4.40}
$$

Since $E^{(3)}_{i\alpha\beta\gamma}$ projects only on the kinetic modes, the final result for $f_i^{(2)}$ is

$$
f_i^{(2)} = -\frac{\eta_s\tau}{c_s\lambda_g}\frac{\partial^2 u_\beta}{\partial r_{1\alpha}\partial r_{1\gamma}}\overline{E^{(3)}_{i\alpha\beta\gamma}} - \frac{\eta_b\tau}{2c_s\lambda_g}\frac{\partial^2 u_\beta}{\partial r_{1\alpha}\partial r_{1\gamma}}E^{(3)}_{\alpha\delta\delta}\delta_{\beta\gamma}. \tag{4.41}
$$

Putting all parts together and using unscaled variables, we get the second order Chapman-Enskog solution for the lattice Boltzmann populations f_i

$$
\begin{aligned}
f_i = w_i\,\rho &+ \frac{w_i c_{i\alpha}}{c_s^2}\rho u_\alpha + \frac{\rho u_\alpha u_\beta}{2c_s^2}E^{(2)}_{i\alpha\beta}\\
&+ \frac{\rho\tau}{\lambda_s}\frac{\partial}{\partial r_\alpha}u_\beta\overline{E^{(2)}_{i\alpha\beta}} + \frac{\rho\tau}{3\lambda_b}\frac{\partial}{\partial r_\alpha}u_\alpha E^{(2)}_{i\gamma\gamma} + \frac{\tau}{c_s\lambda_g}\frac{\partial}{\partial r_\alpha}\frac{\rho u_\beta u_\gamma}{2}E^{(3)}_{i\alpha\beta\gamma}\\
&- \frac{\eta_s\tau}{c_s\lambda_g}\frac{\partial^2 u_\beta}{\partial r_\alpha\partial r_\gamma}\overline{E^{(3)}_{i\alpha\beta\gamma}} - \frac{\eta_b\tau}{2c_s\lambda_g}\frac{\partial^2 u_\beta}{\partial r_\alpha\partial r_\beta}E^{(3)}_{\alpha\delta\delta}.
\end{aligned}
\tag{4.42}
$$

A similar expression has been derived by Ginzburg and d'Humières [64]. There, however, the two-relaxation time model was used which makes the expressions slightly simpler. In contrast, we have here derived the more general solution for the case of independent relaxation rates for shear and bulk modes. The kinetic modes are all relaxed with the same eigenvalue λ_g. In principle, we could also use independent eigenvalues for the kinetic modes by splitting the tensor $\mathsf{E}^{(3)}$ into symmetry-related parts. Since this makes the calculations unnecessarily tedious, we have not done this here.

The second-order solution (4.42) shows that the Chapman-Enskog procedure yields an expansion of the LB populations in terms of the conserved hydrodynamic fields and the gradients of the velocity field, while gradients of the mass density do not contribute. The equilibrium distribution depends on the mass and momentum densities exclusively, and the non-equilibrium contributions are obtained as derivatives of the flow velocity with increasing order. On the Navier-Stokes level, only the gradients of the velocity play a role which enter on the ϵ-scale. As a consequence of the lattice discretization, the ϵ^2-scale yields an additional correction to the viscosity which is not present in the Chapman-Enskog expansion of the continuous Boltzmann equation [50, 52]. The various truncations we have made do

not effect the dynamics on the Navier-Stokes level as long as the Mach number is small. In particular, the error terms stemming from the ϵ^3-scale are negligible [65], and the truncation error in the viscosity is of order $\mathcal{O}(u^2)$ [58]. In the continuous case, the Chapman-Enskog expansion can be continued at higher orders to yield the Burnett and super-Burnett equations. For the lattice Boltzmann equation, this requires to take into account higher-order Hermite approximations of the equilibrium distribution, but in principle it is also possible. It should also be remarked that we have only looked at the mass and momentum equations, but not the heat transport equation (cf. [94] for a treatment of thermal transport).

Another type of asymptotic analysis was carried out by Junk et al. [92], which differs from the Chapman-Enskog expansion in some aspects. First, a purely diffusive scaling $\Delta t \sim \Delta x^2$ is used, while acoustic effects are considered as numerical artifacts. Second, all quantities including the hydrodynamic variables are cast in a regular expansion, whereas in the Chapman-Enskog procedure the hydrodynamic fields are usually not expanded (for reasons explained in Ref. [8]).

The Chapman-Enskog expansion makes the requirements for a successful lattice Boltzmann model explicit, i.e., symmetry requirements and conservation laws. This makes it possible to construct LB models without referring to the Hermite-expansion of the continuous Boltzmann equation. From another point of view, asymptotic expansions like Chapman-Enskog can also be viewed as a means of validating a given LB model. This is specifically important when one pursues extensions of the conventional LB models like multi-phase lattice Boltzmann. Furthermore, initial and boundary conditions have to be validated by asymptotic expansion to show that they are well-behaved in terms of Navier-Stokes hydrodynamics. For initial conditions, this has been carried out by Mei et al. [95], Caiazzo [96]. Asymptotic expansions have been applied to various boundary conditions by Junk and Yang [97] and have led to the development of more accurate reflection rules by Ginzburg and d'Humières [64]. This will be discussed in more detail in chapter 5.

4.2.7 Fluctuations

In the analysis of the preceding subsections, we have not taken into account fluctuations. It is straightforward to incorporate the fluctuating part by setting

$$\Pi^{*\mathrm{neq}} + \Pi^{\mathrm{neq}} = -2\left(\sigma + \mathsf{s}\right) \tag{4.43}$$

and using the stochastic collision operator $\Delta_i = \sum_j \mathcal{L}_{ij} f_j^{\mathrm{neq}} + \Delta_i'$ such that

$$\begin{aligned} \overline{\Pi}_{\alpha\beta}^{*\mathrm{neq}} &= (1+\lambda_s)\overline{\Pi}_{\alpha\beta}^{\mathrm{neq}} + \overline{\sigma}_{\alpha\beta}^{r}, \\ \Pi_{\alpha\alpha}^{*\mathrm{neq}} &= (1+\lambda_b)\Pi_{\alpha\alpha}^{\mathrm{neq}} + \sigma_{\alpha\alpha}^{r}. \end{aligned} \tag{4.44}$$

Using (4.25) we can eliminate the post-collisional stress $\Pi^{*\mathrm{neq}}$ and get

$$\begin{aligned} \overline{\Pi}_{\alpha\beta}^{\mathrm{neq}} &= \frac{\rho c_s^2 \tau}{\lambda_s}\left(\frac{\partial}{\partial r_\alpha}u_\beta + \frac{\partial}{\partial r_\beta}u_\alpha - \frac{2}{3}\frac{\partial}{\partial r_\gamma}u_\gamma \delta_{\alpha\beta}\right) - \frac{1}{\lambda_s}\overline{\sigma}_{\alpha\beta}^{r}, \\ \Pi_{\alpha\alpha}^{\mathrm{neq}} &= \frac{2\rho c_s^2 \tau}{\lambda_b}\frac{\partial}{\partial r_\alpha}u_\alpha - \frac{1}{\lambda_b}\sigma_{\alpha\alpha}^{r}. \end{aligned} \tag{4.45}$$

Inserting into (4.43) leads to

$$\begin{aligned}\overline{\sigma}_{\alpha\beta} + \overline{s_{\alpha\beta}} &= -\frac{\rho c_s^2 \tau}{2}\frac{2+\lambda_s}{\lambda_s}\left(\frac{\partial}{\partial r_\alpha}u_\beta + \frac{\partial}{\partial r_\beta}u_\alpha - \frac{2}{3}\frac{\partial}{\partial r_\gamma}u_\gamma\delta_{\alpha\beta}\right) + \frac{1}{\lambda_s}\overline{\sigma}^r_{\alpha\beta},\\ \sigma_{\alpha\alpha} + s_{\alpha\alpha} &= -\rho c_s^2\tau\frac{2+\lambda_b}{\lambda_b}\frac{\partial}{\partial r_\alpha}u_\alpha + \frac{1}{\lambda_b}\sigma^r_{\alpha\alpha},\end{aligned} \tag{4.46}$$

from which we can read off

$$\begin{aligned}\overline{s_{\alpha\beta}} &= \frac{1}{\lambda_s}\overline{\sigma}^r_{\alpha\beta},\\ s_{\alpha\alpha} &= \frac{1}{\lambda_b}\sigma^r_{\alpha\alpha}.\end{aligned} \tag{4.47}$$

Therefore the fluctuations $s_{\alpha\beta}$ on the hydrodynamic level are related to the random stresses $\sigma^r_{\alpha\beta}$ in the stochastic collision operator via

$$\begin{aligned}\left\langle \overline{s_{\alpha\beta}}\,\overline{s_{\gamma\delta}}\right\rangle &= \frac{1}{\lambda_s^2}\left\langle \overline{\sigma}^r_{\alpha\beta}\overline{\sigma}^r_{\gamma\delta}\right\rangle,\\ \left\langle s_{\alpha\alpha}s_{\beta\beta}\right\rangle &= \frac{1}{\lambda_b^2}\left\langle \sigma^r_{\alpha\alpha}\sigma^r_{\beta\beta}\right\rangle,\\ \left\langle \overline{s_{\alpha\beta}}s_{\gamma\gamma}\right\rangle &= \frac{1}{\lambda_s\lambda_b}\left\langle \overline{\sigma}^r_{\alpha\beta}\sigma^r_{\gamma\gamma}\right\rangle.\end{aligned} \tag{4.48}$$

The fluctuations on the hydrodynamic level are thus different from the fluctuations on the lattice Boltzmann level because the former are present on the convective time scale t_1, while the latter enter on the lattice time scale τ.

Comparing with the expressions (3.13) for fluctuating hydrodynamics,

$$\left\langle s_{\alpha\beta}s_{\gamma\delta}\right\rangle = \frac{2k_BT}{a^3\tau}\eta_{\alpha\beta\gamma\delta} = \frac{2k_BT\eta_s}{a^3\tau}\left(\delta_{\alpha\gamma}\delta_{\beta\delta} + \delta_{\alpha\delta}\delta_{\beta\gamma} - \frac{2}{3}\delta_{\alpha\beta}\delta_{\gamma\delta}\right) + \frac{2k_BT\eta_b}{a^3\tau}\delta_{\alpha\beta}\delta_{\gamma\delta}, \tag{4.49}$$

and taking into account the results (4.29) and (4.30) for the shear and bulk viscosities we finally arrive at

$$\begin{aligned}\left\langle \overline{\sigma}^r_{\alpha\beta}\overline{\sigma}^r_{\gamma\delta}\right\rangle &= -\frac{\rho c_s^2 k_BT}{a^3}\lambda_s(2+\lambda_s)\left(\delta_{\alpha\gamma}\delta_{\beta\delta} + \delta_{\alpha\delta}\delta_{\beta\gamma} - \frac{2}{3}\delta_{\alpha\beta}\delta_{\gamma\delta}\right),\\ \left\langle \sigma^r_{\alpha\alpha}\sigma^r_{\beta\beta}\right\rangle &= -\frac{6\rho c_s^2 k_BT}{a^3}\lambda_b(2+\lambda_b),\\ \left\langle \overline{\sigma}^r_{\alpha\beta}\sigma^r_{\gamma\gamma}\right\rangle &= 0.\end{aligned} \tag{4.50}$$

In section 3.3.4, we have shown that for consistency with statistical mechanics, the variance of the k-th mode must satisfy $\varphi_k^2 = \mu\rho b_k(1-\gamma_k^2)$. Using the formulas (2.67), the b_k-values

in (2.64) for the D3Q19 model and $c_s^2 = 1/3(a/\tau)^2$, we get

$$\begin{aligned}
\langle \sigma_{\alpha\alpha}\sigma_{\beta\beta}\rangle &= \varphi_4^2 = 6\mu\rho c_s^4(1-\gamma_b^2)\\
\langle \overline{\sigma}^r_{xx}\overline{\sigma}^r_{xx}\rangle &= \frac{1}{9}\varphi_5^2 = \frac{4}{3}\mu\rho c_s^4(1-\gamma_s^2)\\
\left\langle \overline{\sigma}^r_{yy}\overline{\sigma}^r_{yy}\right\rangle &= \langle \overline{\sigma}^r_{zz}\overline{\sigma}^r_{zz}\rangle = \left\langle \overline{\sigma}^r_{yy}\overline{\sigma}^r_{zz}\right\rangle = \frac{1}{36}\varphi_5^2 + \frac{1}{4}\varphi_6^2 = \frac{4}{3}\mu\rho c_s^4(1-\gamma_s^2)\\
\left\langle \overline{\sigma}^r_{xx}\overline{\sigma}^r_{yy}\right\rangle &= \langle \overline{\sigma}^r_{xx}\overline{\sigma}^r_{zz}\rangle = -\frac{1}{18}\varphi_5^2 = -\frac{2}{3}\mu\rho c_s^4(1-\gamma_s^2)\\
\left\langle \overline{\sigma}^r_{xy}\overline{\sigma}^r_{xy}\right\rangle &= \varphi_7^2 = \mu\rho c_4^2(1-\gamma_s^2)\\
\langle \overline{\sigma}^r_{xz}\overline{\sigma}^r_{xz}\rangle &= \varphi_8^2 = \mu\rho c_4^2(1-\gamma_s^2)\\
\left\langle \overline{\sigma}^r_{yz}\overline{\sigma}^r_{yz}\right\rangle &= \varphi_9^2 = \mu\rho c_4^2(1-\gamma_s^2).
\end{aligned} \tag{4.51}$$

Plugging in $\gamma_k = 1 + \lambda_k$ and $\mu c_s^2 = k_B T/a^3$ we see that these expressions are equivalent to (3.19). This shows that the result does not only recover fluctuating hydrodynamics, but it is at the same time consistent with statistical mechanics. The fluctuations of the kinetic modes do not influence the hydrodynamic behavior because the non-equilibrium parts of Φ and other kinetic modes do not appear at the Navier-Stokes level. They are only important for proper thermalization on microscopic scales.

4.2.8 External forces

So far we have only looked at the lattice Boltzmann equation without external forces, where momentum is strictly conserved. In many applications it is desirable to be able to transfer momentum to the fluid by an external force density $\mathbf{g}(\mathbf{r}, t)$. In the same spirit as in the case of fluctuations, we incorporate the effect of the external force by adding an additional term Δ_i^g to the collision operator

$$\Delta_i = \sum_j \mathcal{L}_{ij} f_j^{\text{neq}} + \Delta_i^g. \tag{4.52}$$

While the external forces have no effect on the mass density, they transfer an amount $\mathbf{g}\tau$ of momentum to the fluid in one time step. Therefore, the zeroth and first moment of Δ_i^g have to satisfy

$$\sum_i \Delta_i^g = 0, \tag{4.53a}$$

$$\sum_i \Delta_i^g \mathbf{c}_i = \mathbf{g}\tau. \tag{4.53b}$$

Since the momentum before and after the collision differ, but the collisions are assumed to take place instantaneously, the hydrodynamic momentum density is not uniquely defined. Any value between the pre- and the post-collisional value could be used. Consequently, there is an ambiguity which value to use for calculating the equilibrium distribution f_i^{eq}. In the

literature, different propositions have been made to define the momentum density [8, 64, 65]. Here, we continue without an a-priori definition and use the Chapman-Enskog expansion to deduce an appropriate choice afterwards. For this purpose, we introduce the following notations to distinguish between the momentum densities obtained from the different orders of the Chapman-Enskog expansion

$$\mathbf{j}' = \sum_i f_i \mathbf{c}_i = \mathbf{j}^{(0)} + \epsilon \mathbf{j}^{(1)}, \tag{4.54}$$

where

$$\begin{aligned} \mathbf{j}^{(0)} &= \sum_i f_i^{(0)} \mathbf{c}_i, \\ \mathbf{j}^{(1)} &= \sum_i f_i^{(1)} \mathbf{c}_i. \end{aligned} \tag{4.55}$$

Since momentum is not conserved, $\mathbf{j}^{(1)}$ is not necessarily equal to zero.

The forcing term must enter the Chapman-Enskog expansion at order $\mathcal{O}(\epsilon)$, hence

$$\Delta_i^g = \epsilon \Delta_i^{g(1)} + \epsilon^2 \Delta_i^{g(2)} + \mathcal{O}(\epsilon^3). \tag{4.56}$$

As previously, we can expand the lattice Boltzmann equation and evaluate the moments at different orders of ϵ. The first three moments at $\mathcal{O}(\epsilon)$ are

$$\begin{aligned} &\frac{\partial}{\partial t_1}\rho + \frac{\partial}{\partial \mathbf{r}_1} \cdot \mathbf{j}^{(0)} = 0, \\ &\frac{\partial}{\partial t_1}\mathbf{j}^{(0)} + \frac{\partial}{\partial \mathbf{r}_1} \cdot \Pi^{(0)} = g^{(1)}, \\ &\frac{\partial}{\partial t_1}\Pi^{(0)} + \frac{\partial}{\partial \mathbf{r}_1} \cdot \Phi^{(0)} = \frac{1}{\tau}\left(\Pi^{*(1)} - \Pi^{(1)}\right). \end{aligned} \tag{4.57}$$

Here we identify $f_i^{(0)}$ with the equilibrium distribution f_i^{eq}, where we plug in $\mathbf{u}^{(0)} = \mathbf{j}^{(0)}/\rho$ for the flow velocity. We can also evaluate $\Pi^{(0)}$ and $\Phi^{(0)}$. This yields a similar result as in (4.24), but with additional terms due to the forcing contribution in the momentum flux

$$\Pi_{\alpha\beta}^{*(1)} - \Pi_{\alpha\beta}^{(1)} = \rho c_s^2 \tau \left(\frac{\partial}{\partial r_{1\alpha}} u_\beta^{(0)} + \frac{\partial}{\partial r_{1\beta}} u_\alpha^{(0)}\right) + \tau \left(u_\alpha^{(0)} g_\beta^{(1)} + g_\alpha^{(1)} u_\beta^{(0)}\right) + \mathcal{O}(u^3). \tag{4.58}$$

A second relation is again obtained from the collision operator

$$\Pi_{\alpha\beta}^{*(1)} - \Pi_{\alpha\beta}^{(1)} = \lambda_s \overline{\Pi}_{\alpha\beta}^{(1)} + \frac{\lambda_b}{3} \Pi_{\gamma\gamma}^{(1)} \delta_{\alpha\beta} + \sum_i \Delta_i^{g(1)} c_{i\alpha} c_{i\beta}. \tag{4.59}$$

Solving the equation system as before yields

$$\begin{aligned}
\overline{\Pi}^{*(1)}_{\alpha\beta} + \overline{\Pi}^{(1)}_{\alpha\beta} &= \frac{\rho c_s^2 \tau (2+\lambda_s)}{\lambda_s} \left(\overline{\frac{\partial}{\partial r_{1\alpha}} u^{(0)}_\beta} + \overline{\frac{\partial}{\partial r_{1\beta}} u^{(0)}_\alpha} \right) \\
&\quad + \frac{\tau(2+\lambda_s)}{\lambda_s} \left(\overline{u^{(0)}_\alpha g^{(1)}_\beta} + \overline{g^{(1)}_\alpha u^{(0)}_\beta} \right) - \frac{2}{\lambda_s} \sum_i \Delta_i^{g(1)} \overline{c_{i\alpha} c_{i\beta}}, \\
\Pi^{*(1)}_{\alpha\alpha} + \Pi^{(1)}_{\alpha\alpha} &= \frac{2\rho c_s^2 \tau (2+\lambda_b)}{\lambda_b} \frac{\partial}{\partial r_{1\alpha}} u^{(0)}_\alpha \\
&\quad + \frac{2\tau(2+\lambda_b)}{\lambda_b} u^{(0)}_\alpha g^{(1)}_\alpha - \frac{2}{\lambda_b} \sum_i \Delta_i^{g(1)} c_{i\alpha} c_{i\alpha}.
\end{aligned} \tag{4.60}$$

The additional terms due to the forcing can be compensated if the second moment of the collision operator is made to satisfy

$$\begin{aligned}
\sum_i \Delta_i^{g(1)} \overline{c_{i\alpha} c_{i\beta}} &= \frac{(2+\lambda_s)\tau}{2} \left(\overline{u^{(0)}_\alpha g^{(1)}_\beta} + \overline{g^{(1)}_\alpha u^{(0)}_\beta} \right), \\
\sum_i \Delta_i^{g(1)} c_{i\alpha} c_{i\alpha} &= (2+\lambda_b) \tau u^{(0)}_\alpha g^{(1)}_\alpha.
\end{aligned} \tag{4.61}$$

Proceeding to the order $\mathcal{O}(\epsilon^2)$ where the zeroth and first moment equations are

$$\begin{aligned}
&\frac{\partial}{\partial t_2} \rho + \frac{\partial}{\partial \mathbf{r}_1} \cdot \left(\mathbf{j}^{(1)} + \frac{1}{2} \tau \mathbf{g}^{(1)} \right) = 0, \\
&\frac{\partial}{\partial t_2} \mathbf{j}^{(0)} + \frac{\partial}{\partial t_1} \left(\mathbf{j}^{(1)} + \frac{1}{2} \tau \mathbf{g}^{(1)} \right) + \frac{1}{2} \frac{\partial}{\partial \mathbf{r}_1} \cdot \left(\Pi^{*(1)} + \Pi^{(1)} \right) = \mathbf{g}^{(2)}.
\end{aligned} \tag{4.62}$$

Inserting the above result for $\Pi^{(1)}$ in the momentum equation yields

$$\begin{aligned}
&\frac{\partial}{\partial t_2} j^{(0)}_\alpha + \frac{\partial}{\partial t_1} \left(j^{(1)}_\alpha + \frac{1}{2} \tau g^{(1)}_\alpha \right) \\
&\quad + \rho c_s^2 \tau \frac{\partial}{\partial r_{1\beta}} \left[\frac{2+\lambda_s}{2\lambda_s} \left(\overline{\frac{\partial}{\partial r_{1\alpha}} u^{(0)}_\beta} + \overline{\frac{\partial}{\partial r_{1\beta}} u^{(0)}_\alpha} \right) + \frac{2+\lambda_b}{3\lambda_b} \frac{\partial}{\partial r_{1\gamma}} u^{(0)}_\gamma \delta_{\alpha\beta} \right] = g^{(2)}_\alpha,
\end{aligned} \tag{4.63}$$

where we have used (4.60) and (4.61). After merging orders we arrive at

$$\begin{aligned}
&\frac{\partial}{\partial t} \rho + \frac{\partial}{\partial r_\alpha} \left(j'_\alpha + \frac{1}{2} \tau g_\alpha \right) = 0, \\
&\frac{\partial}{\partial t} \left(j'_\alpha + \frac{1}{2} \tau g_\alpha \right) + \frac{\partial}{\partial r_\beta} \left(\rho c_s^2 \delta_{\alpha\beta} + \rho u^{(0)}_\alpha u^{(0)}_\beta \right) \\
&\quad - \frac{\partial}{\partial r_\beta} \left[\eta_s \left(\frac{\partial}{\partial r_\alpha} u^{(0)}_\beta + \frac{\partial}{\partial r_\beta} u^{(0)}_\alpha - \frac{2}{3} \frac{\partial}{\partial r_\gamma} u^{(0)}_\gamma \delta_{\alpha\beta} \right) + \eta_b \frac{\partial}{\partial r_\gamma} u^{(0)}_\gamma \delta_{\alpha\beta} \right] = g_\alpha.
\end{aligned} \tag{4.64}$$

This can be cast in the form of the Navier-Stokes equation by using the following definition for the hydrodynamic momentum density:

$$\mathbf{j} \equiv \mathbf{j}^{(0)} \equiv \mathbf{j}' + \frac{1}{2}\tau\mathbf{g} = \sum_i f_i \mathbf{c}_i + \frac{1}{2}\tau\mathbf{g}. \tag{4.65}$$

Note that this implies

$$\begin{aligned} \sum_i f_i^{\mathrm{eq}} \mathbf{c}_i &= \mathbf{j}, \\ \sum_i f_i^{\mathrm{neq}} \mathbf{c}_i &= -\frac{1}{2}\tau\mathbf{g}. \end{aligned} \tag{4.66}$$

The definition (4.65) corresponds to the arithmetic mean of the pre- and post-collisional momentum density. This has been determined as the optimal value previously, in both numerical and theoretical studies. However, in those works the redefined value was not plugged in the equilibrium distribution. While Ladd and Verberg [65] used $\mathbf{j}'$ and accordingly $\sum_i f_i^{\mathrm{neq}} \mathbf{c}_i = 0$, in Ginzburg and d'Humières [64] two different values are used: $\mathbf{j}'$ for the linear part in f_i^{eq}, and $\mathbf{j}' + \frac{1}{2}\tau\mathbf{g}$ for the nonlinear part. Since only the non-linear part enters the equilibrium stress tensor, this makes no difference in the usual implementation of an MRT model. However, the mixed use of different momentum densities makes the theoretical derivation rather obscure. In contrast, our redefinition is strictly compatible with the Chapman-Enskog expansion and all spurious terms are canceled for a proper choice of the forcing term. Let us define

$$G_{\alpha\beta} = \frac{2+\lambda_s}{2}\left(u_\alpha g_\beta + g_\alpha u_\beta - \frac{2}{3}u_\gamma g_\gamma \delta_{\alpha\beta}\right) + \frac{2+\lambda_b}{3}u_\gamma g_\gamma \delta_{\alpha\beta}. \tag{4.67}$$

The forcing term is determined from the conditions (4.53a), (4.53b) and (4.61), and can be written as

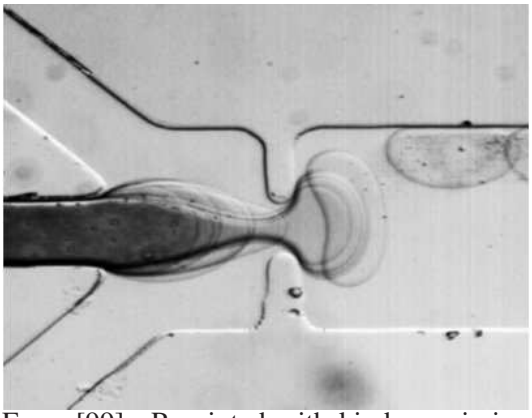
$$\Delta_i^g = \frac{w_i\,\tau}{c_s^2}\left[\mathbf{g}\cdot\mathbf{c}_i + \frac{1}{2c_s^2}\,\mathsf{G} : \left(\mathbf{c}_i\mathbf{c}_i - c_s^2\mathbb{1}\right)\right]. \tag{4.68}$$

This expression leads to the Navier-Stokes equation with the same viscosities as in the case without forcing. Moreover, no additional assumptions about the external force have to be made, i.e., the result holds for time-varying and inhomogeneous forces as well. It was first derived by Guo et al. [98], while in other works only constant or homogeneous forces were treated [65].

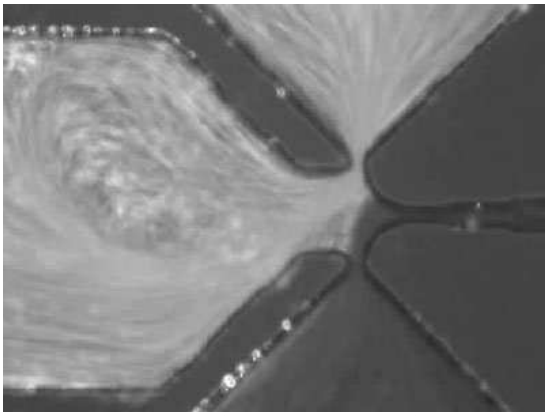
5 Boundary conditions for lattice Boltzmann models

The lattice Boltzmann method has become a popular approach for simulating complex fluids and soft matter. This is due to its mesoscopic nature and the underlying kinetic picture, which opens up the possibility to augment the model with more microscopic information beyond the Navier-Stokes level. For example, one can simulate particulate suspensions where the solute particles do not only interact which each other directly, but also via the exchange of momentum that is propagated through the surrounding solvent, so called solvent-mediated or *hydrodynamic interactions*. In this context, the lattice Boltzmann method plays the role of an efficient solver for the momentum propagation in the solvent and meanwhile is an established alternative to other mesoscopic solvent models, such as dissipative particle dynamics and multi-particle collision dynamics (see for example Yeomans [7] for a recent comparison of lattice Boltzmann and multi-particle collision dynamics). The presence of solute particles makes it necessary to deal with interactions between the fluid and the solid phase. For geometrically extended objects like colloids or walls, boundary conditions have to be invoked at the object's surface to prevent the fluid from leaking into the solid. Another case where boundary conditions become important are confined flows, that is, the fluid is bounded in a geometrical domain of relatively small dimensions. Confined flows can exhibit interesting behavior even at low Reynolds numbers [16], and the observed phenomena depend strongly on the precise nature of the boundary condition. Even a straight wall can give rise to complicated effects, such as boundary layer effects and the Kramers problem [5, 6]. Such effects are of paramount importance in modern microfluidics: The so called "lab on a chip"-paradigm has led to the construction of microfluidic devices where fluids can be manipulated on the micro- or even the nanoscale. Figure 5.1 shows two examples of such microfluidic devices. Due to the large surface-to-volume ratio of such structures, the flow behavior is to a wide extent dominated by surface and interface effects. A successful simulation of the flow through a microfluidic device therefore depends critically on appropriate modeling of the boundary conditions.

One of the main objectives of this thesis was the development and implementation of boundary conditions for the lattice Boltzmann method. It is sometimes perceived that boundaries can be readily mapped to the lattice and hence boundary conditions are simple to implement, e.g., by the bounce-back rule [5]. On closer inspection, however, it turns out that numerous difficulties arise and an efficient and at the same time accurate treatment of the boundary can be an intricate affair [101]. The challenges one is faced with while developing boundary conditions shall be discussed in detail in this chapter. It is organized as follows: In section 5.1 we discuss boundary conditions in hydrodynamics and kinetic theory. Then we move on to boundary conditions in lattice Boltzmann models in section 5.2, where we review the



From [99]. Reprinted with kind permission from Prof. Charles Baroud, LadHyX, Ecole Polytechnique.



From [100]. Reprinted with permission from AAAS.

Figure 5.1: Two examples of microfluidic devices. In the left device, a water stream is injected into an oil phase and passes a "gate" after which droplets are formed. The right device is a microfluidic "memory". Two fluids are injected into the device, a bright one and a dark one. By imposing a pressure peak in the narrow channel on the right, the flow can be switched such that the dark fluid streams to the top and the bright fluid streams to the bottom. In this way the "state" of the device can serve to store a binary number.

commonly used boundary models. In section 5.3, we turn to the phenomenon of boundary slippage and develop a novel boundary condition for tunable slip.

Algorithmic boundary conditions

Before we proceed with the boundary conditions at the surface of solid objects, we shall mention another type of boundary conditions that is omnipresent in computer simulations. Due to memory limitations, any computer simulation is necessarily restricted to a finite simulation volume. Hence, one has to define a rule for what should happen at the boundaries of the finite simulation domain. Such rules are usually also referred to as boundary conditions. Let us call them *algorithmic boundary conditions* here, in order to distinguish from solid-fluid or *hydrodynamic boundary conditions*. The most commonly used algorithmic boundary conditions are periodic boundaries, where any mass portion that leaves the simulation domain on one side, simultaneously re-enters the domain at the opposite side. This rule conserves mass and usually momentum by construction[1], and the simulated system becomes effectively infinite whilst finite-size corrections are taken into account. Another possibility are in- and outflow boundary conditions, where the mass and momentum flux at the domain boundary is prescribed by some rule. For example this could be known values from an alternative simulation of a similar system. A special case of in- and outflow boundary conditions

[1] It is possible to modify the momentum of the periodically re-entering mass portion, but this is then usually not referred to as periodic boundary conditions any more. For example, Lees-Edwards boundary conditions can be applied to generate a shear-profile in the simulation domain.

are open boundaries, where mass just flows out of the simulation domain. Finally, it is of course possible to employ a solid boundary condition at the domain boundary. In this case, the distinction between algorithmic and hydrodynamic boundary conditions is immaterial. Algorithmic boundary conditions will be discussed in conjunction with the implementation of a lattice Boltzmann kernel in appendix A. In the remainder of this section, we will focus exclusively on hydrodynamic boundary conditions at solid surfaces.

5.1 Hydrodynamic boundary conditions

5.1.1 Boundary conditions on the Navier-Stokes level

The Navier-Stokes equation of classical hydrodynamics is a partial differential equation. From the viewpoint of mathematics, the partial differential equation itself is not enough to determine a unique solution, but we have to pose the boundary value problem, that is, the initial and boundary conditions have to be specified.The boundary value problem corresponds to filtering from all admissible solutions to the Navier-Stokes equation exactly the ones that satisfy the specific initial and boundary conditions. Typical boundary conditions for the Navier-Stokes equation are either Dirichlet or Neumann boundary conditions. The Dirichlet boundary conditions prescribes the value of the flow velocity at the boundary

$$\mathbf{u}(\mathbf{r}_B) = \mathbf{U}_B, \tag{5.1}$$

while the Neumann boundary condition prescribes a value to the normal derivatives of the velocity and can be written in the form

$$\mathbf{n} \cdot \left.\frac{\partial \mathbf{u}}{\partial \mathbf{r}}\right|_{\mathbf{r}_B} = \mathbf{h}_B. \tag{5.2}$$

In classical hydrodynamics, it is usually assumed that the fluid at the surface moves along with the same velocity as the solid object. This is the so called *stick* or *no-slip* boundary condition. For an object at rest it reduces to

$$\mathbf{u}(\mathbf{r}_B) = 0, \tag{5.3}$$

i.e., the flow velocity at the boundary is zero. Conversely, the *free-slip* or *full-slip* boundary condition assumes that the fluid glides freely over the surface, which can be expressed by a vanishing normal velocity gradient at the surface. To ensure impermeability of the solid, the normal velocity must equal the normal velocity of the surface.

$$\mathbf{n} \cdot \left.\frac{\partial \mathbf{u}}{\partial \mathbf{r}}\right|_{\mathbf{r}_B} = 0, \qquad \mathbf{n} \cdot \mathbf{u}(\mathbf{r}_B) = \mathbf{n} \cdot \mathbf{U}_B. \tag{5.4}$$

The full-slip boundary condition is an example of a mixed Dirichlet and Neumann boundary condition, which in conjunction with a curved boundary is sometimes referred to as a

Cauchy boundary condition. Virtually all applications in classical hydrodynamics make use of the no-slip boundary condition. It is well justified on the macroscopic level where the characteristic scales of the flow are much larger then molecular scales, i.e., at small Knudsen number. The no-slip boundary condition is also often used as a reference case for the development of boundary conditions in kinetic models like lattice Boltzmann.

In microfluidic devices, the separation between molecular and hydrodynamic length scales is less pronounced and finite Knudsen number effects come into play. Flows can generally be divided into different regimes according to the value of the Knudsen number: continuum flows at small Knudsen numbers $Kn \leq 0.001$, slip flows at intermediate Knudsen numbers $0.001 \leq Kn \leq 0.1$, the transition regime at higher Knudsen numbers $0.1 < Kn < 10$, and free molecular flows beyond $Kn \geq 10$ [102, 103]. The continuum hydrodynamic description of fluids remains valid up to $Kn \sim 0.1$. Microflows are typically in the slip-flow regime, where the Navier-Stokes equation remains valid but the no-slip boundary condition breaks down [104, 105]. It has been observed in various experiments that the velocity of the flow does not entirely vanish at the surface [102, 106–109]. The appearance of the apparent slip velocity is a consequence of the microscopic structure of the surface and its interactions with the fluid, e.g., the wetting properties. The effects of apparent slip in a microfluidic device may include a reduction of the surface stresses such that the flow throughput can significantly be enhanced [110, 111].

The latter observation suggests to formulate a simple linear constitutive equation for the stress at the surface [111]

$$\mathbf{n} \cdot \sigma = \gamma \left(\mathbf{u}(\mathbf{r}_B) - \mathbf{U}_B \right) \equiv \gamma \, \mathbf{u}_{\text{slip}}. \tag{5.5}$$

Combining this with the expression (2.2) for the Newtonian viscous stress in the bulk we arrive at the following slip-flow boundary condition

$$\mathbf{n} \cdot \mathbf{u}(\mathbf{r}_B) = \mathbf{n} \cdot \mathbf{U}_B, \qquad \mathbf{n} \cdot \left. \frac{\partial \mathbf{u}}{\partial \mathbf{r}} \right|_{\mathbf{r}_B} = \frac{\gamma}{\eta_s} \, \mathbf{u}_{\text{slip}}. \tag{5.6}$$

If we choose the coordinate system such that the boundary normal is in z-direction and the flow is in x-direction, we can rewrite (5.6) as the Navier slip boundary condition [112]

$$u_{\text{slip}} = \frac{\eta_s}{\gamma} \left. \frac{\partial u_x}{\partial z} \right|_{z_B} = \delta_B \left. \frac{\partial u_x}{\partial z} \right|_{z_B}. \tag{5.7}$$

Here the slip length $\delta_B = \eta_s / \gamma$ is introduced, which can be illustrated as the distance into the solid at which the linearly extrapolated flow profile is equal to zero, cf. figure 5.2. The limiting case $\delta_B = 0$ corresponds to the no-slip boundary condition, whereas for $\delta_B \to \infty$ equation (5.7) resembles the full-slip boundary condition. A negative slip length indicates an apparent change in the flow direction close to the boundary. According to its definition, the slip length depends on the viscosity η_s and the coefficient γ. It is thus the ratio of a bulk property and a surface property. In other words, the parameter that truly describes the properties of the surface is not the slip length δ_B but the coefficient γ. Therefore, the latter

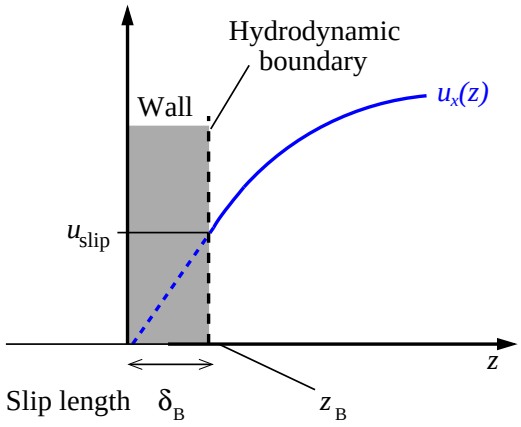


Figure 5.2: Illustration of the slip-length. The flow profile has a finite slip velocity u_{slip} at the hydrodynamic boundary z_B. The slip length is defined as the distance at which the linearly extrapolated profile is equal to zero.

should be varied to model different solid-fluid interfaces. This suggests furthermore that it is desirable to be able to implement the coefficient γ directly in a simulation method. One such method is developed within the lattice Boltzmann model in section 5.3.

5.1.2 Boundary conditions in kinetic theory

Boundary conditions in kinetic theory have the same mathematical origin as Navier-Stokes boundary conditions: the Boltzmann equation is a differential equation and thus the boundary value problem has to be solved. However, the physical picture is quite different because kinetic theory describes the system at much smaller scales, namely at the level of the one-particle distribution function. Therefore molecular details of the interaction between fluid particles and the solid surface can not be ignored completely, but have to be incorporated appropriately into the kinetic description. Influencing factors are for example surface roughness and chemical details of the solid-fluid interaction such as hydrophobicity or chemical bonding. At the mesoscopic level of description, the interactions should be characterized by their statistical properties [110]. The boundary conditions therefore have to be formulated in terms of the distribution functions and transition probabilities, respectively. One of the first systematic accounts of boundary conditions in the kinetic theory of fluids was presented by Maxwell in the appendix of [113]. A more recent overview can be found in the book by Cercignani [51].

A key feature of kinetic boundary conditions is that the velocity space has to be split into incoming velocities and outgoing velocities according to the projection $\mathbf{n} \cdot \mathbf{v}$ onto the wall normal [114–116]. The distinction between incoming and outgoing velocities is in general related to a discontinuity in the distribution function f. The simplest case is a perfectly flat elastic surface, which reflects the impinging particles in such a way that their normal velocity component is reversed while the other components remain unchanged. This is

called *specular reflection* and is described by the relation

$$f(\mathbf{r}_B, \mathbf{v}, t) = f(\mathbf{r}_B, \mathbf{v} - 2\mathbf{n}(\mathbf{n}\cdot\mathbf{v}), t), \qquad \mathbf{n}\cdot\mathbf{v} > 0. \tag{5.8}$$

However, the idealized flat elastic surface is an unrealistic assumption and the specular reflection rule is in general not applicable in practice.

In general, the effect of the surface can be described by a scattering kernel $\mathcal{B}(\mathbf{v}' \to \mathbf{v})$ indicating the probability that an impinging particle with velocity $\mathbf{v}'$ will be deflected to a new velocity $\mathbf{v}$. The distribution function for the reflected (outgoing) particles can then be written in the form [51]

$$|\mathbf{n}\cdot\mathbf{v}|\, f(\mathbf{r}_B, \mathbf{v}, t) = \int_{\mathbf{n}\cdot\mathbf{v}'<0} d\mathbf{v}'\, \mathcal{B}(\mathbf{v}' \to \mathbf{v})\, |\mathbf{n}\cdot\mathbf{v}'|\, f(\mathbf{r}_B, \mathbf{v}', t), \qquad \mathbf{n}\cdot\mathbf{v} > 0. \tag{5.9}$$

The scattering kernel must be non-negative. If the surface is impermeable and non-adsorbing, every impinging particle is re-emitted and the scattering kernel satisfies the normalization condition

$$\int_{\mathbf{n}\cdot\mathbf{v}>0} \mathcal{B}(\mathbf{v}' \to \mathbf{v})\, d\mathbf{v} = 1. \tag{5.10}$$

This is equivalent to the statement that the normal hydrodynamic momentum density at the boundary vanishes:

$$\begin{aligned}
\mathbf{n}\cdot\rho\mathbf{u} &= \int d\mathbf{v}\, \mathbf{n}\cdot\mathbf{v} f(\mathbf{r}_B, \mathbf{v}, t) \\
&= \int_{\mathbf{n}\cdot\mathbf{v}>0} d\mathbf{v}\ |\mathbf{n}\cdot\mathbf{v}|\, f(\mathbf{r}_B, \mathbf{v}, t) - \int_{\mathbf{n}\cdot\mathbf{v}<0} d\mathbf{v}\ |\mathbf{n}\cdot\mathbf{v}|\, f(\mathbf{r}_B, \mathbf{v}, t) \\
&= \int_{\mathbf{n}\cdot\mathbf{v}>0} d\mathbf{v} \int_{\mathbf{n}\cdot\mathbf{v}'<0} d\mathbf{v}' \mathcal{B}(\mathbf{v}' \to \mathbf{v})\, |\mathbf{n}\cdot\mathbf{v}'|\, f(\mathbf{r}_B, \mathbf{v}', t) - \int_{\mathbf{n}\cdot\mathbf{v}<0} d\mathbf{v}\, |\mathbf{n}\cdot\mathbf{v}|\, f(\mathbf{r}_B, \mathbf{v}, t) \\
&= \int_{\mathbf{n}\cdot\mathbf{v}'<0} d\mathbf{v}'\, |\mathbf{n}\cdot\mathbf{v}'|\, f(\mathbf{r}_B, \mathbf{v}', t) - \int_{\mathbf{n}\cdot\mathbf{v}<0} d\mathbf{v}\, |\mathbf{n}\cdot\mathbf{v}|\, f(\mathbf{r}_B, \mathbf{v}, t) = 0,
\end{aligned} \tag{5.11}$$

where we have inserted (5.9) and used (5.10). Specular reflections are just the special case where

$$\mathcal{B}(\mathbf{v}' \to \mathbf{v}) = \delta\left(\mathbf{v}' - \mathbf{v} + 2\mathbf{n}(\mathbf{n}\cdot\mathbf{v})\right). \tag{5.12}$$

If the dynamics of the system at the molecular level is time-reversible, the scattering kernel satisfies the detailed balance condition for thermal equilibrium (cf. figure 5.3) [51]

$$|\mathbf{n}\cdot\mathbf{v}'|\, \mathcal{B}(\mathbf{v}' \to \mathbf{v}) f^{\mathrm{eq}}(\mathbf{v}') = |\mathbf{n}\cdot\mathbf{v}|\, \mathcal{B}(-\mathbf{v} \to -\mathbf{v}') f^{\mathrm{eq}}(\mathbf{v}), \tag{5.13}$$

where f^{eq} denotes the Maxwell-Boltzmann equilibrium distribution. This implies that the equilibrium distribution automatically satisfies the boundary condition

$$|\mathbf{n}\cdot\mathbf{v}|\, f^{\mathrm{eq}}(\mathbf{v}) = \int_{\mathbf{n}\cdot\mathbf{v}'<0} d\mathbf{v}'\ |\mathbf{n}\cdot\mathbf{v}'|\, \mathcal{B}(\mathbf{v}' \to \mathbf{v}) f^{\mathrm{eq}}(\mathbf{v}'). \tag{5.14}$$

Figure 5.3: Illustration of boundary conditions in kinetic theory. (Left) Detailed balance: the blue scattering process $\mathbf{v} \to \mathbf{v}'$ has the same probability as the red time reversed process $-\mathbf{v}' \to \mathbf{v}$. (Right) A particle emerging from a surface element dA with a velocity $\mathbf{v}$ propagates into a skewed volume element $dV = dA\,|\mathbf{n}\cdot\mathbf{v}|\,dt$. This is the reason for the factor $|\mathbf{n}\cdot\mathbf{v}|$ in the kinetic boundary conditions.

In other words, if the incoming particles have a Maxwellian distribution, then the outgoing particles have the same Maxwellian distribution. It should be noted that (5.14) is a weaker requirement than (5.13).

The construction of the scattering kernel $\mathcal{B}(\mathbf{v}' \to \mathbf{v})$ is a formidable task because of the complexity of the underlying microscopic fluid-solid interactions. Usually one has to resort to simplified models that satisfy the basic requirements of the boundary condition such as mass conservation and detailed balance. In his seminal work, Maxwell [113] put forward a boundary model in which a fraction of the impinging particles is specularly reflected from the surface, while the remaining fraction is re-emitted according to a boundary equilibrium distribution f^{eq}. The latter is assumed to have a Maxwellian form where the temperature T_B of the boundary enters. The scattering kernel for this model can be written as [51, 113]

$$\mathcal{B}(\mathbf{v}' \to \mathbf{v}) = (1-\alpha)\,\delta\left(\mathbf{v}' - \mathbf{v} + 2\mathbf{n}(\mathbf{n}\cdot\mathbf{v})\right) + \frac{\alpha}{\mathcal{N}}\,|\mathbf{n}\cdot\mathbf{v}|\,f^{\text{eq}}(\mathbf{v}), \tag{5.15}$$

where the factor $|\mathbf{n}\cdot\mathbf{v}|$ is illustrated in figure 5.3, and the normalization $\mathcal{N}$ is introduced to satisfy (5.10). For simplicity, we have assumed that the boundary is at rest. The generalization to a moving boundary can be easily achieved by substituting $\mathbf{v}$ by $\mathbf{v} - \mathbf{U}_B$. The fraction α of the particles is reflected diffusively, i.e., they completely loose memory of their incoming velocity. After the collision with the surface, they have accommodated a velocity as if they were evaporated from the surface. For this reason, the coefficient α is called *accommodation coefficient*. The value $\alpha = 0$ corresponds to specular reflections (5.8). The other limiting case $\alpha = 1$ corresponds to a completely diffusive boundary condition where any memory of the state before the surface collision is lost. The scattering kernel is then independent of the incoming velocities, and from equation (5.14) we get [51, 114, 117]

$$\mathcal{B}(\mathbf{v}) \equiv \mathcal{B}(\mathbf{v}' \to \mathbf{v}) = \frac{|\mathbf{n}\cdot\mathbf{v}|\,f^{\text{eq}}(\mathbf{v})}{\int_{\mathbf{n}\cdot\mathbf{v}'<0} d\mathbf{v}'\,|\mathbf{n}\cdot\mathbf{v}'|\,f^{\text{eq}}(\mathbf{v}')}. \tag{5.16}$$

Inserting this into (5.9) yields an explicit expression for the outgoing distribution function [117]

$$f(\mathbf{r}_B, \mathbf{v}, t) = f^{\text{eq}}(\mathbf{v})\frac{\int_{\mathbf{n}\cdot\mathbf{v}'<0} d\mathbf{v}'\,|\mathbf{n}\cdot\mathbf{v}'|\,f(\mathbf{r}_B, \mathbf{v}', t)}{\int_{\mathbf{n}\cdot\mathbf{v}'<0} d\mathbf{v}'\,|\mathbf{n}\cdot\mathbf{v}'|\,f^{\text{eq}}(\mathbf{v}')}, \qquad \mathbf{n}\cdot\mathbf{v} > 0. \tag{5.17}$$

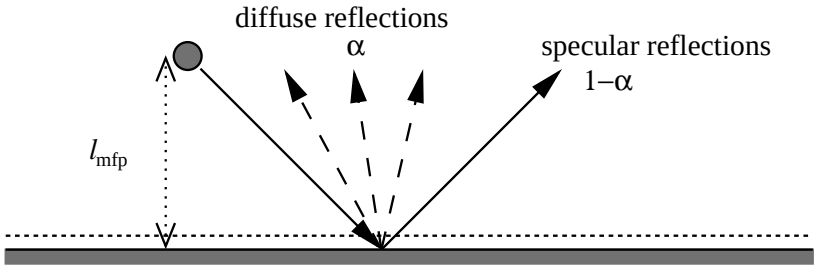


Figure 5.4: Maxwell's diffuse scattering boundary condition. A fraction α of particles is diffusely reflected from the wall and accommodated to the boundary equilibrium distribution. The remaining fraction $1-\alpha$ is specularly reflected. At a fictitious plane infinitesimally close to the surface, half of the particles impinge from a distance a mean free path away from the surface, and the other half is made up of reflected particles.

This is the so-called *diffusive boundary condition* within kinetic theory. For general values of α the reflected distribution is given by

$$f(\mathbf{r}_B,\mathbf{v},t) = (1-\alpha) f(\mathbf{r}_B,\mathbf{v}-2\mathbf{n}(\mathbf{n}\cdot\mathbf{v}),t) + \alpha f^{\text{eq}}(\mathbf{v}) \frac{\int_{\mathbf{n}\cdot\mathbf{v}'<0} d\mathbf{v}' \, |\mathbf{n}\cdot\mathbf{v}'| \, f(\mathbf{r}_B,\mathbf{v}',t)}{\int_{\mathbf{n}\cdot\mathbf{v}'<0} d\mathbf{v}' \, |\mathbf{n}\cdot\mathbf{v}'| \, f^{\text{eq}}(\mathbf{v}')}. \tag{5.18}$$

where $\mathbf{n}\cdot\mathbf{v} > 0$. Once the boundary condition for the distribution function is specified, we can proceed to evaluate the hydrodynamic flow velocity at the boundary. As shown above, the normal velocity at the boundary is zero. The flow velocity at the boundary is

$$\begin{aligned}
\mathbf{u}(\mathbf{r}_B) &= \frac{1}{\rho}\int \mathbf{v} f(\mathbf{r}_B,\mathbf{v},t)\, d\mathbf{v} \\
&= \frac{1}{\rho}\int_{\mathbf{n}\cdot\mathbf{v}\le 0} \mathbf{v} f(\mathbf{r}_B,\mathbf{v},t)\, d\mathbf{v} + \frac{1}{\rho}\int_{\mathbf{n}\cdot\mathbf{v}>0} \mathbf{v} f(\mathbf{r}_B,\mathbf{v},t)\, d\mathbf{v} \\
&= \frac{1}{\rho}\int_{\mathbf{n}\cdot\mathbf{v}\le 0} \mathbf{v} f(\mathbf{r}_B,\mathbf{v},t)\, d\mathbf{v} \\
&\quad + \frac{1-\alpha}{\rho}\int_{\mathbf{n}\cdot\mathbf{v}>0} \mathbf{v} f(\mathbf{r}_B,\mathbf{v}-2\mathbf{n}(\mathbf{n}\cdot\mathbf{v}),t)\, d\mathbf{v} \\
&\quad + \frac{\alpha}{\rho}\int_{\mathbf{n}\cdot\mathbf{v}>0} \mathbf{v} f^{\text{eq}}(\mathbf{v}) \frac{\int_{\mathbf{n}\cdot\mathbf{v}'<0} d\mathbf{v}' \, |\mathbf{n}\cdot\mathbf{v}'| \, f(\mathbf{r}_B,\mathbf{v}',t)}{\int_{\mathbf{n}\cdot\mathbf{v}'<0} d\mathbf{v}' \, |\mathbf{n}\cdot\mathbf{v}'| \, f^{\text{eq}}(\mathbf{v}')}\, d\mathbf{v}.
\end{aligned} \tag{5.19}$$

The evaluation of this expression requires explicit knowledge of the distribution function $f(\mathbf{r}_B,\mathbf{v},t)$ and, as stated above, there is an essential discontinuity at the surface of the boundary. Despite this complication, Maxwell used the bulk solution to approximate the distribution function at the surface [113]. He obtained the tangential flow velocity $u(\mathbf{r}_B) = |\mathbf{u}(\mathbf{r}_B)|$ at the surface in terms of the bulk velocity at a distance of the order of the mean free path away from the surface

$$u(\mathbf{r}_B) = \frac{1}{2}\left[u(\mathbf{r}_B + l_{\text{mfp}}\mathbf{n}) + (1-\alpha)u(\mathbf{r}_B + l_{\text{mfp}}\mathbf{n}) + \alpha U_B\right], \tag{5.20}$$

where $U_B = |\mathbf{U}_B|$ is the absolute velocity of the boundary. The formula is illustrated in figure 5.4. Conversely, the bulk velocity close to the boundary can be Taylor expanded around the surface velocity

$$u(\mathbf{r}_B + l_{\text{mfp}}\mathbf{n}) = u(\mathbf{r}_B) + l_{\text{mfp}}\mathbf{n} \cdot \left.\frac{\partial u}{\partial \mathbf{r}}\right|_{\mathbf{r}_B} + \frac{1}{2} l_{\text{mfp}}^2 \mathbf{nn} : \left.\frac{\partial^2 u}{\partial \mathbf{r}\partial \mathbf{r}}\right|_{\mathbf{r}_B} + \dots . \tag{5.21}$$

Solving these equations for $u(\mathbf{r}_B)$ yields the celebrated expression for the slip velocity as an expansion in powers of the Knudsen number [51, 118]

$$u_{\text{slip}} = u(\mathbf{r}_B) - U_B = \frac{2-\alpha}{\alpha}\left[Kn\ \mathbf{n} \cdot \left.\frac{\partial u}{\partial \hat{\mathbf{r}}}\right|_{\hat{\mathbf{r}}_B} + \frac{Kn^2}{2}\mathbf{nn} : \left.\frac{\partial^2 u}{\partial \hat{\mathbf{r}}\partial \hat{\mathbf{r}}}\right|_{\hat{\mathbf{r}}_B} + \dots \right], \tag{5.22}$$

where the spatial variables $\hat{\mathbf{r}} = \mathbf{r}/L$ have been scaled with the characteristic hydrodynamic length L. To first-order, this resembles the Navier slip boundary condition, cf. equation (5.7), where the slip length is given by

$$\delta_B = \frac{2-\alpha}{\alpha}\, l_{\text{mfp}}. \tag{5.23}$$

In the context of kinetic theory the slip length was earlier referred to as *Gleitungs coefficient* [113, 119]. Again, the slip length depends on both a bulk (l_{mfp}) and a surface (α) property.

Equation (5.22) shows that for finite Knudsen number, Maxwell's kinetic boundary condition always leads to a non-vanishing slip velocity. The slip length δ_B is on the order of the mean free path, if the accommodation coefficient α is close to unity, i.e., a purely diffusive boundary condition. Conversely, the slip length diverges in the limit $\alpha \to 0$ corresponding to the full-slip boundary condition. The hydrodynamic no-slip boundary condition is only valid in the limit of vanishing Knudsen number. The first-order approximation describes the slip flow regime up to Knudsen numbers of the order $Kn \sim 0.3$ [107]. Although beyond this limiting value the validity of the Navier-Stokes equation is in general questionable, some results have been reported where the inclusion of the second-order slip coefficient leads to reasonable improvements [see Refs. 107, 118].

Maxwell's kinetic boundary model is in agreement with a range of experimental and numerical findings [109, 120]. At the same time, it is a very simple model as it describes the surface properties by only one parameter, i.e., the accommodation coefficient α. Therefore it is a promising starting point to devise mesoscopic boundary models for use in computer simulations [110].

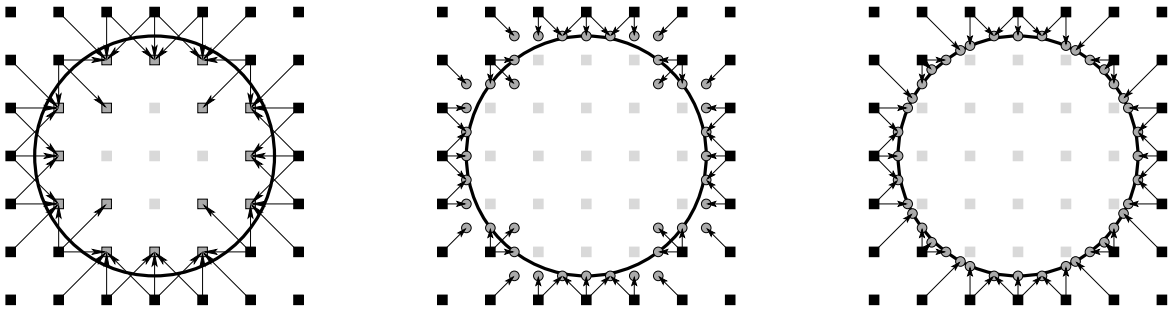


Figure 5.5: Lattice representation of a rigid object. (Left) Node-based representation. (Middle) Mid-link representation. (Right) Link representation with boundary markers.

5.2 Boundary conditions for lattice Boltzmann models

Boundary conditions for computer algorithms differ conceptually from boundary conditions in analytical theories. While in the latter, they are posed as additional equations that filter from the admissible solutions of the differential equation the unique solution to the boundary value problem, in computer simulations the boundary conditions are part of the algorithm that generates the sought-after solution as a trajectory in time. That is, the boundary conditions are part of the dynamical updating scheme and are applied in every iteration step changing the state of the system. Consequently, they have to be constructed in such a way that the theoretical boundary condition is satisfied during the course of the simulation. It can be expected that this is only possible to a certain degree of accuracy, in the same sense as the discretized dynamics can only mimic the real system approximately. As the boundary condition is part of the dynamics, it can potentially deteriorate the whole method if its accuracy is inferior compared to the bulk dynamics. Therefore, it is highly desirable to use boundary conditions that attain at least the same accuracy as the method in general.

In order to represent boundaries in the lattice Boltzmann method, one has to map the solid objects to the lattice structure. This can be done in several ways, three of which are illustrated in figure 5.5 for the example of a circle. A simple way is the node based approach where the lattice sites are divided into solid, fluid and boundary nodes. *Solid nodes* are completely covered by the solid object; *boundary nodes* have at least one velocity link to a solid node whereas *fluid nodes* are only linked to boundary nodes or other fluid nodes. An alternative way is the link based approach where boundary markers are put on the velocity links that connect solid and fluid nodes. If no further information is included, the boundary markers are simply located halfway between the solid and the fluid nodes, which yields a staggered representation of the solid object. In a more elaborate variant of the link based approach, the boundary markers are assumed to lie directly on the boundary surface. This requires some additional effort to determine the exact intersection of the velocity link and the boundary, but it has the benefit that the obtained representation of the solid object is somewhat more precise. Aside from the different positioning of the boundary nodes or markers, all the different versions have in common that some of the velocity links are cut by the surface. On these links, the population moving from the interior to the exterior of the

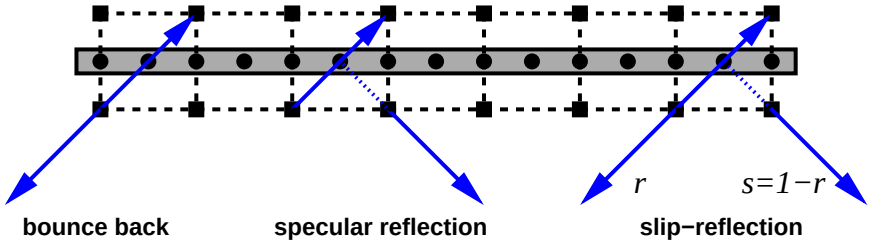


Figure 5.6: Illustration of simple mid-link reflection rules. (Left) Bounce-back reverses the velocity of the impinging population. (Middle) Specular reflections reverse only the normal momentum during reflection of the populations. (Right) Slip-reflections combine bounce-back and specular reflections.

solid object is undefined. These unknown populations have to be specified by the boundary condition. The role of a lattice Boltzmann boundary condition is thus to define a rule for the unknown populations that is compatible with the dynamics of the system and produces the desired hydrodynamic boundary condition. This can either be done by discretizing the kinetic boundary conditions discussed in the previous section, or by imposing explicit rules and verifying the macroscopic behavior. In the following paragraphs, some of the most commonly used boundary conditions for the lattice Boltzmann model are reviewed.

5.2.1 Bounce-back

The oldest but nevertheless still the most widely used boundary condition for lattice Boltzmann is the *bounce-back* rule [10]. It was already introduced in the context of lattice gas automata and is applied to obtain a hydrodynamic stick boundary condition. The rule reflects the populations at the boundary nodes by a bounce-back collision, in which an incoming population is bounced back whereupon its velocity is reversed. Depending on the lattice representation of the solid object, there are two ways to implement the bounce-back. In the node-based implementation, the reversion of the velocity takes place on the boundary nodes $\mathbf{r}_B$

$$f_i(\mathbf{r}_B + \tau\mathbf{c}_i, t + \tau) = f^*_{i^-}(\mathbf{r}_B, t), \tag{5.24}$$

where the index i^- is defined by $\mathbf{c}_{i^-} = -\mathbf{c}_i$. It can be shown theoretically [121, 122], that the node-based bounce-back rule leads to a hydrodynamic boundary that is shifted into the fluid by half a lattice spacing. Hence, the stick boundary is effectively located on the links. It has therefore become common to use a link-based formulation [65, 121, 123]

$$f_i(\mathbf{r}_B, t + \tau) = f^*_{i^-}(\mathbf{r}_B, t), \tag{5.25}$$

where $\mathbf{r}_B$ is now a fluid node linked to a boundary marker by $\mathbf{c}_{i^-}$. The link-based bounce-back rule is illustrated in figure 5.6. The stick boundary is located on the boundary markers in the middle between the nodes. In the following, we will refer to the link-based implementation.

The bounce-back rule has no direct analogon in kinetic theory, as a bounce-back process is rather improbable to occur in the microscopic dynamics. The boundary condition is constructed to enforce a vanishing flow velocity at the boundary. For simple flows, it is possible to solve the linear lattice Boltzmann equation with bounce-back boundary conditions analytically [124–126]. The result for Poiseuille flow shows that the bounce-back rule is only first order exact, i.e., the boundary location is subject to $\mathcal{O}(L^{-1})$ corrections, where L is the channel width. The flow velocity profile deviates from the exact no-slip solution by a constant offset [65, 126]

$$\Delta u(\mathbf{r}_B) = \beta u_{\max} \frac{a^2}{L^2}. \tag{5.26}$$

The correction factor β depends on the collision operator and the definition of the hydrodynamic momentum density. With the definition (4.65) the result is

$$\beta = \frac{1}{3}\left(\frac{16}{\lambda^2} - \frac{16}{\lambda} + 1\right), \tag{5.27}$$

for the BGK collision operator and

$$\beta = \frac{1}{3}\left(\frac{8}{\lambda} - 7\right) \tag{5.28}$$

for the MRT collision operator with equal eigenvalues $\lambda = \lambda_s = \lambda_b$ for the bulk and shear modes [65]. A detailed asymptotic analysis of the bounce-back rule shows further that the pressure is at most first-order accurate, even for boundaries that are aligned along a lattice direction. For inclined boundaries and arbitrary shaped objects, the order of accuracy reduces to first order for velocity and zeroth order for the pressure [97]. These deficiencies have motivated various attempts to improve the bounce-back scheme, some of which will be discussed below.

5.2.2 Specular reflections

An equally simple scheme as bounce-back is the specular reflection rule [see e.g. 121], where an incoming population is reflected from the wall such that only the normal velocity component changes sign while the tangential component is unchanged. The link-based formulation of specular reflections is given by

$$f_i(\mathbf{r}_B + \tau\left[\mathbf{c}_i - \mathbf{n}(\mathbf{n}\cdot\mathbf{c}_i)\right], t+\tau) = f^*_{i^+}(\mathbf{r}_B, t), \tag{5.29}$$

where $\mathbf{n}$ is the boundary normal and the index i^+ is defined by $\mathbf{c}_{i^+} = \mathbf{c}_i - 2\mathbf{n}(\mathbf{n}\cdot\mathbf{c}_i)$. The specular reflections are illustrated in figure 5.6. They directly correspond to their counterpart in kinetic theory, cf. equation (5.8), hence they produce a hydrodynamic full slip boundary condition. As stated above, a full slip boundary without momentum transfer at the surface is usually not desirable in realistic applications. The specular reflection rule is therefore not widely applied and is mainly used in conjunction with alternative rules. Nevertheless, it can function as a starting point to develop more sophisticated boundary schemes, see section 5.3.

5.2.3 Diffuse reflections

The bounce-back and specular reflection rules define the unknown population by a simple reflection of one single population. In general, an outgoing population can be a function of all incoming populations, and an obvious generalization of the link based reflection rules is a linear combination of the known populations [127]

$$f_i(\mathbf{r}_B, t) = \sum_{\mathbf{n}\cdot\mathbf{c}_j<0} \mathcal{B}_{ij}(\mathbf{r}_B, \mathbf{r}_B - \tau\mathsf{P}\mathbf{c}_j, t) f_j^*(\mathbf{r}_B - \tau\mathsf{P}\mathbf{c}_j, t). \tag{5.30}$$

$\mathsf{P} = \mathbb{1} - \mathbf{nn}$ is the projection operator that projects $\mathbf{c}_j$ onto the tangential subspace of the boundary, and $\mathcal{B}_{ij}$ is a scattering matrix that satisfies conservation of mass and normal momentum by the condition

$$\sum_{\mathbf{n}\cdot\mathbf{c}_i>0} \mathcal{B}_{ij}(\mathbf{r}_B, \mathbf{r}_B - \tau\mathsf{P}\mathbf{c}_j, t) = 1. \tag{5.31}$$

In the simplest case, the scattering matrix implements a combination of bounce-back and specular reflections resulting in a slip boundary condition [110, 127–129]. In fact, comparison of equations (5.30) and (5.9) shows that $\mathcal{B}_{ij}$ is nothing but a discrete version of the scattering kernel in kinetic theory. Ansumali and Karlin used this to discretize the diffusive boundary condition[2] and obtained the discrete diffusive boundary condition [117], which can be straightforwardly generalized to a discrete version of Maxwell's accommodation condition, cf. equation (5.18) and [130],

$$f_i(\mathbf{r}_B, t+\tau) = (1-\alpha) f_{i^+}^*(\mathbf{r}_B - \tau\mathsf{P}\mathbf{c}_i, t) + \alpha f_i^{\mathrm{eq}}(\rho_B, \mathbf{u}_B) \frac{\sum_{\mathbf{n}\cdot\mathbf{c}_j<0} |\mathbf{n}\cdot\mathbf{c}_j| f_j^*(\mathbf{r}_B - \tau\mathsf{P}\mathbf{c}_j, t)}{\sum_{\mathbf{n}\cdot\mathbf{c}_j<0} |\mathbf{n}\cdot\mathbf{c}_j| f_j^{\mathrm{eq}}(\rho_B, \mathbf{u}_B)}, \tag{5.32}$$

where $\mathbf{n}\cdot(\mathbf{c}_i - \mathbf{u}_B) > 0$. This rule constitutes a direct implementation of kinetic boundary conditions which is in line with the mesoscopic spirit of the lattice Boltzmann method. It can reproduce the Knudsen number dependent wall slip to very good agreement and yields the same convergence to the hydrodynamic limit as the Boltzmann equation [117]. On the other hand, (5.32) is still a reflection rule based on either a node-based or a link-based representation of the boundary surface. The ramifications for arbitrary shaped objects are to date not very well explored. Moreover, the implementation of the diffusive boundary condition [117] is more complicated compared to the simpler slip-reflection models [127, 128]. It is probably for these reasons, that the kinetic boundary condition is yet rarely used in practical applications, where the bounce back boundary condition is frequently favored for its striking simplicity. This holds in particular for arbitrary geometries and particulate suspensions [8, 131].

[2] In principle this can be done along the same lines as in the bulk. However, due to the occurrence of half-space integrals, the quadrature nodes are different, resulting in a lattice mismatch. Ansumali and Karlin therefore resort to using the bulk quadrature for the boundary nodes as well, which strictly speaking introduces additional discretization errors [117]. It is also to be noted, that in their work the equilibrium distribution of the entropic lattice Boltzmann model was used.

5.2.4 Advanced closure schemes

Apart from the diffusive reflections, many other attempts to improve the accuracy of the bounce-back boundary condition have been proposed. One class of these approaches tries to find a solution for the unknown populations in terms of the populations on adjacent lattice sites. Let us refer to this class as *closure schemes*. Their aim is to generate a set of populations that satisfies the desired boundary conditions at the hydrodynamic level, i.e., the desired velocity field (Dirichlet condition) and its gradients (Neumann condition). Ziegler [132] combined the nodal bounce-back with setting the grazing directions to the average of the incoming directions. This scheme ensures the no-slip condition by construction, but it is not mass conserving on the boundary nodes. Skordos [133] addressed the problem of inversely mapping the hydrodynamic fields to the lattice Boltzmann populations. A modified collision operator was introduced for the boundary nodes, which relaxes the populations towards an equilibrium distribution that includes velocity gradients as additional correction terms. Although a modified equilibrium distribution at the boundary is a reasonable assumption, the inclusion of gradient terms is questionable and lacks a rigorous justification in terms of the Chapman-Enskog expansion. If the velocity gradients are unknown, they must be evaluated using finite-differences. Moreover, the density was assumed to be known at the boundary nodes, which may not always be appropriate. Noble and coworkers [134, 135] developed a two-dimensional closure scheme where the density is a computed quantity and only the velocity components at the boundary have to be prescribed. The scheme is based on dividing the populations into groups that stream in from neighboring fluid nodes, boundary nodes or solid nodes, respectively. The latter of these three are the unknown quantities in an equation system which is obtained from the conservation laws for mass and momentum. Noble et al. solved this equations system for a seven velocity model, and a generalization to the three-dimensional D3Q15 and D3Q18 models was developed by Maier et al. [136]. A similar technique was used by Inamuro et al. [137, 138] and combined with the idea of diffusive scattering. The unknown populations are drawn from an equilibrium distribution for the wall, but with an additional *counterslip velocity* in the tangential direction which is adjusted to satisfy mass conservation on the wall. The Inamuro method yields an equation system that can be solved for arbitrary lattice models. On another route, Zou and He [139] proposed a closure based on the concept of *bounce-back of non-equilibrium parts*. The approach was used to derive pressure and velocity boundary conditions for the D2Q9 and D3Q15 models. However, the inherent mismatch between the number of unknown parameters and the number of constraint equations was only heuristically solved, but no systematic procedure to cope with this problem was devised [5]. Lätt and coworkers [101, 140] have recently applied bounce-back of non-equilibrium parts together with a so-called regularized LBGK model. This approach turns out to be a mixture of the Inamuro and the Zou/He methods, while the Dirichlet condition is ensured by implicitly making use of the moment representation familiar in MRT models.

A more systematic approach was put forward by Ginzbourg and d'Humières [141]. The basic idea is to exploit the Chapman-Enskog result for the populations to compute the necessary derivatives of the velocity field locally at the boundary node without using finite

differences. With the help of the expression (4.42) it is possible to write the unknown outgoing population numbers as linear combinations of the known populations and the prescribed boundary constraints. While it is not obvious that a unique solution for this linear equation system exists in general, an algorithm for a flat wall aligned to a lattice direction was presented in [141]. For simple flows, it yields a local second order boundary condition at the expense, however, of a second-order mass flux across the boundary. More recently, Halliday et al. [142] have revisited the method of Ginzbourg and d'Humiéres and presented an enhanced unified framework to tackle the boundary closure problem in a systematic way. Since most of the above closure schemes can be subsumed under this framework, it deserves to be outlined in a little more detail [see also 101]. For simplicity, we restrict the formal presentation to lattice models with $c_{i\alpha} \in \{-1, 0, +1\}$.[3] If the wall normal is assumed to point in the positive z-direction, the populations can be divided into impinging, grazing and reflected directions according to the z-component of the velocity vector.

Closure for equilibrium parts

The mass density at the boundary node can then be written as

$$\rho = \sum_i f_i = \sum_{c_{iz}<0} f_i + \sum_{c_{iz}=0} f_i + \sum_{c_{iz}>0} f_i. \tag{5.33}$$

Similarly, the normal boundary velocity can be written as

$$\rho u_{B,z} = \sum_i f_i c_{iz} = \sum_{c_{iz}<0} f_i c_{iz} + \sum_{c_{iz}>0} f_i c_{iz} = -\sum_{c_{iz}<0} f_i + \sum_{c_{iz}>0} f_i. \tag{5.34}$$

Eliminating the unknown populations yields an expression for the density in terms of the known incoming populations

$$\rho = \frac{1}{1 - u_{B,z}} \left(2 \sum_{c_{iz}<0} f_i + \sum_{c_{iz}=0} f_i \right). \tag{5.35}$$

A similar expression was used in the works [101, 134, 137, 139]. Having determined the density, it can be used together with the prescribed boundary velocity to compute the equilibrium part f_i^{eq} of the boundary populations using the standard equilibrium distribution (2.56). By this procedure, any Dirichlet condition on the hydrodynamic fields can be satisfied.

Closure for non-equilibrium parts

It remains to determine the non-equilibrium parts of the populations such that the correct velocity gradients at the boundary are recovered. Using the Chapman-Enskog result (4.42),

[3] Halliday et al. [142] have treated only the D2Q9 explicitly. The presentation here is a straightforward generalization.

we are left with the following equation system

$$\begin{aligned}
0 &= \sum_{c_{iz}>0} f_i^{\text{neq}} + \sum_{c_{iz}=0} f_i^{\text{neq}} + \sum_{c_{iz}<0} f_i^{\text{neq}}, \\
0 &= \sum_{c_{iz}>0} f_i^{\text{neq}} c_{i\alpha} + \sum_{c_{iz}=0} f_i^{\text{neq}} c_{i\alpha} + \sum_{c_{iz}<0} f_i^{\text{neq}} c_{i\alpha}, \\
\frac{\rho\tau c_s^2}{\lambda}\left(\frac{\partial u_\alpha}{\partial r_\beta} + \frac{\partial u_\beta}{\partial r_\alpha}\right) &= \sum_{c_{iz}>0} f_i^{\text{neq}} c_{i\alpha} c_{i\beta} + \sum_{c_{iz}=0} f_i^{\text{neq}} c_{i\alpha} c_{i\beta} + \sum_{c_{iz}<0} f_i^{\text{neq}} c_{i\alpha} c_{i\beta}.
\end{aligned} \tag{5.36}$$

This system is in principle under-determined, such that there is no general procedure to determine the solution f_i^{neq}. This is due to the additional degrees of freedom in the lattice Boltzmann method compared to the number of hydrodynamic variables. A solution of (5.36) can be obtained by choosing a set of the incoming populations to take their known values after streaming. The choice of this set should include a maximum number of populations that stream from the bulk to the boundary in order to facilitate the coupling of the bulk and the surface. On the other hand, the explicit choice must guarantee the solvability of the equation system. In [142], this was achieved by computing the determinant for all possible combinations and enumerating all forbidden combinations. Another difficulty arises in measuring the velocity gradients at the boundary. Ginzbourg and d'Humières [141] considered the velocity derivatives as additional unknowns of the closure system, which makes it even more complicated to solve because more known populations are required. Such an approach is thus infeasible in complex geometries. Therefore, Halliday et al. [142] chose to use finite differences to determine the velocity derivatives, like Skordos [133]. In [137] the gradients are not necessary for the equilibrium forcing, whereas in [101, 139] they are fixed implicitly by the bounce-back of non-equilibrium parts. In the works [132, 134–136] the gradients were not accounted for explicitly, which is the reason why the respective schemes are only first order for arbitrary geometries. The different closure schemes show that there is a trade-off between accuracy and locality because second-order closures require the velocity derivatives whose local computation is only possible in simple geometries. Second-order accuracy is in general difficult to achieve locally, a fact which has led to the development of non-local inter- and extrapolation schemes. A comprehensive comparison of several of the above boundary schemes for straight walls aligned to the lattice can be found in Latt et al. [101].

5.2.5 Interpolation and extrapolation schemes

One of the first extrapolation schemes was introduced by Chen et al. [143]. They introduced an additional layer inside the solid and extrapolated the populations at those nodes from the boundary nodes and the first fluid node. After the extrapolation, an equilibrium forcing is applied at the surface similar to the Inamuro method. The location of the wall is, however, only first order in the lattice spacing. This is because the simple node or link-based schemes lead to a staggered representation of curved boundaries. To treat the boundary with higher accuracy, it is necessary to use boundary markers that specify the exact cutting points with

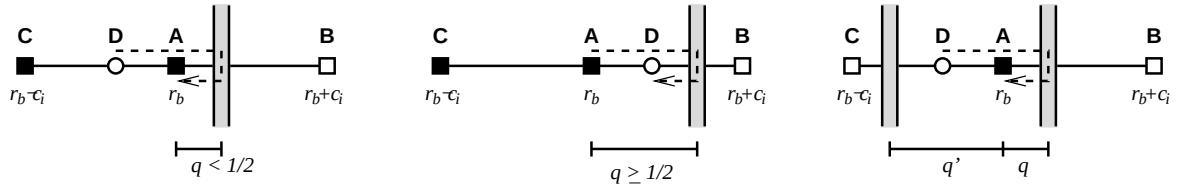


Figure 5.7: Illustration of the interpolation rules used in the boundary condition by Bouzidi et al. [154]. Depending on the value of q, either a pre-collision or a post-collision population is interpolated. The rightmost picture shows the case where only one lattice node is present between two surfaces. While Bouzidi's scheme is not applicable any more, equilibrium interpolation is still possible, cf. section 5.2.6.

the lattice vectors, i.e., for every cut link the fraction which lies outside the solid needs to be known. The application of this representation was pioneered by Filippova and Hänel [144–146] in their boundary-fitting scheme. The unknown reflected populations are computed as a linear combination of the incoming population and a fictitious boundary equilibrium, where the velocity inside the solid is obtained by linear extrapolation from the last fluid node and the prescribed boundary velocity. The weighting coefficients of the linear combination are functions of the respective link's cut ratio and can be determined by a Chapman-Enskog expansion [145]. However, if the link fraction outside the solid becomes small, the method shows strong instabilities for pressure driven channel flows [147, 148]. Mei et al. [147] reexamined the boundary fitting scheme and proposed an improved version by refining the extrapolation for the velocity used in the boundary equilibrium. It was numerically shown that the stability is improved considerably [147, 149].

A different scheme was put forward by Verberg and Ladd [150, 151] under the name *continuous bounce back*. It is related to the volumetric formulations of the lattice Boltzmann method [62, 152, 153] and uses special interpolated bounce-back rules for lattice cells that are partially occupied by solid objects. The continuous bounce back rules for general geometries are however quite complicated. Furthermore, it was found that they lead to impaired stability below a critical shear viscosity [151]. On the other hand, the accuracy of the method is not affected by the shape or position of the fluid-solid interface with respect to the lattice since only the fraction of fluid per node is needed in the algorithm.

In essence, the method of Verberg and Ladd is a special interpolation scheme. A simpler, physically intuitive interpolation scheme has been introduced by Bouzidi et al. [154]. An essential feature of the approach is that only populations along one direction are used for the interpolation. Hence, it is enough to consider the one-dimensional situations depicted in figure 5.7. We seek an interpolation scheme for the reflected population at the fluid node A next to the surface. If we imagine that this population was bounced-back by the surface, it would fictitiously originate at the location D. Let the fraction of the link outside the solid be denoted by q. Then two cases have to be distinguished: If $q < 1/2$, the fictitious population at D can be obtained by interpolation from A and C. In the other case $q \geq 1/2$, the pre-streaming population at D can only be obtained extrapolation, which is inferior in terms of stability and should be avoided. To this end, the fictitious post-streaming population at D

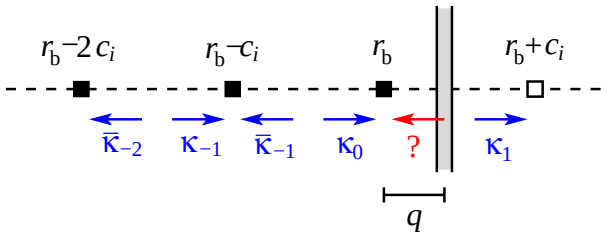


Figure 5.8: Illustration of the multireflection boundary condition of Ginzburg and d'Humières [64]. The five blue populations are weighted with coefficients κ_l to calculate the unknown red population.

can be used to interpolate the sought reflected population at A by interpolation from C and D. This leads to the following linear interpolation scheme for the unknown population at A

$$\begin{aligned} f_{i^-}(\mathbf{r}_B, t+\tau) &= 2q f_i^*(\mathbf{r}_B, t) + (1-2q) f_i^*(\mathbf{r}_B - \tau \mathbf{c}_i, t), & q < \frac{1}{2}, \\ f_{i^-}(\mathbf{r}_B, t+\tau) &= \frac{1}{2q} f_i^*(\mathbf{r}_B, t) + \frac{2q-1}{2q} f_{i^-}^*(\mathbf{r}_B, t), & q \geq \frac{1}{2}. \end{aligned} \tag{5.37}$$

These expressions are continuous in q and for $q = 1/2$ they reduce to the standard bounce-back rule. The scheme can straightforwardly be extended to quadratic interpolation and is applicable to moving boundaries as well [154, 155]. Rohde et al. [156] have applied the interpolation rules to the volumetric schemes and carried out a theoretical analysis for plane Poiseuille flow. They find that the Bouzidi scheme is subject to errors in mass conservation, whereas the volumetric interpolation schemes are mass conservative. For inclined boundaries, however, the methods are still first-order accurate.

Furthermore, the effective location of the boundary depends on the viscosity, and for example in Poiseuille flows, exact parabolic profiles are not obtained for arbitrary inclined channels. This was pointed out by Ginzburg and d'Humières [64] in a seminal paper, in which they present a comprehensive analysis of the accuracy of interpolation boundary conditions. For this purpose, they introduce the *multireflection boundary condition* which subsumes bounce-back and the linear and quadratic interpolation schemes. The multireflection rule uses three fluid nodes and five populations along a lattice direction. The set-up is depicted in figure 5.8. The weighting factors κ_l for the interpolation are derived by matching the second-order Chapman-Enskog result with a Taylor expansion at the boundary. The multireflection rule for general flows is [64, 131]

$$\begin{aligned} f_{i^-}(\mathbf{r}_B, t+\tau) = f_i^*(\mathbf{r}_B, t) - \frac{1-2q-2q^2}{(1+q)^2} f_{i^-}^*(\mathbf{r}_B, t) + \frac{1-2q-2q^2}{(1+q)^2} f_i^*(\mathbf{r} - \tau \mathbf{c}_i, t) \\ - \frac{q^2}{(1+q)^2} f_{i^-}^*(\mathbf{r} - \tau \mathbf{c}_i, t) + \frac{q^2}{(1+q)^2} f_i^*(\mathbf{r} - 2\tau \mathbf{c}_i, t). \end{aligned} \tag{5.38}$$

It constitutes a third order kinetic accurate boundary scheme and is therewith the most accurate boundary condition available. It was also shown in [64] that the standard bounce-back

can be tuned to yield second order accuracy by setting the collision eigenvalues for the kinetic (ghost) modes to

$$\lambda_g(\lambda) = -8\frac{2+\lambda}{8+\lambda}, \tag{5.39}$$

where $\lambda = \lambda_s = \lambda_b$. This choice compensates the viscosity dependent corrections to the location of the boundary surface.

5.2.6 Equilibrium interpolation

Like all interpolation methods, the multireflection boundary condition has the drawback that information from several fluid nodes is needed to determine the unknown populations. Besides rendering the scheme non-local this is problematic in applications where solid objects are close together and the distance is on the order of the lattice spacing. Then there may occasionally not be enough fluid nodes available in between the solid objects, such that the interpolation schemes break down. To overcome this drawback, Chun and Ladd [131] have very recently proposed to interpolate only the equilibrium distribution. This idea is justified by the fact that the non-equilibrium distribution enters the Chapman-Enskog expansion an order later than the equilibrium distribution. Thus a boundary condition which is second-order for the equilibrium distribution, and only first order for the non-equilibrium distribution, will still be sufficient to guarantee overall second-order accuracy. Chun and Ladd suggest to use the Bouzidi method for the equilibrium distribution and simple bounce-back for the non-equilibrium part. The *equilibrium interpolation boundary condition* is given by

$$\begin{aligned}
f_{i^-}^{\mathrm{eq}}(\mathbf{r}_B, t+\tau) &= 2q f_i^{\mathrm{eq}}(\mathbf{r}_B, t) + (1-2q) f_i^{\mathrm{eq}}(\mathbf{r}_B - \tau\mathbf{c}_i, t) && q < \frac{1}{2}, \\
f_{i^-}^{\mathrm{eq}}(\mathbf{r}_B, t+\tau) &= \frac{1-q}{q} f_i^{\mathrm{eq}}(\mathbf{r}, t) + \frac{2q-1}{q} f_i^{\mathrm{eq}}(\mathbf{r}_B + q\tau\mathbf{c}_i) && q \geq \frac{1}{2}, \\
f_{i^-}^{\mathrm{neq}}(\mathbf{r}_B, t+\tau) &= f_i^{\mathrm{neq}}(\mathbf{r}_B, t), &&
\end{aligned} \tag{5.40}$$

where $f_i^{\mathrm{eq}}(\mathbf{r}_B + q\tau\mathbf{c}_i)$ is the boundary equilibrium. Equation (5.40) still requires two fluid nodes in between solid objects. If the objects are very close, the equilibrium distribution on the other surface is used and (5.40) is replaced by

$$f_{i^-}^{\mathrm{eq}}(\mathbf{r}_B, t+\tau) = \frac{q'+2q-1}{q'} f_i^{\mathrm{eq}}(\mathbf{r}_B, t) + \frac{1-2q}{q'} f_i^{\mathrm{eq}}(\mathbf{r}_B - q'\tau\mathbf{c}_i). \tag{5.41}$$

where q' is the fraction of $\hat{\mathbf{c}}_{i^-}$ outside the second surface, cf. figure 5.7. The equilibrium interpolation rule is second-order accurate and requires only one fluid node in between the boundaries. With the choice (5.39) for the collision eigenvalues of the kinetic modes, the location of the boundary is independent of viscosity. It was shown numerically that the equilibrium interpolation boundary condition is more accurate than the linear and quadratic interpolation rules [131]. Equilibrium interpolation is probably the best compromise between accuracy and simplicity of implementation among all boundary conditions presented so far.

5.2.7 Critical discussion of the existing boundary conditions

The body of works on boundary conditions is considerable and quite some progress has been made in recent years. The main findings can be summarized as follows: Simple reflection rules like bounce-back in either node-based or link-based formulations are only first-order accurate with respect to general flows. They are affected by corrections to the flow velocity at the surface and a viscosity dependent location of the hydrodynamic boundary with respect to the underlying lattice. Advanced closure schemes aiming at second-order accuracy can improve matters, but they tend to be complicated to implement and locality of the scheme has to be sacrificed when velocity gradients are required. Furthermore, the improved accuracy is often annuled by inclined or curved boundaries. Arbitrary shaped objects can be handled with inter- and extrapolation schemes. Since extrapolation is connected to deteriorated stability, interpolation is generally preferable. Interpolation rules are easier to implement than closure schemes and can systematically be tuned to the desired accuracy. However, interpolation schemes are inherently non-local and rely on a minimum number of nodes available between solid objects. *In general, there seems to be a trade-off between accuracy and locality of the boundary condition.* In addition, some of the methods are faced with the problem that the local mass conservation constraint is violated.

Another point has to be made with respect to the role of the higher moments of the distribution function. The Chapman-Enskog analysis of the multireflection boundary condition shows that the collisional eigenvalue of the kinetic modes affects the location of the wall. In other words: *Boundary conditions are affected by the dynamics of the kinetic modes of the lattice Boltzmann model.* This suggests that the more flexible MRT collision operator should be favored over the BGK collision operator, which does not allow to tune the relaxation rates of the modes separately.

Kinetic type boundary conditions such as diffusive reflections have received broader attention only recently. A conceptual problem is that the systematic discretization in terms of quadratures leads to a mismatch of nodes and incompatible lattice structures at the boundary (cf. footnote 2 on page 71). So far, this could only be resolved by abandoning the systematic expansion and accepting additional discretization errors. *A fully consistent adoption of kinetic type boundary conditions to the discrete Boltzmann lattice is to date not available.*

The latter point gives evidence that the very nature of lattice Boltzmann boundary conditions is still not understood completely in regard to the mesoscopic origin of the method. This is of particular relevance in view of an upcoming trend to use the lattice Boltzmann method for complex flows beyond the Navier-Stokes equation [46–48, 157]. Much interest is attracted to simulations of microflows at non-vanishing Knudsen number, especially in the slip flow regime. Most of the boundary conditions described above focus on realizing a stick-boundary condition for the Navier-Stokes equation. These methods are clearly not capable of reproducing slip-flows in microchannels. The apparent slip velocity according to (5.26) is merely a numerical artifact [158]. Kinetic type boundary conditions can reproduce the apparent slip-effects in microflows, but as pointed out, their discrete counterparts are affected by discretization errors such that the accuracy of the results is difficult to assess.

It is therefore fair to claim that boundary conditions for simulations in the slip-flow regime need further investigation. The following specific issues are raised:

- The hydrodynamic stick boundary condition is replaced by the Navier slip condition (5.7) where the slip length δ_B enters. The latter is a function of the boundary coefficient γ that models the properties of the boundary. A lattice Boltzmann boundary condition for slip-flow needs to be tunable by an analogous parameter. There is yet no consensus about the best way of implementing a partial slip boundary condition. It is desirable to have a theoretical relation for the slip length and the model parameter and, optimally, an estimate for the numerical errors. These are basic tools needed to compare results with analytical theory, experiments and other simulation methods.
- As the Knudsen number becomes finite, the kinetic nature of the fluid becomes more pronounced and effects beyond Navier-Stokes behavior occur, for example, the appearance of Knudsen layers. The lattice Boltzmann method is in principle a valid tool to simulate such effects because it constitutes an approximation to the Boltzmann equation. This kinetic nature has to be reflected by the boundary condition as well. This brings up two points: firstly, the influence of the higher order moments to the dynamics can eventually not be neglected any more; secondly, two kinds of collision processes have to be taken into account at the boundary, namely interparticle collisions and collisions with the boundary. Both of these points have to be tackled by an appropriate choice of the collision operator at the boundary.
- Another aspect with respect to the lattice Boltzmann approximation of kinetic theory is the impact of discretization errors due to the underlying lattice structure. In the bulk, the systematic discretization up to a given degree naturally leads to a quadrature that implies related symmetries of the emanating lattice. As a consequence, isotropy of tensors up to the rank of the quadrature is automatically satisfied. This situation changes at the boundary where the bulk symmetry is broken. To the best of the author's knowledge, systematic half-range quadratures have not been applied and the effects of the broken symmetry at the boundary have not been treated systematically in any available work on lattice Boltzmann boundary conditions.

The aforementioned issues shall be addressed in the remainder of this work. In the next subsection, a novel way of imposing a partial slip boundary condition in the lattice Boltzmann model is developed. In chapter 6, an attempt is made to devise a conceptually new method for lattice Boltzmann boundary conditions, which is completely local and takes the reduced symmetry at the boundary into account in a systematic and consistent fashion.

5.3 Partial slip boundary conditions

In microfluidic devices, such as those shown in figure 5.1, the surface-to-volume ratio is high. Another way of saying this is that the dimensions of these devices are small, and so is the typical length scale of the fluid flow. With decreasing size of the devices the Knudsen number increases and the flow in typical microchips reaches the slip-flow regime. Experimental studies show that in this regime, the boundary condition is affected by an interplay of a variety of physico-chemical parameters, such as surface charge, hydrophobicity and wetting, surface roughness etc. The most relevant phenomenon is the appearance of an effective slippage of the fluid on the boundary and reduced hydrodynamic stresses, which can lead to a significant enhancement of the flow throughput in microchannels [see Ref. 109 for an overview of slip phenomena in experiments]. Clearly, the no-slip boundary condition is inappropriate in this situation and a more microscopic approach is needed to model details of the fluid-surface interactions. While molecular dynamics and direct simulation Monte Carlo can shed some light on specific aspects of the fluid-surface interactions, such as rarefaction or dewetting [159], they are computationally too expensive to simulate complex flows on reasonable time and length scales. The lattice Boltzmann method is much better suited for simulating flows and, due to its kinetic origin, it is practically well suited to handle the properties of the slip-flow regime [103, 104]. Sbragaglia and Succi [46] have recently argued that the lattice Boltzmann approximation indeed remains valid up to $Kn \sim \mathcal{O}(1)$. An essential requirement is yet an appropriate boundary condition that can model the appearance of partial slippage on a mesoscopic level.

In the following, we will develop a method to implement partial slip in the lattice Boltzmann method. It aims at capturing slip at a coarse-grained level, where the details of the fluid-surface interaction are modeled by a single parameter. This is in contrast to other works, where the fluid-surface interaction is modeled as an explicit potential within a multi-phase LB model [7, 160–162]. In the same spirit, we do not incorporate the roughness of the surface explicitly, as it was for example done by Kunert and Harting [163, 164]. Furthermore, we seek a general coarse-grained model for partial slip which is not specific to the lattice Boltzmann method but can be used in other simulation methods as well, e.g., dissipative particle dynamics. In fact, a collaborating group has recently succeeded in implementing tunable-slip boundaries in DPD in an analogous fashion [165]. This opens the possibility to compare simulation results from different methods, which allows to differentiate the errors that stem from the model from those that arise as artifacts of the specific implementation.

5.3.1 Modeling wall friction

Our starting point is the constitutive equation (5.5) which leads to the Navier slip boundary condition. Let us consider the case where the boundary normal is in z-direction and a flow in the x-direction. Then the slip boundary condition says that the stress exerted on the boundary surface is proportional to the flow velocity at the boundary [111]

$$\sigma_{xz} = \gamma\, u_{\text{slip}}. \tag{5.42}$$

This suggests that we can model the fluid-solid interactions as an effective friction force, which includes all the microscopic details that lead to dissipation of energy at the surface and thereby decelerate the flow

$$F(\mathbf{r}_B) = -\zeta u_x(\mathbf{r}_B). \tag{5.43}$$

The idea is then to combine specular reflections at the boundary with the friction force, where the friction coefficient ζ is meant to control the amount of slippage over the surface. Such an approach has several advantages: first, it is very general and can be applied to off-lattice particle-based methods as well. Second, the friction force is similar to Langevin like forces, which are well established in computer simulations and provide a framework to ensure correct thermodynamics by obeying the fluctuation dissipation relation, that is, a link to statistical mechanics is readily available for our boundary model. A third point to mention is that the friction force is local and hence the validity of local conservation laws can be ensured. Finally, the implementation of forces in the lattice Boltzmann model is possible without major complications.

5.3.2 Analytical solution of the wall friction model for Poiseuille flow

Before we turn to the implementation of the wall friction model in the lattice Boltzmann method, we discuss the analytical solution in the case of a plane Poiseuille flow. This serves as a further justification and is used to compare the simulation results below.

Let us consider a stationary flow of an incompressible fluid driven by a volumetric force f_{ext} in the x-direction and confined between two plane walls whose normals point in the z-direction. It can be described by the Stokes equation

$$\eta_s \frac{\partial^2 u_x}{\partial z^2} = -f_{\text{ext}}. \tag{5.44}$$

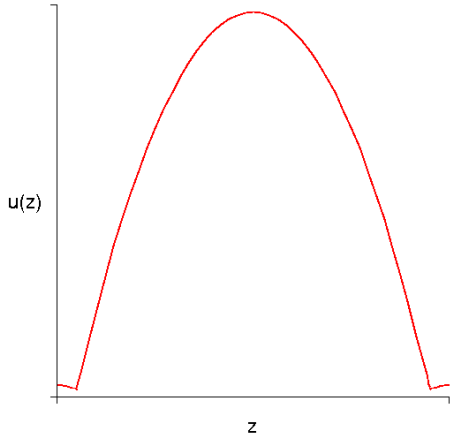
Since we have built the boundary condition on a constitutive equation for the friction force, there are additional terms that enter the bulk equation. The continuum transcription of the friction force on the last layer of fluid nodes comes in terms of δ-distributions. Let us assume that the walls are located at $z = 0$ and $z = L$, then the friction force acts on the layers $z = z_B$ and $z = L - z_B$, and we arrive at the differential equation

$$\eta_s \frac{\partial^2 u_x}{\partial z^2} = -f_{\text{ext}} + \frac{\zeta}{a^2} u(z)\delta(z - z_B) + \frac{\zeta}{a^2} u(z)\delta(z - L + z_B). \tag{5.45}$$

Because of the additional specular reflections, this equation has to be solved with full-slip boundary conditions:

$$u'(0) \equiv \frac{\partial u_x(0)}{\partial z} = 0, \qquad u'(L) \equiv \frac{\partial u_x(L)}{\partial z} = 0. \tag{5.46}$$

The solution of the second order differential equation is piecewise parabolic, as qualitatively shown in figure 5.9,



Integration constants	
A_1	0
A_2	$\frac{f_{\text{ext}}La^2}{2\zeta} + \frac{f_{\text{ext}}z_B^2}{2\eta_s}$
B_1	$\frac{f_{\text{ext}}L}{2\eta_s}$
B_2	$\frac{f_{\text{ext}}La^2}{2\zeta} + \frac{f_{\text{ext}}z_B^2}{2\eta_s} - \frac{fLz_B}{2\eta_s}$
C_1	$\frac{f_{\text{ext}}L}{\eta_s}$
C_2	$\frac{f_{\text{ext}}La^2}{2\zeta} + \frac{f_{\text{ext}}z_B^2}{2\eta_s}$

Figure 5.9: Analytical solution of the partial slip boundary condition. The flow profile is piecewise parabolic. The table lists the integration constants appearing in (5.47).

$$\begin{aligned} u_x(z) &= -\frac{f_{\text{ext}}}{2\eta_s}z^2 + A_1 z + A_2, && z < z_B, \\ u_x(z) &= -\frac{f_{\text{ext}}}{2\eta_s}z^2 + B_1 z + B_2, && z_B \leq z \leq L - z_B, \\ u_x(z) &= -\frac{f_{\text{ext}}}{2\eta_s}z^2 + C_1 z + C_2, && z > L - z_B. \end{aligned} \tag{5.47}$$

From the boundary conditions (5.46) we get $A_1 = 0$ and $C_1 = \frac{f_{\text{ext}}L}{\eta_s}$. From symmetry around the centerline of the channel we get $B_1 = \frac{f_{\text{ext}}L}{2\eta_s}$.

The δ-distributions imply a jump in the first derivatives of the velocity. The height of this jump can be determined by formally integrating over an ϵ-interval centered around the support of the distribution. For the first δ-distribution in (5.45), we get

$$\lim_{\epsilon \to 0} [u_x'(z_B + \epsilon) - u_x'(z_B - \epsilon)] = \frac{\zeta}{\eta_s a^2} u_x(z_B). \tag{5.48}$$

The second δ-distribution can be treated analogously. Plugging in the results obtained so far we arrive at

$$u_x(z_B) = \frac{f_{\text{ext}}La^2}{2\zeta} = u_x(L - z_B). \tag{5.49}$$

This is just another way of expressing that the center piece of the solution satisfies a slip boundary condition with a slip velocity

$$u_{\text{slip}} = \frac{f_{\text{ext}}La^2}{2\zeta} \tag{5.50}$$

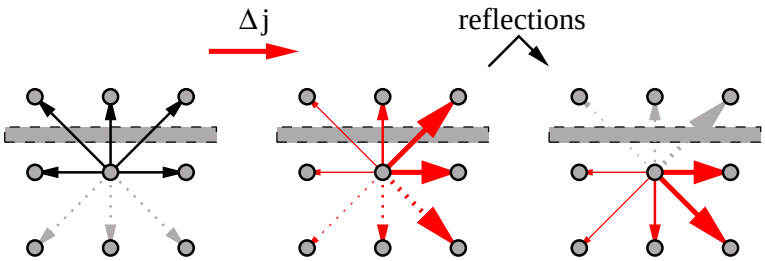


Figure 5.10: Canonical implementation of the wall friction force. Each population has a contribution from the momentum transfer Δj. This implementation leads to artifacts in the velocity profiles measured in simulations.

at the hydrodynamic wall locations $z = z_B$ and $z = L - z_B$. In the limit $z_B \to 0$ we get the velocity profile

$$\begin{aligned} u_x(z) &= -\frac{f_{\text{ext}}}{2\eta_s} z^2 + \frac{f_{\text{ext}} L}{2\eta_s} z + \frac{f_{\text{ext}} L a^2}{2\zeta} \\ &= -\frac{f_{\text{ext}}}{2\eta_s} \left(z - \frac{L}{2} \right)^2 + \frac{f_{\text{ext}} L^2}{8\eta_s} + \frac{f_{\text{ext}} L a^2}{2\zeta}, \end{aligned} \tag{5.51}$$

which is nothing but a Poiseuille profile shifted by u_{slip}. The tangential stress at the wall is

$$\eta_s \frac{\partial u_x(0)}{\partial z} = \frac{f_{\text{ext}} L}{2}, \tag{5.52}$$

and the slip length is

$$\delta_B = \frac{\eta_s}{\zeta} a^2. \tag{5.53}$$

The latter relation shows that the force friction parameter ζ is related to the coefficient γ in the hydrodynamic constitutive equation (5.42).

5.3.3 Implementation of wall friction: "canonical" method

Force implementations in the lattice Boltzmann model have been discussed by several authors [47, 85, 86, 166–169]. The most stringent way is to discretize the forcing term in the continuous Boltzmann equation in terms of a truncated Hermite expansion [47, 166]. The obtained result is to second order equivalent to the forcing terms derived within the Chapman-Enskog expansion, cf. chapter 4 and [8, 65]. Here, we accordingly modify the force coupling method of Ahlrichs and Dünweg [85, 86], which was originally developed to couple polymers to the lattice Boltzmann fluid. The modification consists of adding the second-order correction term derived within the Chapman-Enskog expansion. The forcing term is given by

$$\Delta_i^g = \frac{w_i \tau}{c_s^2} \left[\mathbf{g} \cdot \mathbf{c}_i + \frac{1}{2c_s^2} \mathsf{G} : \left(\mathbf{c}_i \mathbf{c}_i - c_s^2 \mathbb{1} \right) \right], \tag{5.54}$$

where the tensor G is given by equation (4.67). The volumetric force is obtained by the above friction force

$$\mathbf{g}(\mathbf{r}_B) = -\frac{\zeta}{a^3}\frac{\mathbf{j}(\mathbf{r}_B)}{\rho(\mathbf{r}_B)}. \tag{5.55}$$

In principle, the modified definition (4.65) for the hydrodynamic momentum density has to be used. This brings up an additional complication because the redefined momentum contains the force and at the same time the force is a function of the momentum. Strictly speaking, the equations (4.65) and (5.55) have to be solved self-consistently which would require an iterative scheme. However, if the velocity at the boundary is small[4], the corrections are negligible. The localized force density is applied on the last layer of fluid nodes in front of the boundary. In addition, the specular reflections are applied on the boundary links. The complete scheme is illustrated in figure 5.10. We call this scheme the *"canonical" implementation of the friction force* because it uses the forcing term without any further adjustments.

5.3.4 Simulation results for the "canonical" implementation

To validate the new partial slip scheme, we have simulated a Poiseuille flow through a plane channel. The channel is aligned with the lattice such that the wall normals point in the positive or negative z-direction, respectively. The simulation box has a width of 20 lattice spacings in the z-direction, where the walls are located at the first and the last lattice layer. Taking into account the shift of $a/2$ for the hydrodynamic boundaries, the channel has an effective width of $L_{\text{eff}} = 19a$. Periodic boundary conditions are applied in the x- and y-directions. The fluid is driven by a volumetric force f_{ext} in the x-direction. The density of the fluid is $\rho = 1.0$ and the kinematic viscosity is set to $\nu = 3.0$, both measured in lattice units. In figure 5.3.4, the velocity profiles are shown for a lattice spacing of $a = 1.0$ and a driving volumetric force of $f_{\text{ext}} = 0.01$. The value of the friction coefficient ζ varied from 1.0 to 5.0. The analytical solutions, cf. (5.47), for the respective parameter sets are plotted as dotted lines. The velocities on the vertical axis are scaled by the theoretical maximum velocity for $\zeta = 1.0$.

The measured velocity profiles have a parabolic shape in the bulk, as expected. However, they deviate from the analytical prediction significantly and close to the boundary the profiles are distorted due to a kink on the next nearest fluid node. Nevertheless, the velocity on the last fluid node visibly matches the theoretical value. This indicates that, while the friction induced deceleration of the fluid at the boundary is captured correctly, there is an undesirable jump in the tangential stress at the next nearest layer. This has to be interpreted as an artifact in the simulation. As the most likely source of such artifacts is the discreteness of the lattice structure, the effect of reducing the lattice spacing a was investigated. Figure 5.12 shows the results for a lattice spacing of $a = 0.1$ and an effective channel width of

[4] In the simulations that were performed, the fluid is initially at rest. Therefore the slip velocity in the stationary state is approached from below. The friction force increases only up to the value where, together with the viscous stresses in the fluid, the external driving force is balanced.

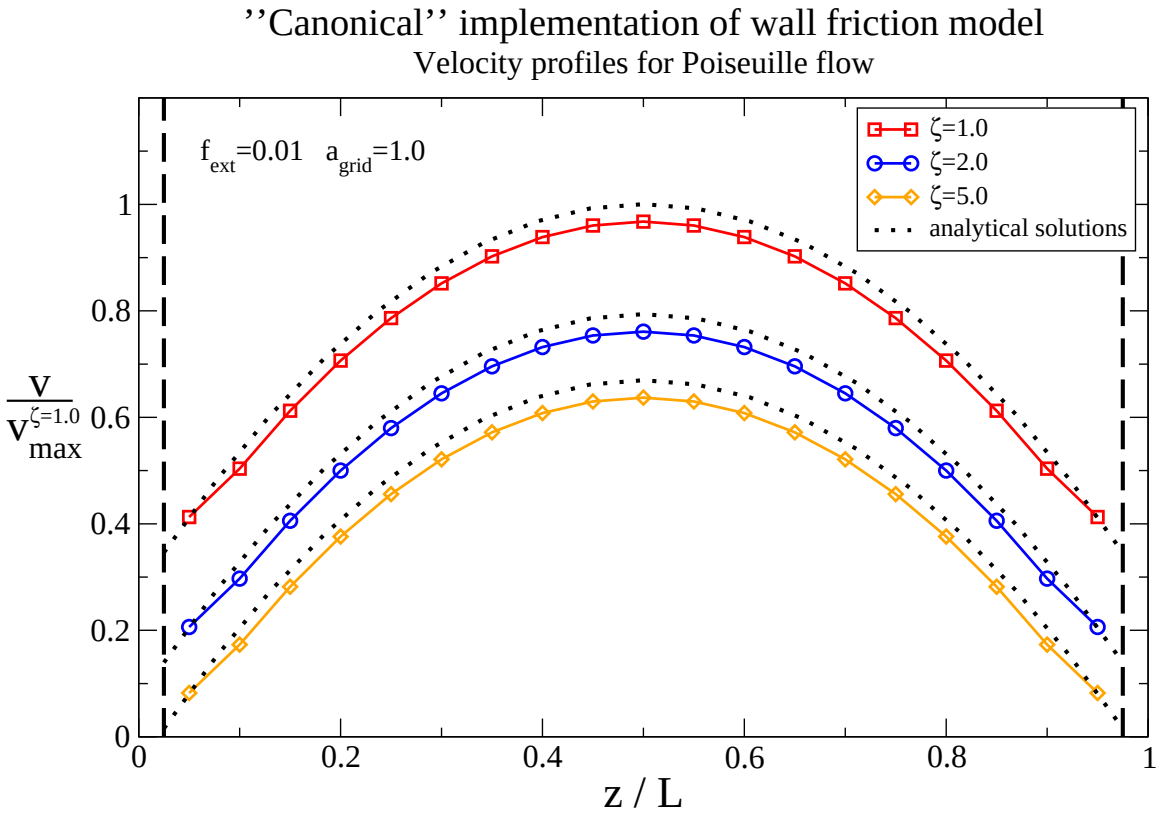


Figure 5.11: Simulation results for the wall friction model with the "canonical" implementation of the friction force. The profiles deviate from the analytical solution due to the occurrence of a kink next to the wall.

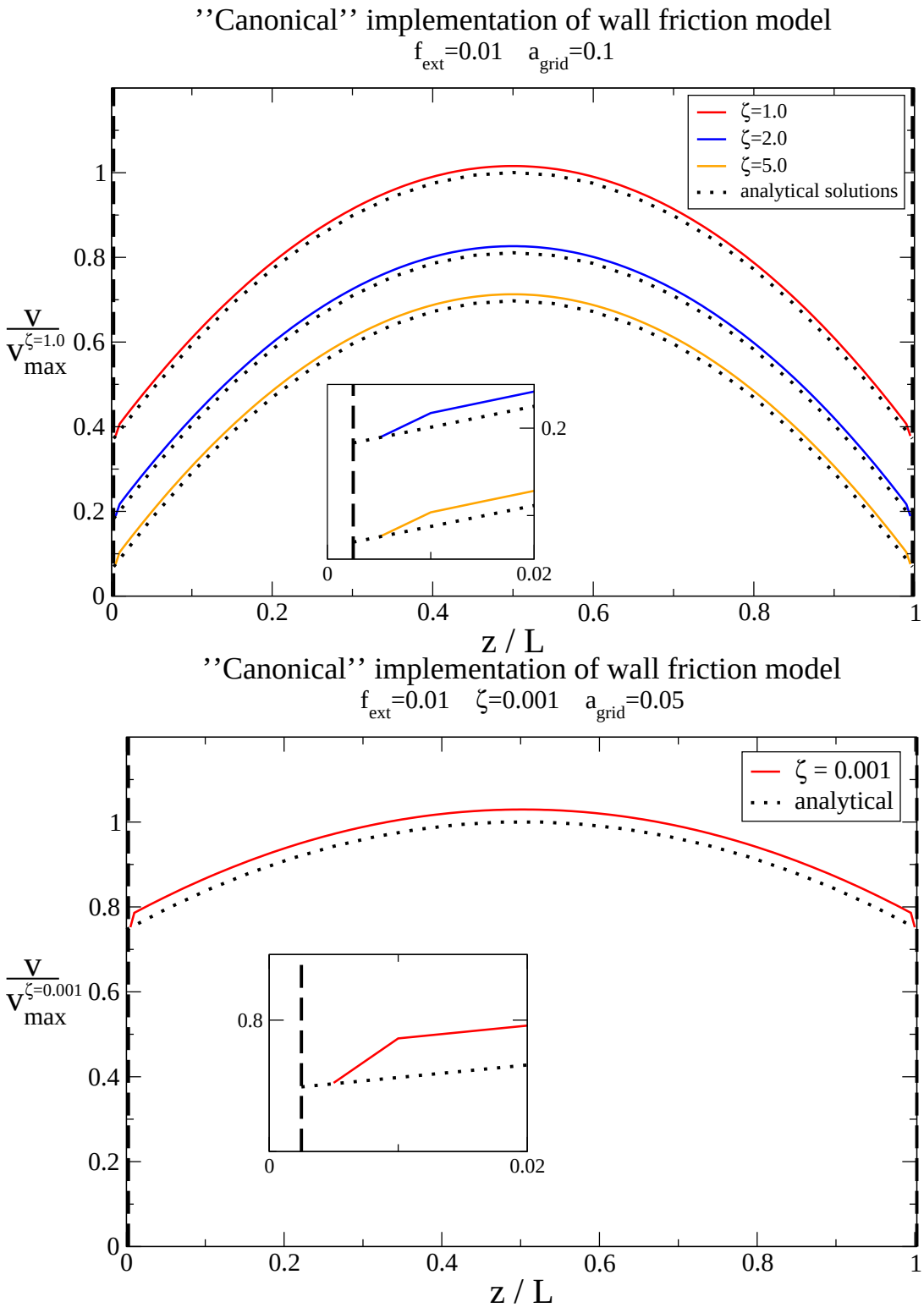


Figure 5.12: Simulation results for reduced lattice spacings a. The kink moves closer to the surface, according to the value of a, but it does not disappear. In the lower plot, the bare friction constant ζ is reduced to keep the effective friction in the same range.

$L_{\text{eff}} = 199a$. Due to the reduced lattice spacing, the kink moves closer to the boundary, but the offset from the analytical curve remains unchanged in its relative order of magnitude. This outcome is not altered when the lattice spacing is further reduced. With a lattice spacing of $a = 0.05$ and an effective channel width of $L_{\text{eff}} = 200a$ the kink is still clearly visible. Note that in this case the friction coefficient was reduced to $\zeta = 0.001$ to keep the absolute velocities in the same order of magnitude to avoid instabilities.[5]

These observations suggest that the kink and the jump in the tangential stress are not just a discretization artifact, but rather are related to the specific implementation of the wall friction force. Apparently, the "canonical" implementation leads to an unphysical tangential stress at the boundary, which must be induced by the way the boundary condition is implemented at the level of the populations. We also verified that this is a first order effect by omitting the second-order contribution from the forcing term (4.68) and the redefinition of the momentum density (4.65), without any visible effect. To investigate this in more depth, a theoretical analysis of the implementation of the wall friction force in the lattice Boltzmann model is necessary.

5.3.5 Theoretical analysis

The analysis in this section is inspired by the work of He et al. [126], in which the lattice BGK equation was solved analytically for a two-dimensional stationary channel flow. We consider the situation where $\rho =$ const. and the flow is invariant in the x and y directions. Then we can write a one-dimensional lattice Boltzmann equation

$$f_i^{\alpha+\hat{c}_{iz}}(t+\tau) = f_i^{\alpha}(t) + \sum_j \mathcal{L}_{ij}\left(f_j^{\alpha}(t) - f_j^{\text{eq}}(\rho, u_x^{\alpha})\right) + \Delta_i^{g,\alpha}, \tag{5.56}$$

where f_i^{α} denotes the population and u_x^{α} the flow velocity at $r_z = \alpha\, a$, and $\alpha = 0 \ldots L$. The forcing term is given by

$$\Delta_i^{g,\alpha} = \frac{w_i \tau}{c_s^2} c_{ix}\left(f_{\text{ext}} - \frac{\zeta}{a^3} u_x^{\alpha}(\delta_{1,\alpha} + \delta_{\alpha,N})\right), \tag{5.57}$$

where the Kronecker deltas implement the wall friction and second order terms have been dropped. In the stationary case, equation (5.56) simplifies to

$$f_i^{\alpha+\hat{c}_{iz}} = \sum_j (\delta_{ij} + \mathcal{L}_{ij})\, f_j^{\alpha} - \sum_j \mathcal{L}_{ij} f_j^{\text{eq}}(\rho, u_x^{\alpha}) + \Delta_i^{g,\alpha}. \tag{5.58}$$

The explicit expressions for the D3Q19 model are given in appendix B.3. These expressions can be used to obtain the finite-difference solution of the lattice Boltzmann equation. The algebra is straightforward but rather tedious. Therefore, it is also carried out in the appendix.

[5] Equation (5.50) shows that the velocity shift u_{slip} scales with the square of the lattice spacing. We could also introduce an effective friction $\zeta_{\text{eff}} = \zeta/a^2$ which takes into account this scaling. The reduction of the bare ζ then corresponds to keeping the effective friction ζ_{eff} roughly constant.

For simplicity, we use the single-relaxation time approximation where all eigenvalues of the collision operator are identical. The final result is

$$\begin{aligned}\eta_s \frac{u_x^{\alpha-1} - 2u_x^{\alpha} + u_x^{\alpha+1}}{a^2} = -f_{\text{ext}} + \frac{8+9\lambda+5\lambda^2}{6\lambda^2}\frac{\zeta}{a^3}u_x^{\alpha}(\delta_{1,\alpha}+\delta_{\alpha,N}) \\ - \frac{5\lambda+4}{6\lambda^2}\frac{\zeta}{a^3}(u_x^{\alpha-1}\delta_{1,\alpha-1} + u_x^{\alpha+1}\delta_{\alpha+1,N}),\end{aligned} \tag{5.59}$$

which is a second-order finite difference approximation of the Stokes equation. The crucial point to note here is the fact that there appear four Kronecker deltas on the right-hand side of equation (5.59), which stem from the friction force. Two of them, $\delta_{1,\alpha}$ and $\delta_{\alpha,N}$, act on the boundary layers, which is the expected effect of the wall friction. The other two however, $\delta_{1,\alpha-1}$ and $\delta_{\alpha+1,N}$, are additional contributions acting on the next nearest layer to the boundary. Their appearance in the second-order finite difference approximation corresponds to a peak in the second derivative of the velocities, which implies a jump in the first derivative and a kink in the velocity profile accordingly. It is a consequence of an effective delocalization of the friction force, which distorts the profile at the next nearest lattice layer. This explains why the canonical implementation of the wall friction force leads to the kink in the profiles. This effect can also be understood descriptively by looking at figure 5.10 again: In the first stage, the force is applied which changes all populations on the lattice site. In the second stage, the reflections occur which take one time step. However, only some of the populations to which the force has been applied do actually collide with the wall, while the others propagate to the bulk immediately (the dashed red ones in the middle picture). In other words, the effect of the force is split in two parts where one of them propagates into the bulk "too early" and distorts the flow profile. In conclusion, the kink induced by the canonical implementation of the friction force can be explained by undesirable contributions on the next nearest lattice sites. These contributions will be eliminated in the following.

5.3.6 Force implementation revisited: "primitive" method

In order to avoid the kink in the velocity profiles, we have to eliminate the Kronecker deltas $\delta_{1,\alpha-1}$ and $\delta_{\alpha+1,N}$ in (5.59). Their source can be easily tracked down by looking at the derivation in the appendix. The contributions from $\delta_{1,\alpha-1}$ and $\delta_{\alpha+1,N}$ entering in (B.41b) and (B.41c) cannot possibly cancel and must be eliminated. In equation (B.42) the unwanted terms stem from f_{11} to f_{14} while in equation (B.45) they stem from f_7 to f_{10}. Consequently, all the Kronecker deltas have to be removed in (B.39) from the bulk populations in order to restore the desired Stokes form of the second-order finite difference equation. This means that the only remaining possibility for the friction force to be applied are the populations which actually collide with the wall. In the D3Q19 model, only four of the populations colliding with the wall have a tangential projection onto the force. Hence, the friction force can be applied by adding half of it to the population with the positive projection, and subtracting the other half from the population with the negative projection:

$$\Delta_i^g = -\frac{1}{2}\frac{\zeta\tau^3}{a^5}(\mathbf{c}_i \cdot \mathsf{P}\mathbf{u}) \qquad\qquad \mathbf{n}\cdot\mathbf{c}_i < 0 \tag{5.60}$$

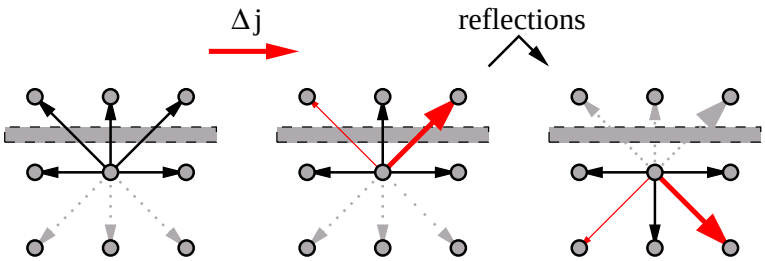


Figure 5.13: Primitive implementation of the wall friction force. Only the populations colliding with the wall get a contribution from the momentum transfer Δj. This implementation leads to correct velocity profiles in the simulations.

The resulting scheme is schematically depicted in figure 5.13. We call it the *"primitive" implementation of the friction force* because it leads back to a very simple force application scheme, which was already discussed by Ahlrichs and Dünweg [85] for the polymer coupling. In that context, however, the scheme turned out to be an unfavorable choice and was superseded by the canonical implementation [86].

5.3.7 Simulation results for "primitive" implementation

The results for the wall friction boundary condition in the primitive implementation are shown in figure 5.14. Plotted are the measured velocity profiles for a Poiseuille flow in a plane channel of width $L_{\text{eff}} = 20$ lattice spacings. The fluid has a density of $\rho = 1.0$ and a kinematic viscosity of $\nu = 3.0$. The flow is driven by a volumetric force of $f = 0.01$, which is implemented as usual. The value of the friction coefficient ζ varied from 1.0 to 5.0. The analytical solutions are plotted as dotted curves. For comparison also the results of the canonical implementation are plotted again as grey dashed curves. The measured velocity profiles for the primitive implementation show a perfect visible match to the analytical prediction. There is no kink in the curves and the amount of slippage is reproduced correctly. This shows that the primitive implementation of the wall friction model yields the desired behavior and is superior compared to the canonical implementation. This is in contrast to the bulk case, where the canonical implementation is more favorable.

5.3.8 Comparison with slip-reflection models

The primitive implementation of the friction force has an interesting connection to the family of heuristic boundary conditions motivated by kinetic theory [110, 127, 128]. This can be seen as follows. First we modify the friction force at the wall such that it acts on the momentum density

$$\mathbf{j}_B = \sum_{\mathbf{n}\cdot\mathbf{c}_i<0} f_i \mathbf{c}_i, \tag{5.61}$$

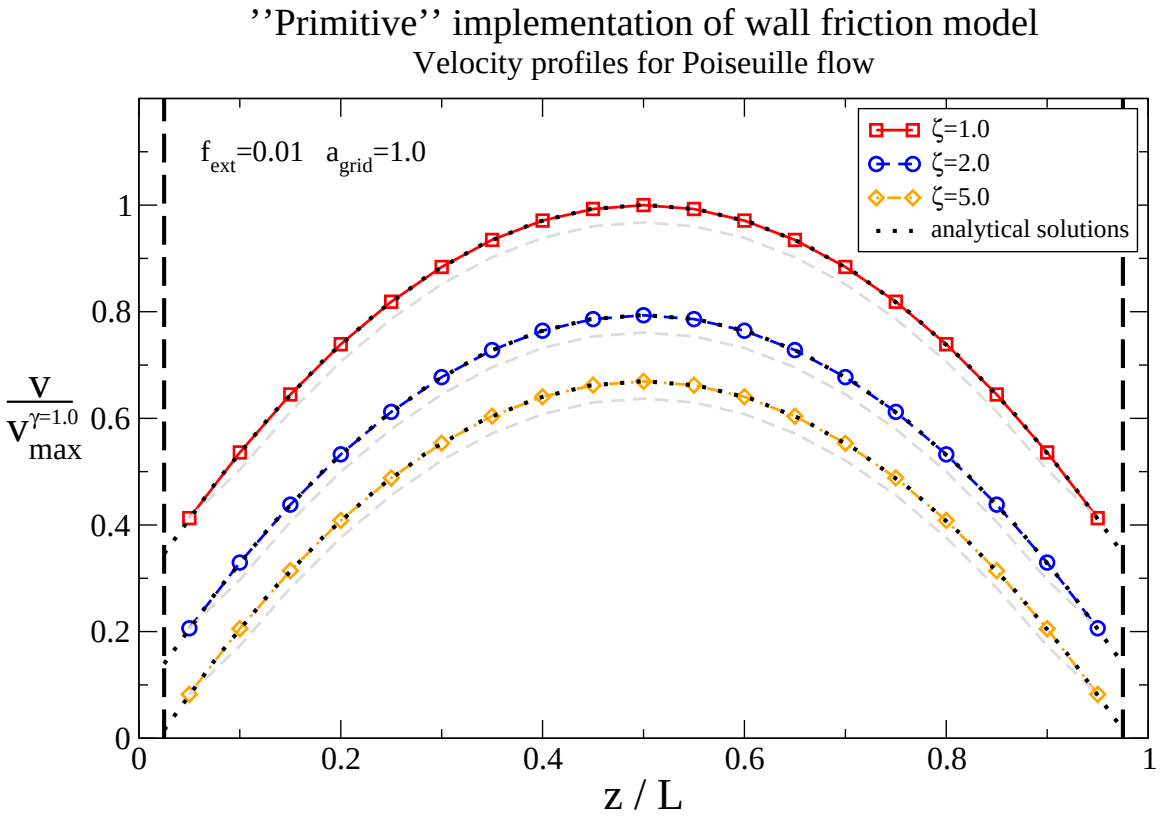


Figure 5.14: Simulation results for the "primitive" implementation of the wall friction model. The match of the measured profiles with the analytical solution is visibly perfect. For comparison, the results from the "canonical" implementation are plotted again as dashed grey curves.

where only the populations colliding with the wall contribute. This corresponds to applying the friction force after the streaming on the first lattice layer *inside* the wall, before the populations are reflected. In the steady state, the effect of this modification is a renormalization of the friction constant. The forcing term (5.60) can then be written as

$$\Delta_i^g = -\frac{1}{2}\frac{\tilde{\zeta}\tau^3}{a^2}(\mathbf{c}_i \cdot \mathsf{P}\mathbf{j}_B) = -\frac{1}{2}\frac{\tilde{\zeta}\tau^3}{a^2}\left(\mathbf{c}_i \cdot \sum_{\mathbf{n}\cdot\mathbf{c}_j<0} f_j \mathsf{P}\mathbf{c}_j\right), \qquad \mathbf{n}\cdot\mathbf{c}_i < 0, \tag{5.62}$$

where $\tilde{\zeta}$ now has units of an inverse time. Since Δ_i^g is applied to the populations that collide with the wall and are reflected, we can combine the forcing term with the specular reflections

$$\begin{aligned} f_i(\mathbf{r}_B, t+\tau) &= f^*_{i^+}(\mathbf{r}_B - \tau\mathsf{P}\mathbf{c}_i, t) - \frac{1}{2}\frac{\tilde{\zeta}\tau^3}{a^2}\left(\mathbf{c}_i \cdot \sum_{\mathbf{n}\cdot\mathbf{c}_j<0} f^*_j(\mathbf{r}_B - \tau\mathsf{P}\mathbf{c}_j, t)\mathsf{P}\mathbf{c}_j\right) \\ &= \sum_{\mathbf{n}\cdot\mathbf{c}_j<0} \mathcal{B}_{ij}(\mathbf{r}_B, \mathbf{r}_B - \tau\mathsf{P}\mathbf{c}_j, t) f^*_j(\mathbf{r} - \tau\mathsf{P}\mathbf{c}_j, t), \end{aligned} \tag{5.63}$$

where we have introduced the boundary kernel

$$\mathcal{B}_{ij}(\mathbf{r}_B, \mathbf{r}_B - \tau\mathsf{P}\mathbf{c}_j, t) = \delta_{j,i^+} - \frac{1}{2}\frac{\tilde{\zeta}\tau^3}{a^2}\mathbf{c}_i \cdot \mathsf{P}\mathbf{c}_j. \tag{5.64}$$

This boundary kernel can be written as a matrix. For the specific case of the D3Q19 model and the wall normal $\mathbf{n}$ pointing in the positive z-direction, we have

$$\begin{pmatrix} f_{15} \\ f_{11} \\ f_5 \\ f_{14} \\ f_{18} \end{pmatrix} = \begin{pmatrix} \frac{1}{2}\tilde{\zeta}\tau & 0 & 0 & 0 & 1-\frac{1}{2}\tilde{\zeta}\tau \\ 0 & \frac{1}{2}\tilde{\zeta}\tau & 0 & 1-\frac{1}{2}\tilde{\zeta}\tau & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 1-\frac{1}{2}\tilde{\zeta}\tau & 0 & \frac{1}{2}\tilde{\zeta}\tau & 0 \\ 1-\frac{1}{2}\tilde{\zeta}\tau & 0 & 0 & 0 & \frac{1}{2}\tilde{\zeta}\tau \end{pmatrix} \begin{pmatrix} f_{16} \\ f_{12} \\ f_6 \\ f_{13} \\ f_{17} \end{pmatrix}. \tag{5.65}$$

The same goes for the D2Q9 model and when we set

$$s = 1 - \frac{1}{2}\tilde{\zeta}\tau, \tag{5.66}$$

exactly the same boundary kernel as for the slip-reflection model in [110, 127] is obtained. The parameter s quantifies the reflectivity of the wall, i.e., a specularly reflecting wall has $s = 1$ while a bounce-back wall has $s = 0$. Sbragaglia and Succi [127] show that a boundary kernel as in (5.65) leads to a slip velocity which to first order is given by

$$u_{\text{slip}} = A_{\mathcal{B}}\text{Kn}\left.\frac{\partial u_x}{\partial \hat{n}}\right|_{\mathbf{r}_B}, \tag{5.67}$$

where in the continuum limit of small time and space increments

$$A_{\mathcal{B}} = \frac{1}{c_s}\frac{s}{1-s}\frac{a}{\tau}. \tag{5.68}$$

Comparing this with the expressions from kinetic theory, cf. (5.22) and (5.23), we obtain an accommodation coefficient of $\alpha = 2 - 2s$ and a slip length of

$$\delta_B = \frac{s}{1-s}\frac{l_{\text{mfp}}}{c_s}\frac{a}{\tau} = \frac{2-\tilde{\zeta}\tau}{\tilde{\zeta}\tau}\frac{\eta_s}{\rho c_s^2}\frac{a}{\tau}. \tag{5.69}$$

This is nothing but the expression (5.53) with a renormalized friction constant. The close connection of the wall friction model to the kinetic slip-reflection models gives confidence that the friction force is indeed a reasonable mesoscopic model for apparent slippage on boundary surfaces.

5.3.9 Discussion of the wall friction model

In summary, the simulation results and the theoretical analysis show that the wall friction model can successfully be used to implement a boundary condition with a tunable amount of slippage at the boundary. The basic concept is very general and can be applied to other simulation methods as well [165]. Some care has to be taken when implementing the wall friction force in the lattice Boltzmann method. If the friction force is implemented in a canonical way, the resulting Poiseuille profiles are distorted by a kink next to the boundary which is a consequence of undesirable momentum flux contributions. These can be eliminated by an apparently more primitive implementation of the friction force where only the populations that actually collide with the wall are changed. This result may seem surprising since the primitive force implementation has no systematic justification and is usually considered unfavorable in the bulk [86]. On the other hand, there is no stringent reason to expect that the formula for the bulk case is applicable at the boundary and in fact our analysis shows that this is not the case. We have rather shown that the correct discrete Stokes profile can be reproduced without artifacts only if the application of the wall friction is restricted to the populations that actually collide with the wall. In retrospective, this stands to reason because it introduces an explicit asymmetry among the different velocity directions. The breaking of the bulk symmetry is an effect of the presence of the boundary, which is explicitly built into the primitive wall friction force implementation. In this sense, this implementation, though called primitive, is more appropriate for boundary conditions. Furthermore, it can be connected to a family of kinetically motivated slip-reflection models. The wall friction force hence seems an appropriate mesoscopic simulation model that gives rise to apparent slip at boundary surfaces. The relation between the slip length and the wall friction parameter is known such that it is possible to tune the slip length systematically. The fact that this is in direct correspondence to the Navier-slip condition is thereby particularly appealing, as it makes the boundary condition compatible with kinetic theory and hydrodynamics at the same time.

The simulation results presented in this section mainly serve as a proof of concept, showing that the wall friction model leads to tunable slip in the desired way. The applicability of the wall friction boundary condition in practical simulations has yet to be explored. A concerning study of electro-osmotic flow in plane channels and a comparison with DPD simulations is currently underway.

6 Reduced symmetries in lattice Boltzmann models

In this chapter, an attempt is made to tackle the issue of lattice Boltzmann boundary conditions in a conceptually novel way. As pointed out in the previous sections, most of the existing boundary conditions can only accomplish second-order accuracy if the locality of the method is abandoned. The respective inter- and extrapolation schemes can be rather complicated and require a minimum number of nodes, which can lead to severe difficulties when the implementation has to be parallelized for execution on modern high-performance computing clusters. Moreover, the conservation laws are not directly built into these rules and, in particular, strict local mass conservation is not guaranteed in a number of schemes. One reason for the deficiencies is the heuristic nature of these boundary conditions, i.e., they are based on reflection rules for the populations that are constructed on rather ad-hoc assumptions. While this leads to the desired macroscopic boundary conditions, the systematic connection to conservation laws and symmetries – key features of the lattice Boltzmann method in the bulk – is somewhat weakened. From the viewpoint of a sound understanding of the foundations of the method this is clearly an unsatisfactory situation. For this reason, we have explored another approach for boundary conditions which is quite different from the existing ones. It is based on a lattice model that explicitly takes into account the reduced symmetry and respects the local conservation laws. The main aims of investigating the model are the following:

- We attempt to develop a completely local boundary condition that allows straightforward and easy implementation in a parallel computing environment. The conservation laws shall be built into the rules locally, and the collisions of fluid particles with each other and with the boundary shall be modeled consistently into a lattice Boltzmann collision operator at the boundary.
- We intend to get a better understanding of boundary conditions with respect to the symmetry properties of the underlying lattice structure. By treating the boundary explicitly in terms of a reduced symmetry model, we hope to be able to clarify the implications of the broken symmetry at the boundary on the lattice Boltzmann dynamics. This is particularly interesting in regard to moment systems, where self-consistent boundary conditions are impossible to provide for the moments [48, 170, 171].

In the next section, we will explain the basic ideas of the new approach and introduce the concept of reduced symmetry. Then we will treat the lattice sums and invariant tensors for the reduced symmetry. In section 6.3, we introduce several variants to construct the equilibrium distribution in the reduced symmetry. This will be used in the following subsections to devise boundary conditions for the reduced symmetry. Finally, we sketch some attempts to carry out a Chapman-Enskog expansion at the boundary in section 6.7.

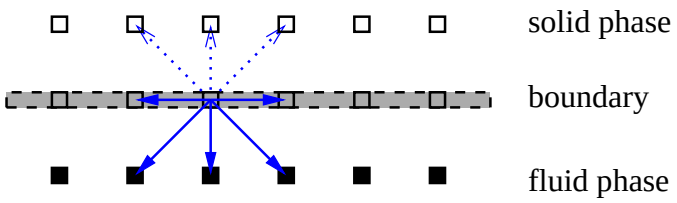


Figure 6.1: Illustration of local boundary conditions. The dotted links point into the solid and must not be populated. They are "forbidden" links that are to be excluded from the model.

6.1 Boundaries and reduced symmetry

The fundamental objective of the boundary condition is to allow for a completely local update of the lattice Boltzmann variables, that is, only information that is available on the local lattice site is used in the boundary scheme. For this purpose, it stands to reason to use a node based representation of the solid object where the surface is located directly on the lattice sites. For simplicity and as a first starting point, we consider the case of a straight wall boundary. Curved boundaries could easily be represented by specifying the surface normal individually on every boundary node. The representation of a straight wall is schematically depicted in figure 6.1. The central idea for the update scheme at the boundary is that it should have a similar structure as the bulk update in terms of streaming and collisions. The figure shows, however, that some of the links on the boundary node point into the solid. If there were populations streaming on those links, they would give rise to a fluid current into the solid. Consequently, we have to require that the links pointing into the solid are not populated at the beginning of the streaming phase. Vice versa, there are links that point into the fluid and which are undefined after the streaming phase because their populations would come from solid nodes. These populations have to be computed during the collision phase of the boundary scheme such that their value can be used in the next streaming step. In addition, we have the tangential links whose populations stream along the surface. They can be propagated in the usual way, but they can also be modified in the collision phase of the boundary rule.

In order to specify the situation formally, let us consider the pre-streaming situation where the links pointing into the solid must not be populated

$$f_i^*(\mathbf{r}_B, t) = 0, \qquad \text{if } \mathbf{c}_i \cdot \mathbf{n} < 0. \tag{6.1}$$

These links are so-called *forbidden links*, and hence they are excluded from the lattice model. For a plane wall on a D3Q19 lattice, there is a total of five forbidden links. The remaining set of 14 links depicted in figure 6.2 forms a subset of the D3Q19 model and obviously has a reduced symmetry compared to the full set. We will refer to the model with 14 links as the *reduced D3Q19 model*.

The idea is now to systematically construct a lattice Boltzmann scheme based on the links of the reduced D3Q19 model. The construction is done on the same footing as in the bulk, that is, we assume that the update of the populations consists of a streaming step and a

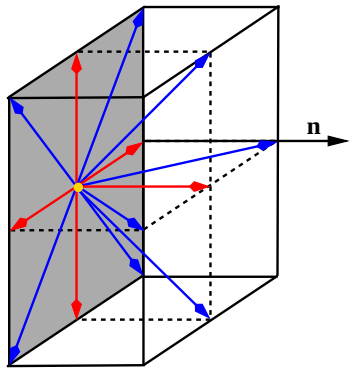


Figure 6.2: Lattice model with reduced symmetry. The forbidden links are excluded from the model. For the D3Q19 model, there are 14 remaining populations. The smaller set of velocity directions has a reduced symmetry that has to be taken into account in a formally consistent fashion.

collision step. As in the bulk, the collision step is local and has to satisfy the conservation laws for mass and momentum in the directions parallel to the plane. We assume further that the collisions can be implemented as a linear relaxation towards a local equilibrium distribution, the form of which has yet to be determined. In analogy to the MRT model, the relaxation is most generally formulated in mode space where the basis has to be constructed from the reduced symmetry D3Q19 vectors. To sum up, the central questions arising for the construction of the reduced D3Q19 model are:

- What is the equilibrium distribution for the reduced symmetry model?
- How can the reduced mode space be constructed?
- What is an appropriate collision operator for the reduced symmetry?

Before we attempt to answer these questions, we first have to look at the lattice sums and invariant tensors in the reduced symmetry.

6.2 Lattice sums and invariant tensors in the reduced symmetry

As in the bulk case, we will assume that the equilibrium distribution can be written as a polynomial expansion in the velocities. The expansion coefficients are determined by requiring that the conservation laws and symmetry properties are satisfied. The explicit calculation involves the moments of the equilibrium distribution, for which the lattice sums have to be evaluated. For the bulk case, this was done in appendix B.2. To prepare the derivation of the equilibrium distribution in the reduced symmetry, we first discuss the lattice sums and their relation to invariant tensors in the reduced symmetry. We focus on the reduced D3Q19 model for a plane wall boundary here. Generalizations to other models or geometrically complex boundaries are tedious and the existence of a solution is not guaranteed for arbitrary lattices.

6.2.1 Lattice sums for a locally plane boundary

Consider a plane wall whose normal points in the positive z-direction $\mathbf{n} = (0,0,1)^T$. The velocity vectors of the corresponding reduced D3Q19 model are the columns of the matrix

$$\mathsf{C}_B = \begin{pmatrix} 0 & 1 & -1 & 0 & 0 & 0 & 1 & -1 & 1 & -1 & 1 & -1 & 0 & 0 \\ 0 & 0 & 0 & 1 & -1 & 0 & 1 & -1 & -1 & 1 & 0 & 0 & 1 & -1 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \end{pmatrix}. \tag{6.2}$$

Any of these vectors satisfies $\mathbf{n} \cdot \mathbf{c}_i \geq 0$. The lattice sums for the reduced symmetry thus come as

$$T^{(n)}_{\alpha_1 \ldots \alpha_n} = \sum_{\mathbf{n}\cdot\mathbf{c}_i \geq 0} w_i c_{i\alpha_1} \ldots c_{i\alpha_n}. \tag{6.3}$$

The n-th lattice sum is a tensor of rank n which is invariant under any symmetry transformation for the cubic lattice and leaves the wall normal invariant. The most general form of such tensors can be constructed from proper combinations of $\delta_{\alpha\beta}$ and n_α plus potential cubic anisotropies of the respective rank, e.g., $\delta_{\alpha\beta\gamma\delta}$. In essence, the tensors can be obtained from the respective expressions in the bulk by replacing all occurrences of $\delta_{\alpha\beta}$ by $\delta^\perp_{\alpha\beta}$,

$$\delta^\perp_{\alpha\beta} = \delta_{\alpha\beta} - n_\alpha n_\beta, \tag{6.4}$$

and accordingly for the anisotropies in higher rank tensors. This procedure yields the following general expressions for the lattice sums up to fifth rank

$$\begin{aligned}
T^{(0)} &= K, \\
T^{(1)}_\alpha &= A n_\alpha, \\
T^{(2)}_{\alpha\beta} &= B_1 \delta_{\alpha\beta} + B_2 n_\alpha n_\beta, \\
T^{(3)}_{\alpha\beta\gamma} &= C_1 \left(n_\alpha \delta_{\beta\gamma} + n_\beta \delta_{\alpha\gamma} + n_\gamma \delta_{\alpha\beta}\right) + C_2 n_\alpha n_\beta n_\gamma, \\
T^{(4)}_{\alpha\beta\gamma\delta} &= D_1 \left(\delta_{\alpha\beta}\delta_{\gamma\delta} + \delta_{\alpha\gamma}\delta_{\beta\delta} + \delta_{\alpha\delta}\delta_{\beta\gamma}\right) \\
&\quad + D_2 \left(n_\alpha n_\beta \delta_{\gamma\delta} + n_\alpha n_\gamma \delta_{\beta\delta} + n_\alpha n_\delta \delta_{\beta\gamma} + n_\beta n_\gamma \delta_{\alpha\delta} + n_\beta n_\delta \delta_{\alpha\gamma} + n_\gamma n_\delta \delta_{\alpha\beta}\right) \\
&\quad + D_3 \delta_{\alpha\beta\gamma\delta} \\
&\quad + D_4 n_\alpha n_\beta n_\gamma n_\delta, \\
T^{(5)}_{\alpha\beta\gamma\delta\epsilon} &= E_1 \left[n_\alpha \delta_{\beta\gamma\delta\epsilon} + n_\beta \delta_{\alpha\gamma\delta\epsilon} + n_\gamma \delta_{\alpha\beta\delta\epsilon} + n_\delta \delta_{\alpha\beta\gamma\epsilon} + n_\epsilon \delta_{\alpha\beta\gamma\delta}\right] \\
&\quad + E_2 \left[n_\alpha \left(\delta_{\beta\gamma}\delta_{\delta\epsilon} + \delta_{\beta\delta}\delta_{\gamma\epsilon} + \delta_{\beta\epsilon}\delta_{\gamma\delta}\right) + n_\beta \left(\delta_{\alpha\gamma}\delta_{\delta\epsilon} + \delta_{\alpha\delta}\delta_{\gamma\epsilon} + \delta_{\alpha\epsilon}\delta_{\gamma\delta}\right)\right. \\
&\qquad + n_\gamma \left(\delta_{\alpha\beta}\delta_{\delta\epsilon} + \delta_{\alpha\delta}\delta_{\beta\epsilon} + \delta_{\alpha\epsilon}\delta_{\beta\delta}\right) + n_\delta \left(\delta_{\alpha\beta}\delta_{\gamma\epsilon} + \delta_{\alpha\gamma}\delta_{\beta\epsilon} + \delta_{\alpha\epsilon}\delta_{\beta\gamma}\right) \\
&\qquad \left. + n_\epsilon \left(\delta_{\alpha\beta}\delta_{\gamma\delta} + \delta_{\alpha\gamma}\delta_{\beta\delta} + \delta_{\alpha\delta}\delta_{\beta\gamma}\right)\right] \\
&\quad + E_3 \left[n_\alpha n_\beta n_\gamma \delta_{\delta\epsilon} + n_\alpha n_\beta n_\delta \delta_{\gamma\epsilon} + n_\alpha n_\gamma n_\delta \delta_{\beta\epsilon} + n_\beta n_\gamma n_\delta \delta_{\alpha\epsilon} + n_\alpha n_\beta n_\epsilon \delta_{\gamma\delta}\right. \\
&\qquad \left. + n_\alpha n_\gamma n_\epsilon \delta_{\beta\delta} + n_\beta n_\gamma n_\epsilon\, \delta_{\alpha\delta} + n_\alpha n_\delta n_\epsilon \delta_{\beta\gamma} + n_\beta n_\delta n_\epsilon \delta_{\alpha\gamma} + n_\gamma n_\delta n_\epsilon \delta_{\alpha\beta}\right] \\
&\quad + E_4\, n_\alpha n_\beta n_\gamma n_\delta n_\epsilon.
\end{aligned} \tag{6.5}$$

weight	$\hat{\mathbf{c}}_i^2$	$\mathbf{n} \cdot \hat{\mathbf{c}}_i$	$\hat{\mathbf{c}}_i$
w_0	0	0	$(0,0,0)^T$
w_1	1	0	$(1,0,0)^T, (-1,0,0)^T, (0,1,0)^T, (0,-1,0)^T$
w_2	2	0	$(1,1,0)^T, (-1,-1,0)^T, (1,-1,0)^T, (-1,1,0)^T$
w_3	1	1	$(0,0,1)^T$
w_4	2	1	$(1,0,1)^T, (-1,0,1)^T, (0,1,1)^T, (0,-1,1)^T$

Table 6.1: Weight factors for the equilibrium distribution of the reduced D3Q19 model. There are five independent weights, according to the length of the links and the projection on the boundary normal.

The coefficients in these expressions are related to the weights w_i. Using the vectors of the reduced D3Q19 model in equation (6.3), we get the following set of equations

$$\begin{aligned}
w_0 + 4w_1 + 4w_2 + w_3 + 4w_4 &= K \\
(w_3 + 4w_4)\,\frac{a}{\tau} &= A \\
(2w_1 + 4w_2 + 2w_4)\,\frac{a^2}{\tau^2} &= B_1 \\
(w_3 + 4w_4)\,\frac{a^2}{\tau^2} &= B_1 + B_2 \\
(2w_4)\,\frac{a^3}{\tau^3} &= C_1 \\
(w_3 + 4w_4)\,\frac{a^3}{\tau^3} &= 3C_1 + C_2 \\
(2w_1 + 4w_2 + 2w_4)\,\frac{a^4}{\tau^4} &= 3D_1 + D_3 \\
(w_3 + 4w_4)\,\frac{a^4}{\tau^4} &= 3D_1 + 6D_2 + D_3 + D_4 \\
(4w_2)\,\frac{a^4}{\tau^4} &= D_1 \\
(2w_4)\,\frac{a^4}{\tau^4} &= D_1 + D_2 \\
(2w_4)\,\frac{a^5}{\tau^5} &= E_1 + 3E_2 \\
0 &= E_2 \\
(2w_4)\,\frac{a^5}{\tau^5} &= E_2 + E_3 \\
(w_3 + 4w_4)\,\frac{a^5}{\tau^5} &= 5E_1 + 15E_2 + 10E_3 + E_4,
\end{aligned} \tag{6.6}$$

where the weights w_0 to w_4 are assigned to the different classes of velocity vectors as listed in table 6.1. The free parameters are the five weights w_i which allow to tune at most five of the coefficients of the lattice sums. The remaining coefficients are related by various

degeneracies, e.g., $A(a/\tau)^3 = (B_1+B_2)(a/\tau)^2 = (3C_1+C_2)(a/\tau) = 3D_1+6D_2+D_3+D_4$. In the D3Q19 model, we had already seen in the bulk that $\hat{c}_{i\alpha} = \hat{c}^3_{i\alpha}$. In the reduced D3Q19 model we have the additional degeneracy $\hat{c}_{iz} = \hat{c}^2_{iz} = \hat{c}^3_{iz} = \dots$ which leads to subtle dependencies of the moments. The choice of a complete set of conditions used to determine the weights w_i is therefore a crucial step in the construction of a reduced symmetry lattice Boltzmann model.

6.3 Approaches for constructing the equilibrium distribution

In this section we treat the equilibrium distribution for the reduced symmetry model. In the course of this work, we have worked out several approaches to devise the equilibrium distribution. Although the statistical mechanics based derivation is the most consistent approach and supersedes the prior ones, the other approaches shall be outlined here as well because in comparing the different methods the important features can be highlighted. In fact, the earlier approaches have inspired and eventually led to the development of the statistical mechanics of the lattice Boltzmann model.

6.3.1 Method 0: Direct ansatz for the equilibrium distribution

In the bulk, the equilibrium distribution is a polynomial expansion in the velocities up to second order. It can be systematically derived from the continuous Maxwell-Boltzmann distribution in terms of an expansion in Hermite tensor polynomials. However, the polynomial expansion can also just be viewed as an ansatz which is justified within the Chapman-Enskog expansion. At the boundary, we have to generalize the ansatz to the reduced symmetry appropriately. Since the normal vector $\mathbf{n}$ of the boundary is an additional invariant, the equilibrium distribution must incorporate respective terms $\mathbf{u}\cdot\mathbf{n}$ up to order $\mathcal{O}(\mathbf{u}^2)$. This suggest to modify the bulk ansatz in the following way:

$$\begin{aligned} f_i^{B,\mathrm{eq}}(\rho,\mathbf{u}) = w_i\rho\Big[1 &+ \tilde{A}\mathbf{u}\cdot\mathbf{c}_i + \tilde{B}(\mathbf{u}\cdot\mathbf{c}_i)^2 + \tilde{C}\mathbf{u}^2 + \tilde{D}\mathbf{u}\cdot\mathbf{n} + \tilde{E}(\mathbf{u}\cdot\mathbf{n})^2 \\ &+ \tilde{F}\mathbf{n}\cdot\mathbf{c}_i + \tilde{G}(\mathbf{u}\cdot\mathbf{n})(\mathbf{u}\cdot\mathbf{c}_i) + \tilde{H}(\mathbf{u}\cdot\mathbf{n})(\mathbf{n}\cdot\mathbf{c}_i) \\ &+ \tilde{I}(\mathbf{u}\cdot\mathbf{u})(\mathbf{n}\cdot\mathbf{c}_i) + \tilde{J}(\mathbf{u}\cdot\mathbf{c}_i)(\mathbf{n}\cdot\mathbf{c}_i) + \tilde{K}(\mathbf{n}\cdot\mathbf{c}_i)(\mathbf{n}\cdot\mathbf{c}_i)\Big]. \end{aligned} \tag{6.7}$$

The coefficients have to be determined such that the moment relations for the hydrodynamic variables hold,

$$\rho = \sum_{\mathbf{n}\cdot\mathbf{c}_i>0} f_i^{B,\mathrm{eq}}, \qquad \mathbf{j} = \sum_{\mathbf{n}\cdot\mathbf{c}_i>0} f_i^{B,\mathrm{eq}}\mathbf{c}_i, \qquad \Pi = \sum_{\mathbf{n}\cdot\mathbf{c}_i>0} f_i^{B,\mathrm{eq}}\mathbf{c}_i\mathbf{c}_i. \tag{6.8}$$

With the help of the lattice sums in the reduced symmetry the mass and momentum equations yield the following set of equations

$$\begin{aligned}
K + A\tilde{F} + (B_1 + B_2)\tilde{K} &= 1,\\
K\tilde{C} + A\tilde{I} + B_1\tilde{B} &= 0,\\
K\tilde{D} + A\tilde{A} + A\tilde{H} + (B_1 + B_2)\tilde{J} &= 0,\\
K\tilde{E} + A\tilde{G} + B_2\tilde{B} &= 0,\\
B_1\tilde{A} + C_1\tilde{J} &= 1,\\
B_1\tilde{G} + 2C_1\tilde{B} &= 0,\\
A + (B_1 + B_2)\tilde{F} + (3C_1 + C_2)\tilde{K} &= 0,\\
A\tilde{D} + (B_1 + B_2)(\tilde{A} + \tilde{H}) + (3C_1 + C_2)\tilde{J} &= 1,\\
A\tilde{C} + (B_1 + B_2)\tilde{I} + C_1\tilde{B} &= 0,\\
A\tilde{E} + (B_1 + B - 2)\tilde{G} + (2C_1 + C_2)\tilde{B} &= 0.
\end{aligned} \tag{6.9}$$

These are ten equations which is not enough to specify the solution for the coefficients $\tilde{A}$, $\tilde{B}$, $\tilde{C}$,... uniquely. In principle, we can get further conditions by requiring that the pressure tensor has the form of the Euler stress. However, some of the additional equations impose certain conditions on the coefficients of the lattice sums at the same time. This renders the whole procedure rather complicated and it is not straightforward how to solve the equation system. It will become clear below that in fact we are trying to determine a solution for all orders up to $\mathcal{O}(\mathbf{u}^2)$ simultaneously here. This is because in the direct ansatz for the equilibrium distribution we can not say more about the origin of the terms appearing in (6.7). Therefore, the line of the calculations is a bit unsystematic for this approach and we will not follow it further here. Instead, we develop a much more consistent formalism to derive the equilibrium distribution from variational principles that can be systematically generalized to the reduced symmetry at the boundary.

6.3.2 Method I: Derivation from quadratic functional

Although the a-priori ansatz for the equilibrium distribution can be justified by asymptotic analysis, it is more desirable to devise the equilibrium distribution systematically from general principles. In particular, we have looked for variational principles that have proven successful in many branches of theoretical physics. The idea is that the equilibrium distribution can be interpreted as a stationary state that can be found by minimizing or maximizing an appropriate functional, possibly subject to constraints. This reasoning is indeed similar to the idea of the entropic lattice Boltzmann method [28–32, 91], where a discrete lattice analogon to the Boltzmann H-function is sought in order to comply with a H-theorem. In that case, the equilibrium is not only a stationary solution of the functional but also an attractor of the dynamics. Here, we do not consider the attracting property. We merely search a functional where with a fixed-point solution that has the usual form of the bulk equilibrium distribution. The aim is then to generalize the functional to the reduced symmetry.

The bulk equilibrium distribution is a polynomial expansion in the velocities up to second order. In terms of the moments, this corresponds to a linear combination of the zeroth, first and second moment. Since the equilibrium form of the moments is known, we can incorporate them as constraints via Lagrange multipliers. A general form of the functional can then be written down as

$$\begin{aligned}\mathcal{F}(\{f_i\}) = \sum_i \mathcal{F}_i(f_i) + \lambda_\rho \left(\rho - \sum_i f_i\right) \\ + \lambda_{\mathbf{j},\alpha}\left(j_\alpha - \sum_i f_i c_{i\alpha}\right) + \lambda_{\Pi,\alpha\beta}\left(\Pi^{\mathrm{eq}}_{\alpha\beta} - \sum_i f_i c_{i\alpha}c_{i\beta}\right),\end{aligned} \tag{6.10}$$

where $\mathcal{F}_i(f_i)$ denotes a functional of a single population f_i. It should be remarked that the inclusion of the stress tensor as a constraint is not dictated by the local conservation laws, and conventionally only the "real" collisional invariants, mass and momentum, are included as constraints [32]. In this sense, the inclusion of the stress tensor yields an over-constrained equilibrium which makes it more complicated to guarantee positive entropy production [29]. That is, however, not our primary concern here. On the other hand, the stress constraint has some technical advantages because it automatically yields Galilean invariance of the equilibrium distribution. Furthermore, it is much easier to obtain the correct form of the equilibrium stress tensor in this way than by finding a functional that implies the correct form.

It remains to find an appropriate form for the $\mathcal{F}_i$. A set of linear equations for the Lagrange multipliers is only obtained, if the functional is quadratic in the populations. The simplest choice would be $\mathcal{F}_i = f_i^2$, but to allow for a relative weighting of the different neighbor shells we choose $\mathcal{F}_i = f_i^2/(2w_i)$ where the weights w_i are model dependent. The complete functional then is

$$\begin{aligned}\mathcal{F}(\{f_i\}) = \sum_i \frac{f_i^2}{2w_i} + \lambda_\rho \left(\rho - \sum_i f_i\right) \\ + \lambda_{\mathbf{j},\alpha}\left(j_\alpha - \sum_i f_i c_{i\alpha}\right) + \lambda_{\Pi,\alpha\beta}\left(\Pi^{\mathrm{eq}}_{\alpha\beta} - \sum_i f_i c_{i\alpha}c_{i\beta}\right).\end{aligned} \tag{6.11}$$

The equilibrium distribution is the stationary distribution of this functional,

$$f_i^{\mathrm{eq}} = w_i\left(\lambda_\rho + \lambda_{\mathbf{j},\alpha}c_{i\alpha} + \lambda_{\Pi,\alpha\beta}c_{i\alpha}c_{i\beta}\right), \tag{6.12}$$

where the Lagrange multipliers can be determined from the constraint equations. The calculation for the bulk case is carried out in appendix B.4. The result is the familiar expression for the lattice Boltzmann equilibrium distribution in the bulk, which justifies the above choice of the functional.

The next step is to generalize the variational formalism to the reduced symmetry. At the boundary, the outgoing populations are enforced to be zero. Formally, this can be included

as another set of constraints of the form

$$\begin{aligned}\mathcal{F}_B &= \mathcal{F} - \sum_{\mathbf{n}\cdot\mathbf{c}_i<0} \lambda_i f_i \\ &= \mathcal{F} - \sum_i \chi_i \lambda_i f_i,\end{aligned} \tag{6.13}$$

where the indicators χ_i are introduced to formally sum over all i

$$\chi_i = \begin{cases} 1 & \text{if } \mathbf{n}\cdot\mathbf{c}_i < 0 \\ 0 & \text{else.} \end{cases} \tag{6.14}$$

The boundary equilibrium then comes as

$$f_i^{B,\mathrm{eq}} = w_i \left(\lambda_\rho + \lambda_{\mathbf{j},\alpha} c_{i\alpha} + \lambda_{\Pi,\alpha\beta} c_{i\alpha} c_{i\beta} + \chi_i \lambda_i\right), \tag{6.15}$$

which differs from the bulk expression only by the last term in the brackets. This can be exploited to rearrange terms in the constraint equations which can be written in the form

$$\rho - \sum_i \chi_i \lambda_i w_i = \sum_i w_i \left(\lambda_\rho + \lambda_{\mathbf{j},\alpha} c_{i\alpha} + \lambda_{\Pi,\alpha\beta} c_{i\alpha} c_{i\beta}\right), \tag{6.16a}$$

$$j_\alpha - \sum_i \chi_i \lambda_i w_i c_{i\alpha} = \sum_i w_i c_{i\alpha} \left(\lambda_\rho + \lambda_{\mathbf{j},\beta} c_{i\beta} + \lambda_{\Pi,\beta\gamma} c_{i\beta} c_{i\gamma}\right) \tag{6.16b}$$

$$\Pi_{\alpha\beta}^{\mathrm{eq}} - \sum_i \chi_i \lambda_i w_i c_{i\alpha} c_{i\beta} = \sum_i w_i c_{i\alpha} c_{i\beta} \left(\lambda_\rho + \lambda_{\mathbf{j},\gamma} c_{i\gamma} + \lambda_{\Pi,\gamma\delta} c_{i\gamma} c_{i\delta}\right), \tag{6.16c}$$

$$\chi_i f_i^{B,\mathrm{eq}} = 0. \tag{6.16d}$$

The right-hand sides of the first three equations have the same structure as in the bulk case, hence the Lagrange multipliers for the boundary can be obtained by replacing the moments ρ, j_α and $\Pi_{\alpha\beta}$ by the respective left-hand sides in (6.16a) to (6.16c). The resulting expressions yield a linear equation system of the form

$$\chi_i \sum_k A_{ik} \lambda_k = -\chi_i f_i^{\mathrm{eq}}. \tag{6.17}$$

In principle, this could be solved numerically in order to obtain the λ_k. However, a solution only exists if the matrix A_{ik} is not singular, which not necessarily needs to be the case. This is because some of the constraint equations may be degenerate. For example, in the D3Q19 model we have

$$\Pi_{zz} = j_z \frac{a}{\tau}, \tag{6.18}$$

because $\hat{c}_{iz}^2 = \hat{c}_{iz}$ for a plane wall in the xy-plane. That is, the zz-momentum flux is linearly dependent on the momentum in z-direction and therefore j_z and Π_{zz} must not be constrained simultaneously.

A possible way to avoid the dependencies between the moments is to reduce the number of stress components that are constrained. Specifically, the constraints on Π_{zz}, Π_{xz} and Π_{yz}

are omitted for the plane wall. This corresponds to not fixing any normal stress component and is similar in spirit to the concept of reaction forces in mechanics, which are initially considered unknown and have to be found with the solution. The functional with the reduced set of constraints reads

$$\begin{aligned}\mathcal{F}_B = \sum_i \frac{f_i^2}{2w_i} + \lambda_\rho \left(\rho - \sum_i f_i\right) + \lambda_{\mathbf{j},\alpha}\left(j_\alpha - \sum_i f_i c_{ix}\right) - \lambda_{\Pi,xy}\sum_i f_i c_{ix}c_{iy} \\ + \lambda_{\Pi,xx}\left(\rho c_s^2 + \frac{j_x^2}{\rho} - \sum_i f_i c_{ix}c_{ix}\right) + \lambda_{\Pi,yy}\left(\rho c_s^2 - \sum_i f_i c_{iy}c_{iy}\right) - \sum_i \chi_i \lambda_i f_i\end{aligned} \tag{6.19}$$

and yields an equation system that can now be solved for the unknown Lagrange multipliers.

Before the explicit calculations are carried out, some remarks are in order to clarify the meaning of the weights w_i. For symmetry reasons, not every velocity direction i can have an individual weight. In the bulk, the w_i have to be constant within a neighbor shell. On the boundary, where the symmetry is reduced, there is more freedom to vary the weights: within a neighbor shell, the velocity directions can have different weights according to their projection on the boundary normal $\mathbf{n} \cdot \mathbf{c}_i$. In the D3Q19 model, for example, there are three different weights in the bulk while five different weights are possible at the boundary, cf. table 6.1. One could in principle attempt to just use the bulk weights for the boundary as well. This leads to some problems, however, as will be discussed below.

Let us now sketch the calculation of the stationary distribution of the functional $\mathcal{F}_B$. Minimization of (6.19) leads to

$$f_i^{B,\mathrm{eq}} = w_i\left(\lambda_\rho + \lambda_{\mathbf{j},\alpha}c_{i\alpha} + \lambda_{\Pi,xx}c_{ix}^2 + \lambda_{\Pi,yy}c_{iy}^2 + \lambda_{\Pi,xy}c_{ix}c_{iy} + \chi_i\lambda_i\right), \tag{6.20}$$

and the constraint equations are

$$\begin{aligned}\rho &= \sum_{\mathbf{n}\cdot\mathbf{c}_i \geq 0} f_i^{B,\mathrm{eq}}, \\ j_\alpha &= \sum_{\mathbf{n}\cdot\mathbf{c}_i \geq 0} f_i^{B,\mathrm{eq}} c_{i\alpha}, \\ \Pi_{xx}^{\mathrm{eq}} &= \sum_{\mathbf{n}\cdot\mathbf{c}_i \geq 0} f_i^{B,\mathrm{eq}} c_{ix}c_{ix}, \\ \Pi_{yy}^{\mathrm{eq}} &= \sum_{\mathbf{n}\cdot\mathbf{c}_i \geq 0} f_i^{B,\mathrm{eq}} c_{iy}c_{iy}, \\ \Pi_{xy}^{\mathrm{eq}} &= \sum_{\mathbf{n}\cdot\mathbf{c}_i \geq 0} f_i^{B,\mathrm{eq}} c_{ix}c_{iy}.\end{aligned} \tag{6.21}$$

The evaluation of the right-hand sides of the constraints involves the lattice sums for the

reduced symmetry. With the formulas devised in section 6.2 we get the equation system

$$\begin{aligned}
\rho &= K\lambda_\rho + An_\alpha\lambda_{\mathbf{j},\alpha} + B_1\left(\lambda_{\mathsf{\Pi},xx} + \lambda_{\mathsf{\Pi},yy}\right), \\
j_\alpha &= An_\alpha\lambda_\rho + \left(B_1\delta_{\alpha\beta} + B_2 n_\alpha n_\beta\right)\lambda_{\mathbf{j},\beta} + C_1 n_\alpha\left(\lambda_{\mathsf{\Pi},xx} + \lambda_{\mathsf{\Pi},yy}\right), \\
\Pi^{\mathrm{eq}}_{xx} &= B_1\lambda_\rho + C_1 n_\alpha\lambda_{\mathbf{j},\alpha} + (3D_1 + D_3)\lambda_{\mathsf{\Pi},xx} + D_1\lambda_{\mathsf{\Pi},yy}, \\
\Pi^{\mathrm{eq}}_{yy} &= B_1\lambda_\rho + C_1 n_\alpha\lambda_{\mathbf{j},\alpha} + D_1\lambda_{\mathsf{\Pi},xx} + (3D_1 + D_3)\lambda_{\mathsf{\Pi},yy}, \\
\Pi^{\mathrm{eq}}_{xy} &= D_1\lambda_{\mathsf{\Pi},xy}.
\end{aligned} \tag{6.22}$$

Solving this system involves some lengthy algebra which finally yields

$$\begin{aligned}
\lambda_\rho &= \frac{(B_1+B_2)(4D_1+D_3) - 2C_1^2}{R}\rho + \frac{2B_1C_1 - A(4D_1+D_3)}{R}j_z \\
&\quad + \frac{AC_1 - B_1(B_1+B_2)}{R}(\Pi^{\mathrm{eq}}_{xx} + \Pi^{\mathrm{eq}}_{yy}), \\
\lambda_{\mathbf{j},x} &= B_1^{-1}j_x, \\
\lambda_{\mathbf{j},y} &= B_1^{-1}j_y, \\
\lambda_{\mathbf{j},z} &= \frac{2B_1C_1 - A(4D_1+D_3)}{R}\rho + \frac{K(4D_1+D_3) - 2B_1^2}{R}j_z \\
&\quad + \frac{AB_1 - KC_1}{R}(\Pi^{\mathrm{eq}}_{xx} + \Pi^{\mathrm{eq}}_{yy}), \\
\lambda_{\mathsf{\Pi},xx} &= \frac{AC_1 - B_1(B_1+B_2)}{R}\rho + \frac{AB_1 - KC_1}{R}j_z \\
&\quad + \frac{K(B_1+B_2) - A^2}{2R}(\Pi^{\mathrm{eq}}_{xx} + \Pi^{\mathrm{eq}}_{yy}) + \frac{1}{2(4D_1+D_3)}(\Pi^{\mathrm{eq}}_{xx} - \Pi^{\mathrm{eq}}_{yy}), \\
\lambda_{\mathsf{\Pi},yy} &= \frac{AC_1 - B_1(B_1+B_2)}{R}\rho + \frac{AB_1 - KC_1}{R}j_z \\
&\quad + \frac{K(B_1+B_2) - A^2}{2R}(\Pi^{\mathrm{eq}}_{xx} + \Pi^{\mathrm{eq}}_{yy}) - \frac{1}{2(4D_1+D_3)}(\Pi^{\mathrm{eq}}_{xx} - \Pi^{\mathrm{eq}}_{yy}), \\
\lambda_{\mathsf{\Pi},xy} &= D_1^{-1}\Pi_{xy},
\end{aligned} \tag{6.23}$$

where

$$R = K(4D_1+D_3)(B_1+B_2-\frac{A^2}{K}) - 2B_1^2(B_1+B_2) + 2C_1(2AB_1 - KC_1). \tag{6.24}$$

The coefficients K, A, B_1, ... are related to the weights w_i. In principle, any set of weights that is compatible with the reduced symmetry at the boundary can be chosen. The systematic determination of an appropriate set of weights shall be postponed to the next subsection. Nevertheless, we can already make some observations. The coefficient K is the sum of all the weights, which represents a normalization and hence the choice $K = 1$ suggests itself. Furthermore, the coefficient D_3 is the prefactor of the cubic anisotropy in the fourth-rank lattice sum. Since we still have rotational invariance in the xy-plane, the cubic anisotropy should vanish, i.e., $D_3 = 0$. If the bulk weights were used as a first guess for the boundary, we would get $K = 5/6$ and $D_3 = -1/18$, in contradiction to what has just been proposed. Therefore, the bulk weights are inappropriate at the boundary.

At this point, a guideline for choosing further conditions for the coefficients and the weights is not obvious. This is because there is no clear physical interpretation of the quadratic functional, which merely forms a formal starting point for deriving the equilibrium distribution from a variational formalism. The form of the functional is only justified a-posteriori by the correct outcome for the equilibrium distribution in the bulk. It is much less straightforward to construct the equilibrium distribution at the boundary because the connection to physical principles remains somewhat obscure. The situation can be improved by using the underlying statistical mechanics of the lattice Boltzmann method to apply it at the boundary. This will be done in the next subsection, which supersedes the previous results.

6.3.3 Method II: Statistical mechanics based derivation

In chapter 3, we have discussed the statistical mechanics of the lattice Boltzmann method. It provides a systematic derivation of the equilibrium distribution from the entropy of the generalized lattice gas model. The statistical considerations remain valid for the reduced symmetry lattice model, hence the entropy at the boundary keeps the same form as in the bulk but with the sum running only over allowed links

$$S(\{\nu_i\}) = -\sum_{\mathbf{n}\cdot\mathbf{c}_i>0} \left(\nu_i \ln \nu_i - \nu_i - \nu_i \ln \bar{\nu}_i + \bar{\nu}_i\right). \tag{6.25}$$

The constraints of mass and momentum conservation are again taken into account via Lagrange multipliers. The functional to maximize at the boundary is then

$$\mathcal{S}(\{\nu_i\}, \chi, \boldsymbol{\lambda}) = S(\{\nu_i\}) + \chi\left(\sum_{\mathbf{n}\cdot\mathbf{c}_i>0} \nu_i - \frac{\rho}{\mu}\right) + \boldsymbol{\lambda}\cdot\left(\sum_{\mathbf{n}\cdot\mathbf{c}_i>0} \nu_i\mathbf{c}_i - \frac{\mathbf{j}}{\mu}\right). \tag{6.26}$$

The formal solution for the allowed links has still the same form as in the bulk, if expressed in terms of the Lagrange multipliers

$$\nu_i^{B,\mathrm{eq}} = \bar{\nu}_i \exp\left(\chi + \boldsymbol{\lambda}\mathbf{c}_i\right). \tag{6.27}$$

We can again expand the equilibrium distribution about the reference state where the fluid is at rest. Furthermore, we assume that the mass density of the boundary node is distributed among the velocity directions according to

$$f_i^{B,\mathrm{eq}}(\rho, \mathbf{j} = 0) = w_i\rho, \tag{6.28}$$

which fixes the normalization of the weights to

$$\sum_{\mathbf{n}\cdot\mathbf{c}_i>0} w_i = K = 1. \tag{6.29}$$

The reduced symmetry is now taken into account by a larger set of possible weights w_i for the different velocity directions, cf. section 6.2 and table 6.1. After expanding the

equilibrium distribution up to second order, we have

$$\begin{aligned} f_i^{B,\mathrm{eq}} = w_i\rho\Big[1 + \epsilon\chi^{B,(1)} + \epsilon\boldsymbol{\lambda}^{B,(1)}\cdot\mathbf{c}_i \\ + \frac{1}{2}\epsilon^2\left(\chi^{B,(1)} + \boldsymbol{\lambda}^{B,(1)}\cdot\mathbf{c}_i\right)^2 + \epsilon^2\chi^{B,(2)} + \epsilon^2\boldsymbol{\lambda}^{B,(2)}\cdot\mathbf{c}_i\Big]. \end{aligned} \tag{6.30}$$

This looks still the same as the bulk expression, but the reduced symmetry is contained in the Lagrange multipliers. On the first order, the mass and momentum constraints yield

$$\begin{aligned} \rho &= \rho\left(1 + \chi^{B,(1)} + A\lambda_z^{B,(1)}\right) \\ j_\alpha &= \rho\left[An_\alpha(1+\chi^{B,(1)}) + (B_1\delta_{\alpha\beta} + B_2 n_\alpha n_\beta)\lambda_\beta^{B,(1)}\right] \end{aligned} \tag{6.31}$$

where we have used the lattice sums for the reduced symmetry. The solution for the first-order Lagrange multipliers is

$$\begin{aligned} \chi^{B,(1)} &= -\frac{A}{P}(u_z^{(1)} - A), \\ \lambda_x^{B,(1)} &= \frac{u_x^{(1)}}{B_1}, \\ \lambda_y^{B,(1)} &= \frac{u_y^{(1)}}{B_1}, \\ \lambda_z^{B,(1)} &= \frac{u_z^{(1)} - A}{P}, \end{aligned} \tag{6.32}$$

where

$$P = B_1 + B_2 - A^2. \tag{6.33}$$

Proceeding to the second order, we get

$$\begin{aligned} 0 &= \frac{1}{2}(\chi^{B,(1)})^2 + A\chi^{B,(1)}\lambda_z^{B,(1)} + \frac{1}{2}(B_1\delta_{\alpha\beta} + B_2 n_\alpha n_\beta)\lambda_\alpha^{B,(1)}\lambda_\beta^{B,(1)} \\ &\quad + \chi^{B,(2)} + A\lambda_z^{B,(2)}, \\ 0 &= \frac{A}{2}(\chi^{B,(1)})^2 + (B_1+B_2)\chi^{B,(1)}\lambda_z^{B,(1)} + A\chi^{B,(2)} + (B_1+B_2)\lambda_z^{B,(2)} \\ &\quad + \frac{C_1}{2}\lambda_\alpha^{B,(1)}\lambda_\alpha^{B,(1)} + \frac{2C_1 + C_2}{2}(\lambda_z^{B,(1)})^2, \\ 0 &= B_1\chi^{B,(1)}\lambda_{x,y}^{B,(1)} + B_1\lambda_{x,y}^{B,(2)} + 2C_1\lambda_{x,y}^{B,(1)}\lambda_z^{B,(1)}, \end{aligned} \tag{6.34}$$

and the second-order Lagrange multipliers thus are

$$\begin{aligned}
\chi^{B,(2)} &= \left(\frac{AZ}{P} - \frac{P}{2}\right)\frac{(u_z^{(1)} - A)^2}{P^2} - \left(\frac{A(AB_1 - C_1)}{2P} + \frac{B_1}{2}\right)\frac{(u_x^{(1)})^2 + (u_y^{(1)})^2}{B_1^2}, \\
\lambda_x^{B,(2)} &= \frac{AB_1 - C_1}{B_1}\frac{u_x^{(1)}(u_z^{(1)} - A)}{B_1 P}, \\
\lambda_y^{B,(2)} &= \frac{AB_1 - C_1}{B_1}\frac{u_y^{(1)}(u_z^{(1)} - A)}{B_1 P}, \\
\lambda_z^{B,(2)} &= \frac{AB_1 - C_1}{2P}\frac{(u_x^{(1)})^2 + (u_y^{(1)})^2}{B_1^2} - \frac{Z}{P}\frac{(u_z^{(1)} - A)^2}{P^2},
\end{aligned} \tag{6.35}$$

where

$$Z = A^3 - \frac{3}{2}A(B_1 + B_2) + \frac{1}{2}(3C_1 + C_2). \tag{6.36}$$

These expressions are generally valid for a locally plane boundary whose normal points in the positive z-direction. In the next step, the coefficients of the lattice sums have to be determined. This will be done by looking at the pressure tensor. We have already seen in the previous section that it is useful to start with the tangential components Π_{xx}, Π_{yy} and Π_{xy}, for which we have

$$\begin{aligned}
\rho c_s^2 + \rho u_x^2 = \rho \Bigg[& B_1 \left(1 + \epsilon\chi^{B(1)} + \frac{1}{2}(\epsilon\chi^{B,(1)})^2 + \epsilon^2\chi^{B,(2)}\right) \\
& + C_1\left(\epsilon\lambda_z^{B,(1)} + \epsilon^2\chi^{B,(1)}\lambda_z^{B,(1)} + \epsilon^2\lambda_z^{B,(2)}\right) \\
& + \frac{D_1}{2}\epsilon^2\lambda_\gamma^{B,(1)}\lambda_\gamma^{B,(1)} + \frac{2D_1 + D_3}{2}\left(\epsilon\lambda_x^{B,(1)}\right)^2 + \frac{D_2}{2}\left(\epsilon\lambda_z^{B,(1)}\right)^2\Bigg] \\
= \rho \Big[& B_1 + (C_1 - AB_1)\epsilon\lambda_z^{B,(1)} \\
& + \left(-\frac{AB_1(AB_1 - C_1)}{2P} - \frac{B_1^2}{2} \right. \\
& \qquad \left. + \frac{C_1(AB_1 - C_1)}{B_1} + \frac{3D_1 + D_3}{2}\right)\left(\epsilon\lambda_x^{B,(1)}\right)^2 \\
& + \left(-\frac{AB_1(AB_1 - C_1)}{2P} - \frac{B_1^2}{2} \right. \\
& \qquad \left. + \frac{C_1(AB_1 - C_1)}{B_1} + \frac{D_1}{2}\right)\left(\epsilon\lambda_y^{B,(1)}\right)^2 \\
& + \left(\frac{B_1A^2}{2} + \frac{B_1AZ}{P} - \frac{B_1P}{2} \right. \\
& \qquad \left. - AC_1 - \frac{C_1Z}{P} + \frac{D_1 + D_2}{2}\right)\left(\epsilon\lambda_z^{B,(1)}\right)^2\Big],
\end{aligned} \tag{6.37}$$

$$
\begin{aligned}
\rho c_s^2 + \rho u_y^2 = \rho \Bigg[& B_1 \left(1 + \epsilon \chi^{B(1)} + \frac{1}{2} (\epsilon \chi^{B,(1)})^2 + \epsilon^2 \chi^{B,(2)} \right) \\
& + C_1 \left(\epsilon \lambda_z^{B,(1)} + \epsilon^2 \chi^{B,(1)} \lambda_z^{B,(1)} + \epsilon^2 \lambda_z^{B,(2)} \right) \\
& + \frac{D_1}{2} \epsilon^2 \lambda_\gamma^{B,(1)} \lambda_\gamma^{B,(1)} + \frac{2D_1 + D_3}{2} \left(\epsilon \lambda_x^{B,(1)} \right)^2 + \frac{D_2}{2} \left(\epsilon \lambda_z^{B,(1)} \right)^2 \Bigg] \\
= \rho \Big[& B_1 + (C_1 - AB_1) \epsilon \lambda_z^{B,(1)} \\
& + \left(- \frac{AB_1(AB_1 - C_1)}{2P} - \frac{B_1^2}{2} \right. \\
& \qquad \left. + \frac{C_1(AB_1 - C_1)}{B_1} + \frac{D_1}{2} \right) \left(\epsilon \lambda_x^{B,(1)} \right)^2 \\
& + \left(- \frac{AB_1(AB_1 - C_1)}{2P} - \frac{B_1^2}{2} \right. \\
& \qquad \left. + \frac{C_1(AB_1 - C_1)}{B_1} + \frac{3D_1 + D_3}{2} \right) \left(\epsilon \lambda_y^{B,(1)} \right)^2 \\
& + \left(\frac{B_1 A^2}{2} + \frac{AZB_1}{P} - \frac{B_1 P}{2} \right. \\
& \qquad \left. - AC_1 - \frac{ZC_1}{P} + \frac{D_1 + D_2}{2} \right) \left(\epsilon \lambda_z^{B,(1)} \right)^2 \Bigg],
\end{aligned}
\tag{6.38}
$$

and

$$
\rho u_x u_y = \rho \left[D_1 \epsilon^2 \lambda_x^{B,(1)} \lambda_y^{B,(1)} \right]. \tag{6.39}
$$

This almost directly leads to

$$
\begin{aligned}
B_1 &= c_s^2, \\
C_1 &= AB_1, \\
D_1 &= c_s^4, \\
D_2 &= B_1 B_2, \\
D_3 &= 0.
\end{aligned}
\tag{6.40}
$$

6.3.4 Weights w_i for the reduced D3Q19 model

Since in the D3Q19 model we have, cf. equation (6.6)

$$
B_1 \frac{a^2}{\tau^2} = 3D_1 + D_3, \tag{6.41}
$$

the speed of sound is fixed to $c_s^2 = \frac{1}{3} \frac{a^2}{\tau^2}$, which is compatible with the bulk speed of sound. If we plug in the results obtained so far in the equation system (6.6), we get the following

equations for the coefficients

$$\begin{aligned}
K &= 1, \\
B_1 &= c_s^2, \\
B_2 &= A\frac{a}{\tau} - c_s^2, \\
C_1 &= Ac_s^2, \\
C_2 &= A\frac{a^2}{\tau^2} - 3Ac_s^2 = 0, \\
D_1 &= c_s^4, \\
D_2 &= Ac_s^2\frac{a}{\tau} - c_s^4, \\
D_3 &= 0, \\
D_4 &= A\frac{a^3}{\tau^2} + 3c_s^4 - 6Ac_s^2\frac{a}{\tau}.
\end{aligned} \tag{6.42}$$

This system shows that we are left with only one free parameter A, for which we have to find an additional condition.

Up to here, only local constitutive equations for the moments were considered. At the boundary, however, it is also important to take into account the fluxes between the bulk and the boundary. In particular, it has to be assured that the boundary does not lead to an accumulation of mass on the wall, that is, the mass fluxes between the bulk and the surface have to balance. In equilibrium, the amount of mass that streams to the wall has to be compensated by an equal amount of mass that leaves the wall. We will refer to this condition as the *bulk balance condition*. In a pure bulk system, the balance is guaranteed because the equilibrium distribution is the same on all lattice sites. On the boundary, where the equilibrium is different from the bulk, we have to satisfy bulk balance as an additional condition. In a fluid at rest, the mass streaming from the bulk to the surface is given by

$$\rho^{\text{in}} = \sum_{\mathbf{n}\cdot\mathbf{c}_i<0} f_i^{\text{eq}}(\mathbf{u}=0). \tag{6.43}$$

Vice versa, the mass streaming from the surface to the bulk is given by

$$\rho^{\text{out}} = \sum_{\mathbf{n}\cdot\mathbf{c}_i>0} f_i^{B,\text{eq}}. \tag{6.44}$$

The condition of bulk balance then reads

$$\rho^{\text{in}} = \rho^{\text{out}}. \tag{6.45}$$

The evaluation of the sum of the bulk equilibrium can be carried out explicitly for $\mathbf{u} = 0$ and yields

$$\rho^{\text{in}} = \sum_{\mathbf{n}\cdot\mathbf{c}_i<0} f_i^{\text{eq}} = \frac{1}{6}\rho. \tag{6.46}$$

On the boundary, we consequently have to require that $\rho^{\text{out}} = \rho/6$. We get

$$\rho^{\text{out}} = \sum_{\mathbf{n}\cdot\mathbf{c}_i>0} w_i \rho = A\rho\frac{\tau}{a}, \tag{6.47}$$

which delivers the desired condition on the coefficient A

$$A = \frac{1}{6}\frac{a}{\tau}. \tag{6.48}$$

Now we have enough equations, namely five, to determine the five free weights of the reduced D3Q19 model. From the equation system (6.6) we finally obtain

$$w_0 = \frac{7}{18}, \qquad w_1 = \frac{1}{12}, \qquad w_2 = \frac{1}{36}, \qquad w_3 = \frac{1}{18}, \qquad w_4 = \frac{1}{36}. \tag{6.49}$$

This is a nontrivial result which was previously not known.

6.3.5 Boundary equilibrium for reduced D3Q19 model

The boundary equilibrium distribution for the reduced D3Q19 model is now completely specified. The most instructive way to write it down is in terms of the first order Lagrange multipliers $\lambda_\alpha^{B,(1)}$. Collecting all intermediate results we finally arrive at

$$\begin{aligned} f_i^{B,\text{eq}} = \rho w_i \Bigg\{ 1 + \epsilon\left[c_{i\alpha} - A n_\alpha\right]\lambda_\alpha^{B,(1)} + \epsilon^2 \Bigg[& \frac{1}{2}c_{i\alpha}c_{i\beta} - \frac{B_1}{2}\delta_{\alpha\beta} \\ & - A n_\alpha c_{i\beta} - \frac{Z}{P} n_\alpha n_\beta n_\gamma c_{i\gamma} \\ & + \left(\frac{AZ}{P} - \frac{P}{2} + \frac{A^2}{2} + \frac{B_1}{2}\right) n_\alpha n_\beta \Bigg] \lambda_\alpha^{B,(1)}\lambda_\beta^{B,(1)} \Bigg\} \end{aligned} \tag{6.50}$$

where the Lagrange multipliers can be generally expressed as

$$\lambda_\alpha^{B,(1)} = \frac{u_\beta^{(1)}}{B_1}\delta_{\alpha\beta} + \left(\frac{u_\beta^{(1)} - A}{P} - \frac{u_\beta^{(1)}}{B_1}\right) n_\alpha n_\beta. \tag{6.51}$$

In the expression (6.50) we can conveniently identify the terms that are already present in the bulk equilibrium distribution, and the new terms that are due to the reduced symmetry at the boundary. The x- and y-components of the Lagrange multipliers are still the same as in the bulk, whereas the z-component has to be modified at the boundary. We can observe that there is a shift A for the normal component u_z of the fluid velocity in the Lagrange multiplier. This means that the reference state around which we expand actually has a non-vanishing velocity component normal to the boundary of $u_z = 1/6\,(a/\tau)$. The reason for this is the bulk balance condition, which requires that there is a mass flux from the surface that balances the incoming mass flux from the bulk. This is important for the implementation

equilibrium distribution
$f_i^{B,\mathrm{eq}} = \rho w_i \Big\{ 1 + \epsilon \left[c_{i\alpha} - A n_\alpha \right] \lambda_\alpha^{B,(1)} + \epsilon^2 \left[\frac{1}{2} c_{i\alpha} c_{i\beta} - \frac{B_1}{2} \delta_{\alpha\beta} - A n_\alpha c_{i\beta} - \frac{Z}{P} n_\alpha n_\beta n_\gamma c_{i\gamma} + \left(\frac{AZ}{P} - \frac{P}{2} + \frac{A^2}{2} + \frac{B_1}{2} \right) n_\alpha n_\beta \right] \lambda_\alpha^{B,(1)} \lambda_\beta^{B,(1)} \Big\}$

Lagrange multipliers	
$\chi^{B,(1)}$	$-\frac{A}{P}(u_z^{(1)} - A)$
$\lambda_x^{B,(1)}$	$\frac{u_x^{(1)}}{B_1}$
$\lambda_y^{B,(1)}$	$\frac{u_y^{(1)}}{B_1}$
$\lambda_z^{B,(1)}$	$\frac{u_z^{(1)} - A}{P}$
$\chi^{B,(2)}$	$-\frac{(u_x^{(1)})^2 + (u_y^{(1)})^2}{2B_1} + \left(\frac{AZ}{P} - \frac{P}{2} \right) \frac{(u_z^{(1)} - A)^2}{P^2}$
$\lambda_x^{B,(2)}$	0
$\lambda_y^{B,(2)}$	0
$\lambda_z^{B,(2)}$	$-\frac{Z}{P^3}(u_z^{(1)} - A)^2$

coefficients	
A	$\frac{1}{6}\frac{a}{\tau}$
B_1	$\frac{1}{3}\frac{a^2}{\tau^2}$
B_2	$-\frac{1}{6}\frac{a^2}{\tau^2}$
C_1	$\frac{1}{18}\frac{a^3}{\tau^3}$
C_2	0
D_1	$\frac{1}{9}\frac{a^4}{\tau^4}$
D_2	$-\frac{1}{18}\frac{a^4}{\tau^4}$
D_3	0
D_4	$\frac{1}{6}\frac{a^4}{\tau^4}$
P	$\frac{5}{36}\frac{a^2}{\tau^2}$
Z	$\frac{5}{108}\frac{a^3}{\tau^3}$

weights	
w_0	$\frac{7}{18}$
w_1	$\frac{1}{12}$
w_2	$\frac{1}{36}$
w_3	$\frac{1}{18}$
w_4	$\frac{1}{36}$

Table 6.2: Summary of the results for the equilibrium distribution of the reduced D3Q19 model.

of the algorithm, because it means that we have to plug in the value of the reflected velocity into the equilibrium distribution, and not the desired hydrodynamic velocity of the boundary. The hydrodynamic velocity is given by the arithmetic mean of the pre- and post-reflection velocities at the boundary node. This is in accordance with the definition (4.65) of the hydrodynamic velocity in the presence of external forces. To conclude the derivation of the equilibrium distribution for the reduced D3Q19 model, we summarize the results for the relevant quantities again in table 6.2.

6.4 MRT model for the reduced symmetry

The equilibrium distribution is already enough to implement a simple BGK model at the boundary. However, as stated above, the collision processes at the surface are more complex and hence it is desirable to have a more flexible collision operator. In the following, we therefore attempt to construct a MRT-like collision operator for the boundary. In analogy to the bulk, this requires first to construct the basis for mode space in which the collision operator is assumed to be diagonal. The construction of the moment basis follows the same reasoning as in the bulk: Starting with the conserved moments, we systematically orthogonalize polynomials of the $\hat{\mathrm{c}}_i$ by the Gram-Schmidt procedure. The orthogonality relation

$$\sum_i w_i e_{ki} e_{li} = b_k \delta_{kl} \tag{6.52}$$

has the same form as in the bulk, but the weights w_i are replaced with the weights for the boundary equilibrium. The same holds for the weights used to calculate the normalization factors

$$b_k = \sum_i w_i e_{ki}^2. \tag{6.53}$$

The mass mode still corresponds to the basis vector

$$e_{0i} = 1. \tag{6.54}$$

Next, we orthogonalize the polynomials $\hat{\mathbf{c}}_{i\alpha}$ and get

$$\begin{aligned} e_{1i} &= \hat{c}_{ix}, \\ e_{2i} &= \hat{c}_{iy}, \\ e_{3i} &= \hat{c}_{iz} - \frac{1}{6}. \end{aligned} \tag{6.55}$$

The last of these basis vectors shows that $\hat{c}_{iz}$ is not automatically orthogonal to the mass mode any more, which is a result of the missing parity in the reduced symmetry. The form of the basis vector is another hint that the bulk balance condition for the mass flux is important at the boundary. To proceed to the next basis vectors, we orthogonalize quadratic polynomials. The reduced symmetry implies the degeneracy $\hat{c}_{iz} = \hat{c}_{iz}^2$ such that $\hat{c}_{iz}^2$ is not independent any more and must not be orthogonalized. We thus get five basis vectors from quadratic polynomials

$$\begin{aligned} e_{4i} &= \hat{c}_{ix}^2 + \hat{c}_{iy}^2 - \frac{2}{3}, \\ e_{5i} &= \hat{c}_{ix}^2 - \hat{c}_{iy}^2, \\ e_{6i} &= \hat{c}_{ix}\hat{c}_{iy}, \\ e_{7i} &= \hat{c}_{ix}(\hat{c}_{iz} - \frac{1}{6}), \\ e_{8i} &= \hat{c}_{iy}(\hat{c}_{iz} - \frac{1}{6}). \end{aligned} \tag{6.56}$$

In analogy to the bulk, we can identify e_{4i} as a bulk-like mode, and e_{5i} to e_{8i} as shear like modes. Continuing the Gram-Schmidt procedure with the higher order polynomials and carefully sorting out any degeneracies, we obtain the remaining five basis vectors

$$\begin{aligned} e_{9i} &= (\hat{c}_i^2 - \frac{3}{5}\hat{c}_{iz}^2 - \frac{7}{5})\hat{c}_{ix}, \\ e_{10i} &= (\hat{c}_i^2 - \frac{3}{5}\hat{c}_{iz}^2 - \frac{7}{5})\hat{c}_{iy}, \\ e_{11i} &= (\hat{c}_{ix}^2 + \hat{c}_{iy}^2 - \frac{2}{3})(\hat{c}_{iz} - \frac{1}{12}), \\ e_{12i} &= (\hat{c}_{ix}^2 - \hat{c}_{iy}^2)(\hat{c}_{iz} - \frac{1}{4}), \\ e_{13i} &= \hat{c}_i^4 - \frac{19}{11}(\hat{c}_{ix}^2 + \hat{c}_{iy}^2) - \frac{14}{11}(\hat{c}_{ix}^2 + \hat{c}_{iy}^2)\hat{c}_{iz} - \frac{67}{55}\hat{c}_{iz} + \frac{12}{55}. \end{aligned} \tag{6.57}$$

These are the kinetic modes of the reduced D3Q19 model. From equation (6.53) we get the normalization factors of the boundary basis vectors

$$\mathbf{b} = \left(1, \frac{1}{3}, \frac{1}{3}, \frac{5}{36}, \frac{4}{9}, \frac{4}{9}, \frac{1}{9}, \frac{5}{108}, \frac{5}{108}, \frac{1}{15}, \frac{1}{15}, \frac{11}{324}, \frac{1}{12}, \frac{28}{165}\right). \tag{6.58}$$

With these basis vectors, the populations can be transformed into the mode space of the boundary via the transformation

$$\mathbf{m} = \mathsf{M}^B \mathbf{f}, \tag{6.59}$$

where the transformation matrix is obtained from the basis vectors e_{ki} as

$$\mathsf{M}^B = \begin{pmatrix}
1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\
0 & 1 & -1 & 0 & 0 & 0 & 1 & -1 & 1 & -1 & 1 & -1 & 0 & 0 \\
0 & 0 & 0 & 1 & -1 & 0 & 1 & -1 & -1 & 1 & 0 & 0 & 1 & -1 \\
-\frac{1}{6} & -\frac{1}{6} & -\frac{1}{6} & -\frac{1}{6} & -\frac{1}{6} & \frac{5}{6} & -\frac{1}{6} & -\frac{1}{6} & -\frac{1}{6} & -\frac{1}{6} & \frac{5}{6} & \frac{5}{6} & \frac{5}{6} & \frac{5}{6} \\
-\frac{2}{3} & \frac{1}{3} & \frac{1}{3} & \frac{1}{3} & \frac{1}{3} & -\frac{2}{3} & \frac{4}{3} & \frac{4}{3} & \frac{4}{3} & \frac{4}{3} & \frac{1}{3} & \frac{1}{3} & \frac{1}{3} & \frac{1}{3} \\
0 & 1 & 1 & -1 & -1 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & -1 & -1 \\
0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & -1 & -1 & 0 & 0 & 0 & 0 \\
0 & -\frac{1}{6} & \frac{1}{6} & 0 & 0 & 0 & -\frac{1}{6} & \frac{1}{6} & -\frac{1}{6} & \frac{1}{6} & \frac{5}{6} & -\frac{5}{6} & 0 & 0 \\
0 & 0 & 0 & -\frac{1}{6} & \frac{1}{6} & 0 & -\frac{1}{6} & \frac{1}{6} & \frac{1}{6} & -\frac{1}{6} & 0 & 0 & \frac{5}{6} & -\frac{5}{6} \\
0 & -\frac{2}{5} & \frac{2}{5} & 0 & 0 & 0 & \frac{3}{5} & -\frac{3}{5} & \frac{3}{5} & -\frac{3}{5} & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & -\frac{2}{5} & \frac{2}{5} & 0 & \frac{3}{5} & -\frac{3}{5} & -\frac{3}{5} & \frac{3}{5} & 0 & 0 & 0 & 0 \\
\frac{1}{18} & -\frac{1}{36} & -\frac{1}{36} & -\frac{1}{36} & -\frac{1}{36} & -\frac{11}{18} & -\frac{1}{9} & -\frac{1}{9} & -\frac{1}{9} & -\frac{1}{9} & \frac{11}{36} & \frac{11}{36} & \frac{11}{36} & \frac{11}{36} \\
0 & -\frac{1}{4} & -\frac{1}{4} & \frac{1}{4} & \frac{1}{4} & 0 & 0 & 0 & 0 & 0 & \frac{3}{4} & \frac{3}{4} & -\frac{3}{4} & -\frac{3}{4} \\
\frac{12}{55} & -\frac{28}{55} & -\frac{28}{55} & -\frac{28}{55} & -\frac{28}{55} & 0 & \frac{42}{55} & \frac{42}{55} & \frac{42}{55} & \frac{42}{55} & 0 & 0 & 0 & 0
\end{pmatrix}.$$

The relations between the moments and the hydrodynamic variables at the boundary are

$$\begin{aligned}
\rho &= m_0 \\
j_x &= m_1\, a/\tau \\
j_y &= m_2\, a/\tau \\
j_z &= (m_3 + m_0/6)\, a/\tau \\
\Pi_{xx} &= \frac{1}{2}\left(m_4 + m_5 + \frac{2}{3} m_0\right)\, (a/\tau)^2 \\
\Pi_{yy} &= \frac{1}{2}\left(m_4 - m_5 + \frac{2}{3} m_0\right)\, (a/\tau)^2 \\
\Pi_{xy} &= m_6\, (a/\tau)^2 \\
\Pi_{xz} &= (m_7 + m_1/6)\, (a/\tau)^2 \\
\Pi_{yz} &= (m_8 + m_2/6)\, (a/\tau)^2.
\end{aligned} \tag{6.60}$$

To proceed further, the projections of the mode basis on the equilibrium distribution have to evaluated. With the expression (6.50) for the equilibrium distribution and the mode basis e_{ki},

we get

$$
\begin{aligned}
m_0^{\text{eq}} &= \rho \\
m_1^{\text{eq}} &= \rho u_x \,(a/\tau)^{-1} \\
m_2^{\text{eq}} &= \rho u_y \,(a/\tau)^{-1} \\
m_3^{\text{eq}} &= \rho \left(u_z - \frac{1}{6}\frac{a}{\tau} \right) (a/\tau)^{-1} \\
m_4^{\text{eq}} &= \rho (u_x^2 + u_y^2) \,(a/\tau)^{-2} \\
m_5^{\text{eq}} &= \rho (u_x^2 - u_y^2) \,(a/\tau)^{-2} \\
m_6^{\text{eq}} &= \rho u_x u_y \,(a/\tau)^{-2} \\
m_7^{\text{eq}} &= \rho u_x \left(u_z - \frac{1}{6}\frac{a}{\tau} \right) (a/\tau)^{-2} \\
m_8^{\text{eq}} &= \rho u_y \left(u_z - \frac{1}{6}\frac{a}{\tau} \right) (a/\tau)^{-2} \\
m_9^{\text{eq}} &= 0 \\
m_{10}^{\text{eq}} &= 0 \\
m_{11}^{\text{eq}} &= 0 \\
m_{12}^{\text{eq}} &= 0 \\
m_{13}^{\text{eq}} &= 0.
\end{aligned}
\tag{6.61}
$$

Like in the bulk, the kinetic modes have no projection on the equilibrium distribution because of the inclusion of the weights w_i in (6.52). We further note the special role of the z-component of the flow velocity, which always appears in its orthogonalized form $u_z - 1/6\,(a/\tau)$ in the above expressions. This observation has a connection to the bulk balance condition, as can be seen from the relation

$$
\rho^{\text{out}} \frac{a}{\tau} = j_z. \tag{6.62}
$$

It was already noted above that in order to satisfy bulk balance, the reflected momentum has to be used for j_z and that it has to be $j_z = 1/6\,\rho\,(a/\tau)$ for a fluid at rest. Here we observe now, that the corresponding moment m_3 vanishes in this case, that is, for a fluid at rest the momentum-like moments m_1 to m_3 are consistently all zero. Moreover, there is a relation between the momentum modes and the first order Lagrange multipliers

$$
\begin{aligned}
\rho c_s^2 \lambda_x^{B,(1)} &= m_1\,(a/\tau), \\
\rho c_s^2 \lambda_y^{B,(1)} &= m_2\,(a/\tau), \\
\rho P \lambda_z^{B,(1)} &= m_3\,(a/\tau).
\end{aligned}
\tag{6.63}
$$

These relations are similar to the ones obtained in the bulk, except for the z-component which encodes the effects of the reduced symmetry. It is a striking feature of our formalism, that the reduced symmetry can be systematically incorporated and automatically yields consistent expressions, if the additional bulk balance condition is taken into account.

6.4.1 Collision operator at the boundary

It remains to construct a linear collision operator at the boundary. As in the bulk, we assume that the operator is diagonal in mode space such that the collisions describe a linear relaxation of the non-equilibrium part of moments

$$m_k^{*\text{neq}} = (1 + \lambda_k) m_k^{\text{neq}}. \tag{6.64}$$

The choice of the eigenvalues λ_k has to be guided by the symmetry properties of the model. Looking at the basis vectors, we can divide the modes at the boundary into the conserved modes m_0 to m_3, the bulk-like mode m_4, the shear-like modes m_5 to m_8 and the kinetic modes m_9 to m_{13}. We choose to relax the bulk and shear modes with the same eigenvalues as in the bulk

$$\begin{aligned} m_4^{*\text{neq}} &= (1 + \lambda_b) m_4^{\text{neq}}, \\ m_k^{*\text{neq}} &= (1 + \lambda_s) m_k^{\text{neq}} \qquad 5 \leq k \leq 8. \end{aligned} \tag{6.65}$$

For the kinetic modes, we use $\lambda_k = -1$ which corresponds to project them out during the collision step. It should be emphasized that this is only a first guess. Strictly speaking, the choice of the eigenvalues has to be justified by an asymptotic analysis. This turns out to be rather complicated and eventually impossible, such that we have to stick to the guess at this point. It will turn out, unfortunately, that this choice of the MRT collision operator leads to undesirable artifacts in simulations of simple Poiseuille flow.

6.5 Results for the reduced symmetry model

We have tested the reduced symmetry model for boundary conditions in the case of a Poiseuille flow in a plane channel, similar to the case in section 5.3. The channel has a width of $L_{\text{eff}} = 20a$ in the z-direction. Periodic boundary conditions were applied in the x- and y-direction. The fluid has a density of $\rho = 1.0$ and a kinematic viscosity of $\nu = 3.0$, both measured in lattice units. The flow is driven by a volumetric force $f_{\text{ext}} = 0.01$. The boundary nodes are located on the first and last lattice layer with respect to the z-coordinate. On these nodes, boundary conditions are applied in the following way. After the streaming step, we transform the incoming populations into moment space according to equation (6.59). Note that some of the entries in the matrix M^B change their sign depending on whether we treat the top or bottom wall. After the transformation, we can apply the necessary operations in moment space: the z-component of the flow velocity is reversed, while the x- and y-components are decreased by a frictional force in order to generate a tunable slip boundary condition, cf. section 5.3. The shear and bulk-like modes are then relaxed towards their equilibrium value which is evaluated with the intermediate flow velocity. The eigenvalues $\lambda_b = \lambda_s = \lambda$ are calculated from the viscosity ν. Having applied these modifications to the moments, they are transformed back to the outgoing populations with the respective inverse matrix for the space spanned by the reflected directions. This completes the boundary collision process. It should be noted that all operations require local information only, and that

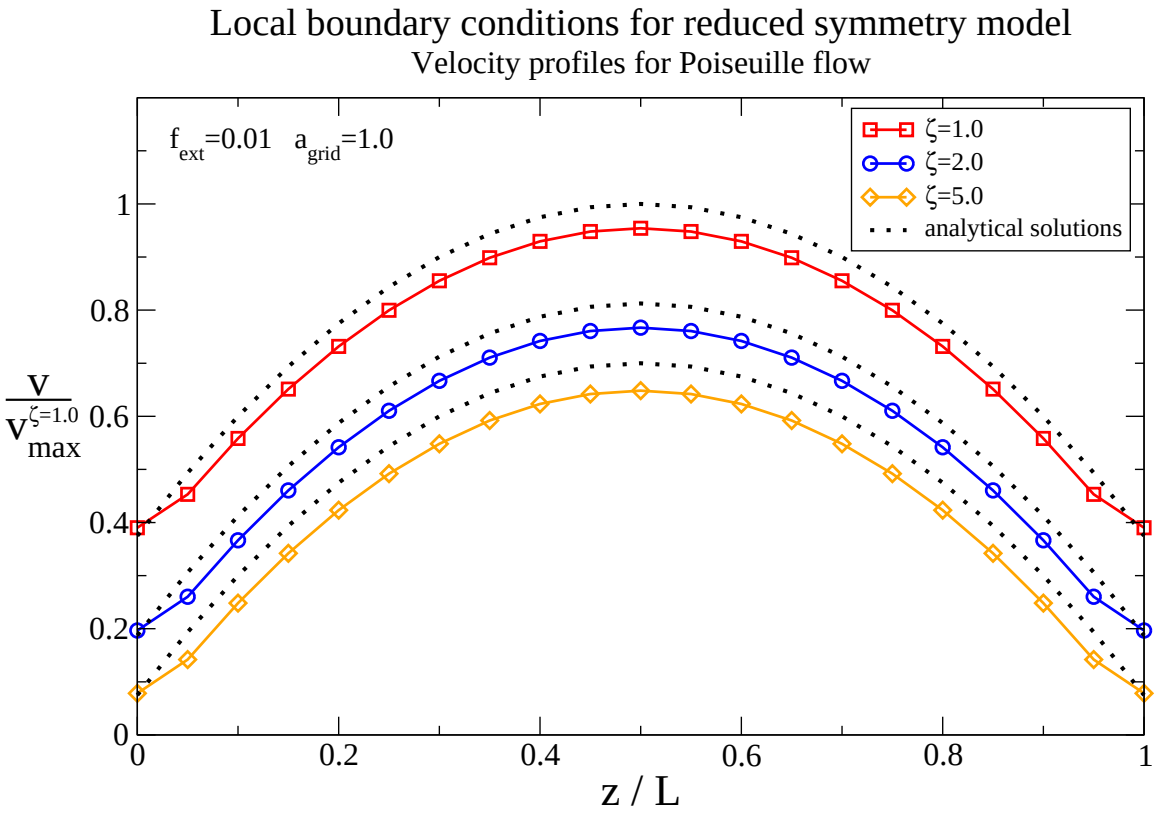


Figure 6.3: Simulation results with the local boundary condition in the reduced symmetry. The profiles deviate from the analytical solution and show a kink next to the boundary. The latter is caused by an unphysical momentum transfer due to artifacts stemming from the reduced symmetry.

only two parameters are involved, namely the viscosity ν (a bulk property) and the friction coefficient ζ (a surface property).

The results for the velocity profiles in the channel are shown in figure 6.5 for three different friction coefficients between 1.0 and 5.0. The plots show a parabolic profile but, once more, a kink is present at the fluid node next to the boundary. Although visibly the curves with the kink look similar to those observed in section 5.3, the kink has a different origin here, as will be brought out below. To clarify that the kink is not an artifact related to the force application, we have tried different variants of applying the force directly to the populations rather than in mode space. The results for the velocity profiles remain visibly unchanged. By experimenting with the eigenvalues of the collision operator, the modes m_7 and m_8 can be identified as being relevant for the kink. This stands to reason because these modes are connected to the normal stress components Π_{xz} and Π_{yz} which must have a jump, if the velocity profiles has a kink. However, the quantitative impact of those modes can not be systematically assessed so far. In principle it is required to conduct an asymptotic analysis or Chapman-Enskog expansion of the boundary condition for the reduced symmetry model. Some attempts to do this are presented below, but they eventually remained unsuccessful. Nevertheless, the origin of the kink can be semi-systematically identified by investigating how the Stokes equation emerges from the lattice Boltzmann equation.

6.5.1 The Stokes equation and the reduced symmetry lattice Boltzmann model

One of the central ideas of the boundary condition in the reduced symmetry is that the correct macroscopic boundary values emerge from the dynamics on the lattice level, in the same way that the lattice Boltzmann equation generates Navier-Stokes behavior in the bulk. Instead of rigorously proving this by a Chapman-Enskog expansion, a slightly more heuristic argument shall be followed here.

The lattice Boltzmann equation consists of the collision and the streaming phase. During the latter, populations propagate on the lattice and transport information about the hydrodynamic fields from one lattice site to another. The interplay of the collisions and the streaming eventually leads to the macroscopic flow profile. To study this in more detail, one can look at the x-component of the momentum density. We first consider the behavior in the bulk. Using the definition (2.30) and the lattice Boltzmann equation (2.39), we have

$$\begin{aligned} j_x(\mathbf{r},t) &= \sum_i f_i(\mathbf{r},t)c_{ix} \\ &= \sum_{\mathbf{n}\cdot\mathbf{c}_i>0} f_i(\mathbf{r},t)c_{ix} + \sum_{\mathbf{n}\cdot\mathbf{c}_i=0} f_i(\mathbf{r},t)c_{ix} + \sum_{\mathbf{n}\cdot\mathbf{c}_i<0} f_i(\mathbf{r},t)c_{ix} \\ &= \sum_{\mathbf{n}\cdot\mathbf{c}_i>0} f_i^*(\mathbf{r}-\tau\mathbf{c}_i,t-\tau)c_{ix} + \sum_{\mathbf{n}\cdot\mathbf{c}_i=0} f_i^*(\mathbf{r}-\tau\mathbf{c}_i,t-\tau)c_{ix} \\ &\quad + \sum_{\mathbf{n}\cdot\mathbf{c}_i<0} f_i^*(\mathbf{r}-\tau\mathbf{c}_i,t-\tau)c_{ix}. \end{aligned} \tag{6.66a}$$

If we assume that the flow is in the stationary state and that the profile is invariant in the x- and y-direction, the momentum density is only dependent on the z-coordinate and we can write

$$j_x(z) = \sum_{c_{iz}=+1} f_i^*(z-a)c_{ix} + \sum_{c_{iz}=0} f_i^*(z)c_{ix} + \sum_{c_{iz}=-1} f_i^*(z+a)c_{ix}. \tag{6.67}$$

The second-order Chapman-Enskog result for this specific flow is given by, cf. (4.42)

$$\begin{aligned} f_i^* = w_i\rho + w_i c_{ix}\frac{\rho u_x}{c_s^2} + \frac{\rho u_x^2}{2c_s^2}E_{ixx}^{(2)} + \rho\tau\left(\frac{1}{\lambda_s}+1\right)\frac{\partial u_x}{\partial r_z}E_{ixz}^{(2)} \\ + \frac{\tau}{2c_s^2}\left(\frac{1}{\lambda_g}+1\right)\left(\frac{\tau}{2c_s^2}\frac{\partial \rho u_x^2}{\partial r_z} - \frac{\tau\eta_s}{c_s}\frac{\partial^2 u_x}{\partial r_z^2}\right)E_{ixxz}^{(3)} + \frac{w_i\tau}{c_s^2}c_{ix}f_{\text{ext}}. \end{aligned} \tag{6.68}$$

If we plug this expression into (6.67) and exploit that sums over odd polynomials in c_{ix} vanish, we obtain

$$\begin{aligned} \sum_{c_{iz}=+1} f_i^*(z-a)c_{ix} &= \frac{1}{6}\rho u_x(z-a) + \frac{\rho a}{6}\left(\frac{1}{\lambda_s}+1\right)\left.\frac{\partial u_x}{\partial r_z}\right|_{z-a} + \frac{\tau}{6}f_{\text{ext}}, \\ \sum_{c_{iz}=0} f_i^*(z)c_{ix} &= \frac{2}{3}\rho u_x(z) + \frac{2\tau}{3}f_{\text{ext}}, \\ \sum_{c_{iz}=-1} f_i^*(z+a)c_{ix} &= \frac{1}{6}\rho u_x(z+a) - \frac{\rho a}{6}\left(\frac{1}{\lambda_s}+1\right)\left.\frac{\partial u_x}{\partial r_z}\right|_{z+a} + \frac{\tau}{6}f_{\text{ext}}, \end{aligned} \tag{6.69}$$

where the bulk weights w_i of the D3Q19 model were explicitly used. The hydrodynamic flow velocity u_x can be Taylor expanded around z, and the same goes for the gradient. If only second derivative terms are kept we get

$$\begin{aligned} u_x(z\pm a) &= u_x(z) \pm a\left.\frac{\partial u_x}{\partial r_z}\right|_z + \frac{a^2}{2}\left.\frac{\partial^2 u_x}{\partial r_z^2}\right|_z \pm \ldots, \\ \left.\frac{\partial u_x}{\partial r_z}\right|_{z\pm a} &= \left.\frac{\partial u_x}{\partial r_z}\right|_z \pm a\left.\frac{\partial^2 u_x}{\partial r_z^2}\right|_z + \ldots. \end{aligned} \tag{6.70}$$

Putting all results together yields a differential equation for the flow velocity profile

$$\begin{aligned} \rho u_x(z) &= \frac{1}{6}\rho u_x(z-a) + \frac{\rho a}{6}\left(\frac{1}{\lambda_s}+1\right)\left.\frac{\partial u_x}{\partial r_z}\right|_{z-a} + \frac{\tau}{6}f_{\text{ext}} \\ &\quad + \frac{2}{3}\rho u_x(z) + \frac{2\tau}{3}f_{\text{ext}} \\ &\quad + \frac{1}{6}\rho u_x(z+a) - \frac{\rho a}{6}\left(\frac{1}{\lambda_s}+1\right)\left.\frac{\partial u_x}{\partial r_z}\right|_{z+a} + \frac{\tau}{6}f_{\text{ext}} \\ &= \rho u_x(z) + \frac{\rho a^2}{6}\left.\frac{\partial^2 u_x}{\partial r_z^2}\right|_z - \frac{\rho a^2}{3}\frac{1+\lambda_s}{\lambda_s}\left.\frac{\partial^2 u_x}{\partial r_z^2}\right|_z + \tau f_{\text{ext}} \\ &= \rho u_x(z) - \frac{\rho c_s^2\tau^2}{2}\frac{2+\lambda_s}{\lambda_s}\left.\frac{\partial^2 u_x}{\partial r_z^2}\right|_z + \tau f_{\text{ext}}. \end{aligned} \tag{6.71}$$

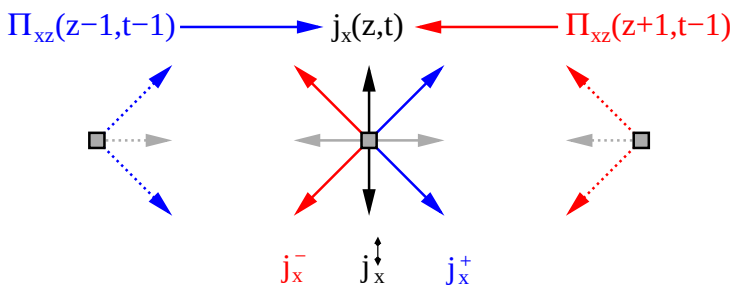


Figure 6.4: Schematic illustration of the emergence of the Stokes equation on the lattice. The full difference equation arises from a combination of terms that carry information about the flow profile from both sides of any given lattice site. At the boundary, the information from one side is missing, thus the Stokes equation is not reproduced.

Finally we arrive at

$$\eta_s \frac{\partial^2 u_x}{\partial r_z^2} = -f_{\text{ext}}, \tag{6.72}$$

which is nothing but the Stokes equation. The important point to note here is that the Stokes equation emerges from the combination of terms streaming to the node at z *from both sides*, as illustrated in figure 6.4. If either one of the terms with argument $z - a$ or $z + a$ is omitted, the resulting equation is different from the Stokes equation. In other words, the Stokes equation is inherently a bulk equation. Therefore it is not necessarily fulfilled at a boundary node, where information streams in from only one side. As a consequence, it can not be expected that the correct Stokes profile is generated by the boundary condition in the reduced symmetry. To be more precise, this is because the information about how the profile extrapolates beyond the boundary is missing. The local boundary scheme hence has to impose the desired velocity at the boundary, in the very sense of a Dirichlet boundary condition. In this aspect, local boundary conditions differ crucially from the simple link-based reflection schemes. Following this line of thinking further, it becomes clear that in fact all hydrodynamically relevant moments have to be prescribed correctly at the boundary. Besides the mass and momentum, this includes the gradients of velocity field, i.e., the non-equilibrium part of the stress tensor. The need to fix Π^{neq} was already pointed out by Lätt [101, 140] in regard to various closure relations. In the reduced symmetry, however, the attempt to fix the normal stresses at the boundary reveals another complicacy.

6.5.2 A closer look on the collisions in the reduced mode space

The tangential stresses Π_{xz} and Π_{yz} are related to the modes m_7 and m_8 in the reduced symmetry, cf. expressions (6.60). Consequently, in order to prescribe the tangential stresses we have to find the correct values to fix m_7 and m_8. However, a second look reveals another subtlety here. In the reduced symmetry mode space, Π_{xz} does not only depend on m_7 but also on $m_1 \propto j_x$. Looking at this relation in another way, this means that the stress component Π_{xz} has projections on the populations in the parallel subspace, i.e., $\mathbf{c}_i = (\pm 1, 0, 0)$.

This leads to a problem when relaxing the modes during the collision process because there is an inherent coupling of degrees of freedom that one would rather relax independently. For example, if we were to relax the stress component Π_{xz}, this would imply a relaxation of m_7 and m_1 simultaneously according to equation (6.60). This is clearly not feasible as it would violate momentum conservation. Conversely, the relaxation of the mode m_7 can not be interpreted in terms of the hydrodynamic variables, since the reduced symmetry mixes m_7 with the conserved mode m_1. It remains unclear, how the relaxations in the reduced mode space can be made consistent with the hydrodynamic fields in the bulk. We believe that more light could be shed on this issue by asymptotic analysis. Since our attempts have remained unfruitful so far, this constitutes an unresolved problem.

6.6 Revised boundary model

Although the above analysis may deliver some doubts about the feasibility of the reduced symmetry approach, we shall show that the model can still be tamed to our needs, at least for the simple Poiseuille flow. For this purpose, we will take the various points of the analysis as hints for additional modifications of the boundary condition in the reduced symmetry. The first essential observation is that none of the listed issues is related to the equilibrium distribution, which we can therefore keep in its current form. We have to modify the boundary scheme to impose the Dirichlet conditions on the hydrodynamic variables. Instead of modifying the momentum density by a friction force, we prescribe the desired slip velocity at the boundary by setting the momentum modes to the corresponding values. Fixing of the non-equilibrium stresses is yet not straightforward because of the structure of the mode space. Therefore we resort to an indirect way using a bounce-back scheme for the non-equilibrium parts of the populations

$$f_i^{+B,\text{neq}} = f_{i^-}^{B,\text{neq}}, \tag{6.73}$$

where $\mathbf{c}_{i^-} = -\mathbf{c}_i$. This does not alter the conserved moments and it assures that the non-equilibrium stress before and after the reflection is the same

$$\sum_{\mathbf{n}\cdot\mathbf{c}_i\leq 0} f_i^{B,\text{neq}}\mathbf{c}_i\mathbf{c}_i = \sum_{\mathbf{n}\cdot\mathbf{c}_i\geq 0} f_i^{B,\text{neq}}\mathbf{c}_i\mathbf{c}_i. \tag{6.74}$$

Furthermore, the relaxation at the wall has to be modified as the MRT-like model can not be implemented consistently. In this context, we resort to the simpler BGK model with a single relaxation rate λ. This has the advantage that it does not rely on the structure of the mode space and it can directly be formulated in terms of the populations

$$f_i^{*B,\text{neq}} = -\lambda f_i^{B,\text{neq}}. \tag{6.75}$$

The difference compared to the MRT-like relaxations used above is that the kinetic modes are not projected out instantaneously, but they are relaxed at the same rate as the hydrodynamic modes.

In summary, we are now back to a boundary closure scheme that expresses the outgoing populations in terms of the incoming populations at the boundary node

$$f_i^{*B} = f_i^{B,\mathrm{eq}} - \lambda \left(f_{i^-}^B - f_{i^-}^{B,\mathrm{eq}} \right), \tag{6.76}$$

where the equilibrium distributions are calculated by the formula for the reduced symmetry.

The modified boundary condition with bounce-back of non-equilibrium parts and BGK-like relaxation was once more applied to the case of Poiseuille flow in a plane channel. The setup of the system was as described above and the same parameters for the fluid were used. Instead of varying the friction coefficient, several values for the slip velocity at the boundary were imposed, ranging from stick boundaries to $u_{\mathrm{slip}} = 0.4$. The results are plotted in figure 6.5. There is no kink visible in the curves and the match with the analytical solution is visibly perfect. The amount of slippage can be controlled by the prescribed value of the slip velocity u_{slip} as desired. This shows that the modifications to the boundary condition can indeed avoid the artifacts observed for the earlier attempts. The reason is that the boundary scheme is now capable of controlling the momentum flux at the boundary node, an essential prerequisite to reproduce the Stokes equation at the hydrodynamic level. However, it should be noted that this is only possible by avoiding the mode space of the reduced symmetry model and thereby the advantages of MRT-like models are sacrificed. Moreover, a rigorous justification in terms of a Chapman-Enskog expansion is still missing. On the pro-side, it should be pointed out that the results for the Poiseuille flow show that reduced symmetry lattice Boltzmann models are a feasible approach to treat rigid boundaries. The equilibrium distribution for such models can be derived systematically in the statistical mechanics framework and can be readily applied to straight boundaries. The resulting boundary scheme satisfies the conservation laws and is completely local, which makes it very useful for parallel implementations. It has yet to be shown whether the novel boundary condition yields acceptable results for more complicated flows than plane Poiseuille flow.

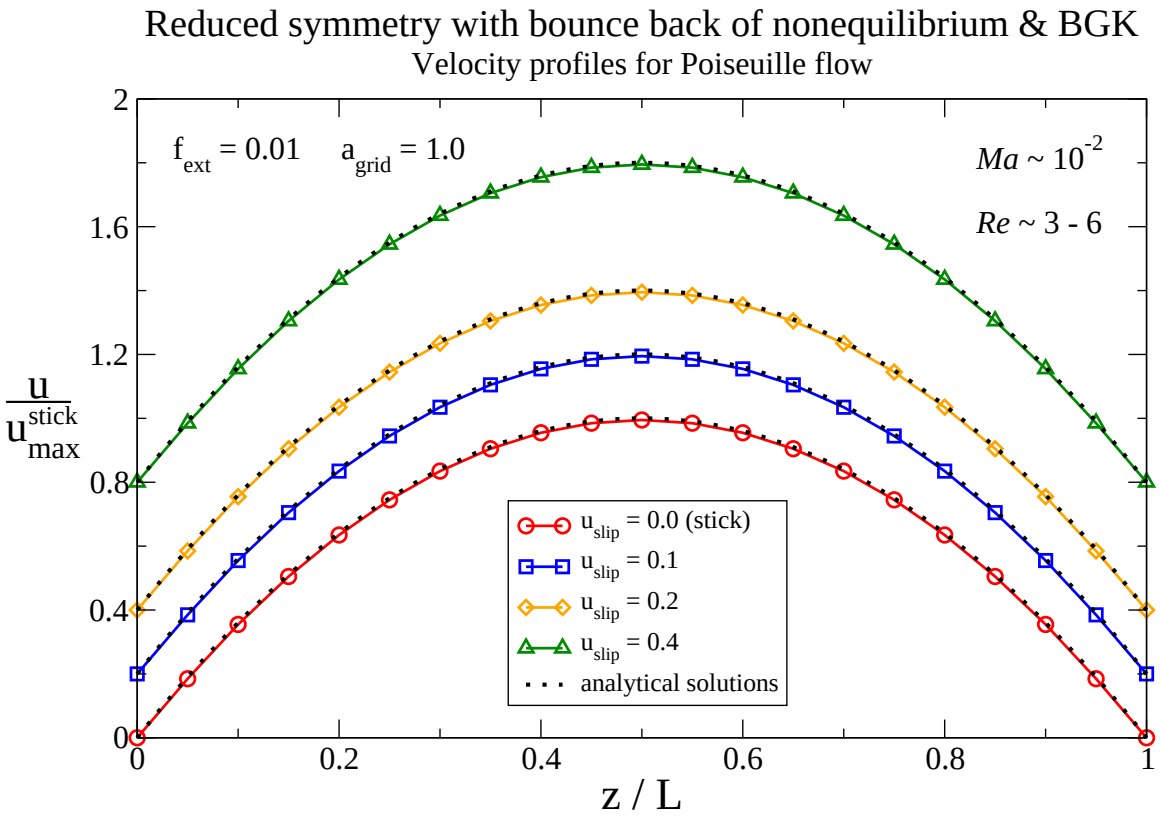


Figure 6.5: Simulation results for the reduced symmetry model with bounce-back of non-equilibrium parts and a BGK collision operator. No kink is visible in the profiles and an accurate match to the analytical solution is achieved.

6.7 Attempts for a Chapman-Enskog expansion at the boundary

The construction of the boundary condition in the reduced symmetry is mainly based on heuristic arguments so far. A rigorous proof of its correctness and an assessment of the order of accuracy requires a more systematic analysis. The Chapman-Enskog expansion is the standard technique to derive the hydrodynamic equations in the bulk case. In this section, we attempt to apply the Chapman-Enskog expansion to the reduced symmetry model. Although we have not completely succeeded in finding a consistent way of doing this, it is worthwhile to sketch some pieces of the puzzle in order to point out some of the difficulties involved with the boundary.

6.7.1 Second and third moment in the reduced symmetry

Some of the components of the pressure tensor were not explicitly evaluated during the construction of the equilibrium distribution. Since they are needed in the Chapman-Enskog expansion, we evaluate them now. From equations (6.60) and (6.61) we immediately get

$$\begin{aligned}
\Pi_{xx}^{B,\mathrm{eq}} &= \rho c_s^2 + \rho u_x^2 \\
\Pi_{yy}^{B,\mathrm{eq}} &= \rho c_s^2 + \rho u_y^2 \\
\Pi_{xy}^{B,\mathrm{eq}} &= \rho u_x u_y, \\
\Pi_{xz}^{B,\mathrm{eq}} &= \rho u_x u_z, \\
\Pi_{yz}^{B,\mathrm{eq}} &= \rho u_y u_z, \\
\Pi_{zz}^{B,\mathrm{eq}} &= \rho u_z \frac{a}{\tau}.
\end{aligned} \tag{6.77}$$

The last expression makes the degeneracy of the zz-stress and the z-momentum explicit. On the lowest order in the Chapman-Enskog expansion, it is also necessary to evaluate the third moment Φ of the equilibrium distribution. At the boundary we get

$$\begin{aligned}
\Phi_{\alpha\beta\gamma}^{B,\mathrm{eq}} &= \sum_{\mathbf{n}\cdot\geq 0} f_i^{B,\mathrm{eq}} c_{i\alpha} c_{i\beta} c_{i\gamma} \\
&= \rho \Bigg[\left(1 + \epsilon\chi^{B,(1)} + \frac{1}{2}(\epsilon\chi^{B,(1)})^2 + \epsilon^2 \chi^{B,(2)} \right) \sum_{\mathbf{n}\cdot\geq 0} w_i c_{i\alpha} c_{i\beta} c_{i\gamma} \\
&\qquad + \left(\epsilon\lambda_\delta^{B,(1)} + \epsilon^2 \chi^{B,(1)} \lambda_\delta^{B,(1)} + \epsilon^2 \lambda_\delta^{B,(2)} \right) \sum_{\mathbf{n}\cdot\geq 0} w_i c_{i\alpha} c_{i\beta} c_{i\gamma} c_{i\delta} \\
&\qquad + \frac{1}{2} \epsilon^2 \lambda_\delta^{B,(1)} \lambda_\epsilon^{B,(1)} \sum_{\mathbf{n}\cdot\geq 0} w_i c_{i\alpha} c_{i\beta} c_{i\gamma} c_{i\delta} c_{i\epsilon} \Bigg].
\end{aligned} \tag{6.78}$$

Using the lattice tensors for the reduced symmetry and the explicit values for the reduced D3Q19 model we can determine the components of the third moment as

$$\begin{aligned}
\Phi^{B,\mathrm{eq}}_{xxx} &= \rho u_x \frac{a^2}{\tau^2}, \\
\Phi^{B,\mathrm{eq}}_{yyy} &= \rho u_y \frac{a^2}{\tau^2}, \\
\Phi^{B,\mathrm{eq}}_{zzz} &= \rho u_z \frac{a^2}{\tau^2}, \\
\Phi^{B,\mathrm{eq}}_{xxy} &= \frac{\rho c_s^2}{\frac{a}{\tau} - A} \left(u_y \frac{a}{\tau} - u_y u_z \right), \\
\Phi^{B,\mathrm{eq}}_{xyy} &= \frac{\rho c_s^2}{\frac{a}{\tau} - A} \left(u_x \frac{a}{\tau} - u_x u_z \right), \\
\Phi^{B,\mathrm{eq}}_{xxz} &= \rho c_s^2 \left(u_z + \frac{A}{c_s^2} u_x^2 - \frac{A}{2c_s^2} u_y^2 \right), \\
\Phi^{B,\mathrm{eq}}_{yyz} &= \rho c_s^2 \left(u_z - \frac{A}{2c_s^2} u_x^2 + \frac{A}{c_s^2} u_y^2 \right), \\
\Phi^{B,\mathrm{eq}}_{xzz} &= \rho u_x u_z \frac{a}{\tau}, \\
\Phi^{B,\mathrm{eq}}_{yzz} &= \rho u_y u_z \frac{a}{\tau}, \\
\Phi^{B,\mathrm{eq}}_{xyz} &= 0.
\end{aligned} \tag{6.79}$$

These expressions are clearly very different from the bulk expression (4.21). If they were plugged into the first order equation (4.9b), the derivative

$$\frac{\partial}{\partial t_1} \Pi^{\mathrm{eq}}_{\alpha\beta} + \frac{\partial}{\partial r_\gamma} \Phi^{\mathrm{eq}}_{\alpha\beta\gamma} \tag{6.80}$$

would yield additional spurious terms in the normal components. However, it is doubtful whether the Chapman-Enskog expansion directly carries over from the bulk to the boundary. This will be discussed in the next subsection.

6.7.2 Anisotropic Chapman-Enskog expansion: a potential way out?

The pivotal point in the Chapman-Enskog expansion is the scale-separated version of the lattice Boltzmann equation (4.8) which can then be solved stepwise by inserting the solution on lower orders into the equations on higher orders. In the derivation of the scale-separated equations two expansion steps are involved. The first step is the introduction of the scaling parameter ϵ and the expansion of the populations in powers of ϵ. The second step is the introduction of the coarse-grained variables $\mathbf{r}_1$, t_1, t_2 and the Taylor expansion of the populations written in terms of the coarse-grained variables. In the philosophy of our boundary condition, we have assumed that the notion of an equilibrium distribution is still valid and

that the populations relax towards that equilibrium during the collisions. In this sense, the expansion in powers of ϵ remains valid

$$f_i^B = f_i^{B,\mathrm{eq}} + \epsilon f_i^{B,(1)} + \epsilon^2 f_i^{B,(2)} + \dots . \tag{6.81}$$

However, since the equilibrium distribution on the boundary is different from the equilibrium distribution in the bulk, there is a spatial discontinuity in the populations at the boundary. Therefore the Taylor expansion around a boundary node is not applicable. We have to avoid expanding the populations in the direction of the discontinuity. In the case of a straight boundary with normal in the z-direction this means that the Taylor expansion can only be executed in the x- and y-direction but not in the z-direction. In this way, we may try to expand the populations as

$$\begin{aligned} &f_i(r_{1x} + \epsilon\tau c_{ix}, r_{1y} + \epsilon\tau c_{iy}, r_z + \tau c_{iz}, t_1 + \epsilon\tau, t_2 + \epsilon\tau^2) \\ &\quad = f_i(r_{1x}, r_{1y}, r_z + \tau c_{iz}, t_1, t_2) \\ &\qquad + \epsilon\tau \left(\frac{\partial}{\partial t_1} + c_{ix}\frac{\partial}{\partial r_{1x}} + c_{iy}\frac{\partial}{\partial r_{1y}} \right) f_i(r_{1x}, r_{1y}, r_z + \tau c_{iz}, t_1, t_2) \\ &\qquad + \epsilon^2\tau \left[\frac{\partial}{\partial t_2} + \frac{\tau}{2} \left(\frac{\partial}{\partial t_1} + c_{ix}\frac{\partial}{\partial r_{1x}} + c_{iy}\frac{\partial}{\partial r_{1y}} \right)^2 \right] f_i(r_{1x}, r_{1y}, r_z + \tau c_{iz}, t_1, t_2). \end{aligned} \tag{6.82}$$

For directions i with $c_{iz} = 0$ this reduces to the same expression as in the bulk, thus the anisotropy induced by the boundary does not affect the parallel directions. In the normal direction, however, there is an additional non-local difference term involving populations at r_z and $r_z + \tau c_{iz}$ which contributes at all orders of the expansion. In particular the zeroth-order equation becomes

$$f_i^{(0)}(r_{1x}, r_{1y}, r_z + \tau c_{iz}, t_1, t_2) - f_i^{(0)}(r_{1x}, r_{1y}, r_z, t_1, t_2) = \Delta_i^{(0)}(r_{1x}, r_{1y}, r_z, t_1, t_2). \tag{6.83}$$

That is, the collision operator has non-vanishing contributions at the zeroth order due to the discontinuity at the boundary. The difference-term on the left hand side leads to difficulties when one tries to construct the moment equations because the sum $\sum_i f_i^{(0)}(r_{1x}, r_{1y}, r_z + \tau c_{iz}, t_1, t_2)$ runs through different locations, i.e., it is not a local moment on a lattice site. It is therefore unclear how macroscopic equations should be derived within the anisotropic Chapman-Enskog expansion.

There is actually an even more severe problem, which has to do with the reflections at the wall. Let us look at an outgoing link $\mathbf{c}_i \cdot \mathbf{n} > 0$ at a boundary node. The post-streaming population of this link is by construction zero

$$f_i(\mathbf{r}_B, t) = 0, \qquad \mathbf{n} \cdot \mathbf{c}_i > 0, \tag{6.84}$$

because no populations stream from the solid into the boundary node. Consequently we have

$$f_i^{\mathrm{neq}}(\mathbf{r}_B, t) = f_i(\mathbf{r}_B, t) - f_i^{\mathrm{eq}}(\mathbf{r}_B, t) = -f_i^{\mathrm{eq}}(\mathbf{r}_B, t). \tag{6.85}$$

This leads to a contradiction because according to the expansion (6.81), the left-hand side is of the order $\mathcal{O}(\epsilon^1)$ while the right-hand side is of the order $\mathcal{O}(\epsilon^0)$. Hence, the suggested anisotropic Chapman-Enskog expansion is faced with an inconsistency. The source are the reflections at the wall, which instantaneously change the populations by an $O(1)$ term. Another way to look at this is to notice that the subspace of the full lattice model that is allowed to be populated is changed by the reflections, i.e., the reduced symmetry before and after the reflections is not the same but is changed by a parity transformation. In this sense, the equilibrium distribution is not an invariant under reflections. We have not succeeded in incorporating this into a consistent Chapman-Enskog procedure, which we therefore have to leave as an open issue.

In conclusion, it can be said that the reduced symmetry seems so far inaccessible to the Chapman-Enskog expansion. The reasons are the discontinuity of the populations at the wall and the reduced symmetry, which preclude a straightforward expansion of the lattice Boltzmann equation. A rigorous asymptotic analysis turns thus out to be highly complicated. Probably it even requires to develop more sophisticated mathematical techniques which are beyond the scope of this work [cf. Refs. 92, 97 and the references therein].

7 Conclusions, discussion and outlook

This thesis presents aspects of method development for simulations of complex fluids and microflows within the lattice Boltzmann model. The lattice Boltzmann model was chosen because it has proven to be successful in simulating hydrodynamic interactions in soft matter systems. It has an established foundation in terms of kinetic theory and can be systematically linked to the hydrodynamic level by methods from asymptotic analysis. The properties of the solvent, such as density and viscosity, can be directly adjusted by the respective simulation parameters. In addition, lattice Boltzmann is very flexible and extensible, for example, it is possible to couple coarse-grained representations of colloids and polymers to the hydrodynamic flow field. The basics of the lattice Boltzmann method and its asymptotic analysis were introduced in chapters 2 and 4.

One major topic of this work was the treatment of thermal fluctuations in the lattice Boltzmann model, which was discussed in chapter 3. Previously, there was a lack of clarity about the correct way of adding thermal fluctuations to the lattice Boltzmann variables. In particular, it was debated whether the kinetic modes have to be thermalized even though they yield no contribution to the hydrodynamic equations. To tackle this problem, the generalized lattice gas model was developed and subsequently used to derive the equilibrium distribution of the lattice Boltzmann populations within a maximum entropy formalism. In doing so, the connection to the underlying statistical mechanics was restored. If the collision operator includes thermal fluctuations, it can be viewed as a Monte-Carlo process. The statistical perspective then implies that detailed-balance has to be satisfied, which leads to thermal fluctuations that are simultaneously consistent with statistical mechanics and hydrodynamics. The crucial result is that detailed-balance is only satisfied if the kinetic modes are thermalized accordingly. This ultimately clarifies the role of fluctuations in the lattice Boltzmann model, a question which, until now, was not answered satisfactorily.

The statistical mechanics framework of the lattice Boltzmann equation bears potential for future extensions. One particular case where the generalized lattice gas model will be useful is the development of advanced multiphase models. The difficulty in this context is to couple the thermodynamics of the interface between the phases to the hydrodynamics of the fluids in a consistent way. The existing multiphase models are constructed in a rather heuristic fashion. We believe that the statistical mechanics approach helps to improve these algorithms systematically.

In chapters 5 and 6, boundary conditions for the lattice Boltzmann model were discussed. Boundary conditions become particularly important in microfluidic devices where the surface to volume ratio is large. In this case, the fluid flow can be strongly affected by fluid-solid interactions at the boundary. The classical hydrodynamic stick boundary condition is

only valid up to Knudsen numbers of the order of 0.1. Beyond this value, apparent slip is observed at the boundary and the no-slip condition is no longer appropriate. Most work on boundary conditions in the lattice Boltzmann model has focused on the no-slip boundary condition. In this work, a boundary condition for tunable slip was developed that is based on the idea of a mesoscopic friction force at the boundary. The model has the advantage that it only introduces a single additional parameter for the boundary condition, which can be used to tune the amount of slippage. Moreover, the concept of a friction force at the boundary can be directly applied in other mesoscopic simulation methods, such as dissipative particle dynamics or multi-particle collision dynamics. In the lattice Boltzmann method, the details of the microscopic implementation of the friction force are highly relevant. This was demonstrated by the comparison of the "canonical" and the "primitive" implementation in section 5.3. While the canonical force implementation was known to be superior in the bulk, it has to be discarded at the boundary in favor of the primitive force implementation. The reason was identified by an analytical analysis of the flow profile revealing that the canonical implementation is affected by spurious momentum terms that lead to a kink in the measured velocity profile. In contrast, the primitive implementation produces smooth velocity profiles that match the analytical prediction. This shows that the wall friction model can be used to model a slip velocity in microflows. The friction parameter is related to the slip length, which is a quantity that can be either measured in experiments or fitted to available data. The method can thus be used to study the impact of the slip length on the behavior of microflows. So far, only Poiseuille flow in a plane channel geometry has been simulated, which served as a basic test case. Since this is a very special situation, it does not say too much about the general behavior of the boundary condition. For example, the accuracy of the method in curved geometries remains to be investigated. Also patterned surfaces with locally varying boundary friction are interesting and particularly relevant in regard to generalizations of the boundary condition to the recently introduced concept of tensorial slip [172]. Furthermore, it would be very interesting to compare the lattice Boltzmann implementation of the wall friction model to its counterpart in dissipative particle dynamics.

With regard to the Navier slip boundary condition in hydrodynamics, several other open questions remain to be investigated. Recent experimental results suggest that the linear constitutive equation for the slip velocity has to be replaced by a second-order boundary condition above a critical Knudsen number of 0.3. Such a nonlinear dependence can probably not be modeled accurately within the wall friction model. It needs to be investigated how far the validity of the wall friction model extends. Another point concerns thermal effects in microflows. It is known that, besides velocity slip, flows in microchannels show a temperature jump at the wall. Moreover, the viscous dissipation at low Reynolds number can lead to considerable heat generation in the fluid. It is therefore questionable whether the isothermal assumption is still valid in microflows. On the hydrodynamic level, the Navier-Stokes-Fourier description might be more appropriate in this case. As for the boundary condition, it is unclear whether the slip length is enough to capture the temperature jump or whether another parameter is needed. These questions could be tackled by using a thermal lattice Boltzmann model which is able to reproduce the heat transport equation. This was beyond the scope of this thesis, but it forms an interesting topic for future research.

In chapter 6, a new conceptual approach to implementing lattice Boltzmann boundary conditions was developed. It is based on the notion of reduced symmetry, which is a consequence of the fact that at the boundary some of the links of the lattice point into the solid and must have zero fluid populations. The remaining set of "allowed" links has a reduced symmetry compared to the bulk symmetry of the lattice. The idea in chapter 6 was to use this set of links as the basis for a reduced symmetry lattice Boltzmann model, which can be systematically constructed within the framework developed earlier in this thesis. Namely, the generalized lattice gas model can be used to derive the equilibrium distribution in the reduced symmetry. The boundary condition should then be incorporated in the collision process in the reduced symmetry. It turned out, however, that it is not clear how to set up an appropriate MRT model at the boundary because the mode space in the reduced symmetry is affected by degeneracies of the model that obscure the physical interpretation of the moments. A back door was found by resorting to BGK-like collisions and using bounce-back of the non-equilibrium parts to fix the normal stresses at the boundary. The resulting boundary condition requires only local information and can therefore be easily implemented in a parallel simulation environment. However, a systematic analysis in terms of a Chapman-Enskog expansion proves to be highly complicated, such that a rigorous assessment of the accuracy remains an open issue. Nevertheless, the exploration of reduced symmetry models sheds some light on the effects of the small velocity set at the boundary. An interesting question for future work would be, for example, whether the degeneracies in mode space can be removed by using a higher number of discrete velocities that include more neighbor shells.

In the course of this work, it became clear that the technical details in the reduced symmetry are quite involved and require careful treatment. Applications of the newly devised methods therefore had to take a back seat and mainly served as proof (and even more frequently as disproof) of the concepts. There is no doubt that many more practical simulations are needed to further verify the applicability and accuracy of the methods. To round up this discussion, some routes that could be followed in the future shall be outlined.

A field that attracts non-ceasing interest in the lattice Boltzmann community is turbulence. With regard to soft matter the phenomenon of turbulent drag reduction has received growing attention. It is known that dilute polymer solutions show a substantially reduced drag when pumped through channels. One hypothesis to explain this effect is that there is an interplay between polymer dynamics and turbulence that damps vortex structures near the wall. The lattice Boltzmann method is an ideal candidate to study this effect in simulations, because both the fluid-polymer coupling and the boundary effects are readily included. The question is whether sufficient time and length scales can be simulated within acceptable computing time. More sophisticated multiscale techniques that try to cope with the problem are currently under active development.

Another potentially very interesting application is to study electrophoresis and electroosmosis in microchannels. The presence of charged particles adds another level of complexity to a soft matter system as electrokinetic effects come into play. In a channel filled with a polyelectrolyte, the walls usually become charged through the release of counterions

into the solution. The counterions form a charged Debye layer close to the wall, which can be acted on by an external electric field thereby generating a plug flow in the channel. The electro-osmotic mobility is determined by the chemico-physical properties of the surface, and a finite slip length on the wall increases the mobility. This effect could be exploited to improve the efficiency of electro-osmotic pumps. So far, there are only a few simulations of electro-osmotic flow which mostly focus on the free-flow case. The interplay of boundary conditions and electro-kinetic effects has not been studied extensively and forms an interesting application for the tunable slip boundary condition developed. Going one step further in the method development, one could think about how polarization effects of the solvent can be incorporated in the simulation methods.

A different application field which is more related to the fluctuating lattice Boltzmann is formed by multiphase flows. For example, the flow behavior of emulsions is highly relevant for commercial applications in the cosmetic or food industries. A number of interesting questions arises concerning the phase behavior under flow, e.g., mixing, demixing and various kinds of pattern formation. Of particular interest is the dynamics of liquid droplets in microdevices. Multiphase flows are governed by the interplay of interface and surface effects. Concerning boundary conditions, the wetting properties of the different phases have to be taken into account. The difficulty in a computer simulation is that both thermodynamics and hydrodynamics have to be accurately reproduced. As to the lattice Boltzmann method, is has to be taken into account that the speed of sound and the equation of state in the different phases may vary. Hence, a standard D3Q19 model is clearly not appropriate for multiphase flows. A variable speed of sound requires a larger set of discrete velocities. The generalized lattice gas model is a promising starting-point for the development of new methods that treat the thermodynamics of interfaces in a consistent way. Apart from droplets, this could be combined with electrokinetic methods to study the behavior of jets and sprays.

Finally, a future direction that encompasses several of the above mentioned aspects is the simulation of deformable particle suspensions. Deformable particles are omnipresent in biological systems, the most prominent examples being vesicles and cells. In contrast to rigid colloids, vesicles and cells make it necessary to treat their intrinsic deformation mechanics. The coupling of the elastic energy of the deformable particles and the viscous dissipation in the fluid can lead to viscoelastic behavior of the suspension, which in turn may give rise to non-Newtonian rheological effects. An important example where these effects are highly relevant is blood, the flow behavior of which is strongly determined by the deformability of the red blood cells. The flow of blood through microvessels is subject to migration and aggregation mechanisms of the cells which cause a shear-thinning behavior of blood. The microscopic mechanisms underlying the blood flow are under vivid investigation, and an efficient simulation method could help to enhance the scientific progress. For the treatment of deformable particles with the lattice Boltzmann method, a combination of force-coupling schemes and boundary conditions seems a promising approach which has so far not been explored much.

These examples are, of course, far from being exhaustive. Nevertheless, they give an impression of what might be done with lattice Boltzmann simulations and why method development for computer simulations continues to be an interesting and challenging research topic. Altogether, it can be said that the road is paved with exciting applications for the lattice Boltzmann method.

A Implementation of the lattice Boltzmann method

In this appendix, the lattice Boltzmann implementation in the ESPResSo software package [3, 4] is described. The implementation features a D3Q19 model with a MRT collision operator and thermal fluctuations. Molecular dynamics particles can be coupled to the LB fluid by the method of Ahlrichs and Dünweg [85]. Various boundary conditions are implemented as well. The algorithms are parallelized using the message passing interface MPI [173].

A.1 Usage in ESPResSo

In the ESPResSo software package, every simulation is a sequence of TCL commands. The lattice Boltzmann method can be used with the command `lbfluid` which has the following syntax:

```
lbfluid (<lbvariable> <value>)+
```

The possible choices for `<lbvariable>` are listed in table A.1. The parameter values `<value>` have to be given in the MD unit system used throughout the TCL script, which typically means Lennard-Jones units. Although any variable can be set individually, it is recommended to use the `lbfluid` command in the form

```
lbfluid  grid <grid> tau <tau> dens <dens> visc <viscosity>
```

where abbreviated forms of the variable names can be used. The coupling of MD particles to the LB fluid can be switched on separately by

LB variable	Description
`grid`	lattice spacing a
`tau`	lattice Boltzmann time step τ
`density`	average density ρ of the fluid
`viscosity`	kinematic viscosity ν of the fluid
`friction`	friction coefficient ζ_{bare} for the fluid-particle coupling
`ext_force`	external volumetric force $\mathbf{g}$ driving the fluid flow

Table A.1: ESPResSo variables for the various parameters of the lattice Boltzmann model. They can be set with the `lbfluid` command.

```
lbfluid friction <friction>
```

where `<friction>` is the bare friction coefficient for the Stokes drag force.

The fluctuating lattice Boltzmann model is implemented as a thermostat. Thermal fluctuations can be switched on with the command

```
thermostat lb <temp>
```

where `<temp>` is the desired temperature. The `thermostat` command sets the temperature for both the fluctuations of the LB populations and the random noise balancing the friction force.

Boundaries are implemented as constraints and can be defined with the `constraint` command (see the ESPResSo documentation for details). So far, only wall constraints are supported by the lattice Boltzmann implementation. The type of lattice Boltzmann boundary conditions can be chosen with the command

```
lbboundaries <type> <parameters>*
```

The possible `<type>`s that have been implemented so far comprise `bounce_back`, `specular_reflection`, `slip_reflection` and `partial_slip`.

A.2 Internal unit conversions

Since ESPResSo is originally a Molecular Dynamics package, the simulation parameters set on the TCL level are typically measured in Lennard-Jones units. We have decided to keep this convention for the lattice Boltzmann parameters. Internally, however, lattice units are used where $a = 1$, $\tau = 1$ and $m_p = 1$. The conversion of the unit systems is done in the initialization routines. This has the advantage that all conversions are collected at a central place and need to be executed only when parameters change. The only point where a unit conversion has to be done on-the-fly is the calculation of the fluid-particle coupling. The recipe for unit conversions is straightforward: A quantity that has dimensions $(\text{mass})^k(\text{length})^l(\text{time})^m$ is transformed from MD to LB units through division by ${m_p}^k a^l \tau^m$ where the respective values of m_p, a and τ in the MD unit system are used. The quantities in lattice units can then be used in the equations where all occurrences of m_p, a and τ have been dropped. The reverse conversion goes along the same lines.

A.3 The lattice Boltzmann kernel

In the basic lattice Boltzmann algorithm, the following steps have to be performed in each time-step update $t \to t+1$:

1. Calculation of the local moments m_k from the actual populations f_i according to
$$m_k = \sum_i f_i \hat{e}_{ki}.$$
2. Calculation of the equilibrium moments m_k^{eq} from the mass density $\rho = m_0$ and the momentum densities $j_x = m_1$, $j_y = m_2$ and $j_z = m_3$.
3. Relaxation of the moments towards their equilibrium value
$$m_k^* = m_k^{\text{eq}} + (1+\lambda_k)(m_k - m_k^{\text{eq}}).$$
4. Back-transformation from the moments to the populations according to
$$f_i^* = w_i \sum_k b_k^{-1} m_k^* \hat{e}_{ki}.$$
5. Propagation of the post-collisional populations $f_i^*(\mathbf{r}, t)$ along the lattice links $\hat{\mathbf{c}}_i$ to the new populations $f_i(\mathbf{r} + \hat{\mathbf{c}}_i, t+1)$.

The steps one to four constitute the collision phase, while the fifth step is the streaming step. By counting the number of arithmetic operations, it is found that the collision phase is the computationally more intensive part. The streaming step just shifts data in memory. On modern computing hardware, however, the last step turns out to be the most time consuming because floating point operations can be executed very fast while memory access is limited by bandwidth and latency.

A.3.1 Naive implementation

The above algorithm is straightforward to implement. A crucial observation is that the collision phase in steps one to four is completely local and can be executed independently for all cells whereas the propagation step is non-local and replaces the old populations with the new ones. The simplest possibility for the implementation is the following: One uses two separate sweeps for collisions and streaming, each consisting of three nested loops for the three spatial directions. In the first sweep, the collisions are executed and in the second sweep, the post-collisional populations are loaded and written back to the shifted position. The streaming step has to be done in a carefully chosen order to not overwrite any relevant data. At the boundaries of the domain, an additional halo layer is needed to implement periodic boundary conditions and for the parallelization, respectively. This halo layer has to be filled with the periodic images of the populations at the beginning of the streaming step. The naive implementation could look like in the following code snippet.

```
for (x = 1; x <= n_grid[0]; x++) {
  for (y = 1; y <= n_grid[1]; y++) {
    for (z = 1; z <= n_grid[2]; z++ { /* collisions */

      double moments[19], m_eq[19];

      lb_calc_moments(x,y,z,f,moments);
      lb_calc_equilibrium(x,y,z,moments,m_eq);
      lb_relaxation(x,y,z,moments,m_eq);
      lb_calc_f(x,y,z,f,moments);

    }
  }
}

lb_halo_update();

for (x = n_grid[0]; x >=0 ; x--) {
  for (y = n_grid[1]; y >= 0; y--) {
    for (z = m_grid[2]; z >= 0; z--) { /* streaming 'upwards' in memory */

      lb_stream_up(x,y,z,f);

    }
  }
}

for (x=1; x <= n_grid[0]+1; x++) {
  for (y=1; y <= n_grid[1]+1; y++) {
    for (z=1; z <= n_grid[2]+1; z++ { /* streaming 'downwards' in memory */

      lb_stream_down(x,y,z,f);

    }
  }
}
```

A.3.2 Combined collisions and streaming

Closer inspection of the memory access pattern in the naive implementation reveals that every population is read and written twice, once in the collision phase and once in the streaming phase. On modern computing hardware, where memory access is a bottleneck, this limits the performance of the implementation. The situation can be improved by reducing the number of data transfers from and to memory, which is achieved by combining collisions and streaming in one loop [174, 175]. This leads to either the "pull" scheme or the "push" scheme, both of which are illustrated in figure A.1. In the pull scheme, the propagation is realized first while in the push scheme, the propagation is the last step after the collisions. In order not to overwrite any relevant data, the new populations are stored in a separate array and after each time step the roles of the two arrays are interchanged. An implementation with combined collisions and streaming could look like in the following code snippet.

```
lb_halo_update();

for (x = 1; x <= n_grid[0]; x++) {
  for (y = 1; y <= n_grid[1]; y++) {
    for (z = 1; z <= n_grid[2]; z++ {
```

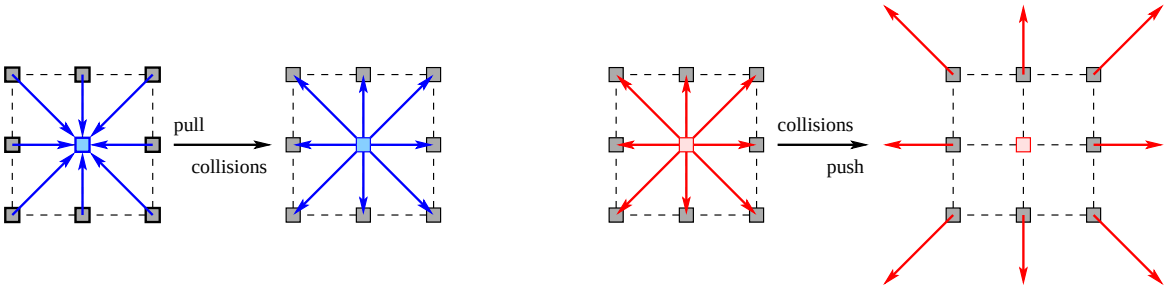


Figure A.1: Illustration of the pull scheme (stream-collide) and the push scheme (collide-stream). In the pull scheme, the populations are read from the neighbor sites for the collision process, while in the push scheme, they are written to the neighbor sites after the collisions.

```
      double moments[19], m_eq[19];

#ifdef PUSH

      lb_calc_moments(x,y,z,f[0],moments);
      lb_calc_equilibrium(x,y,z,moments,m_eq);
      lb_relaxation(x,y,z,moments,m_eq);
      lb_calc_f_and_push(x,y,z,moments,f[1]);

#endif

#ifdef PULL

     lb_pull_f_and_calc_moments(x,y,z,f[0],moments);
     lb_calc_equilibrium(x,y,z,moments,m_eq);
     lb_relaxation(x,y,z,moments,m_eq);
     lb_calc_f(x,y,z,moments,f[[1]);

#endif

    }
  }
}

swap_pointers(f[0],f[1]);
```

A.3.3 Data layout optimization

In the combined implementation every population is read and written only once in each time step. Although this seems already optimal, the performance can yet depend critically on the actual data layout in memory.

First, it has to be noted that every array is a sequential data structure in memory, no matter how many logical dimensions there are. In other words, any array is physically one-dimensional in memory. Consequently, the three nested loops over the three spatial directions boil down to a loop over one-dimensional memory locations. The crucial point is how the logical dimensions are mapped to the physical memory locations and in which order the sequential data in memory is accessed. This is especially important on cache-based archi-

tectures where non-consecutive data access is connected to severe performance impairment. The reason is the way in that data is transferred from main memory to the caches.

Cache memories are intermediate memories that are located on the processor chip and provide high bandwidth and low latency at the expense of being much smaller than the main memory. They are organized in cache lines that are always fetched from or stored to main memory as a whole contiguous block of data. Since not the whole main memory fits into cache, cache lines have to be replaced frequently during the course of the program. The aim is to use the cache lines as efficient as possible, that is, accessing data should lead to a minimum number of fetch and store operations for the same cache line. Optimally, all entries of a cache line are used once it resides in the cache memory. This can be achieved by organizing the data layout such that consecutively used data is contiguous in memory and can be grouped into the same cache line.

The lattice Boltzmann populations can be stored in a five dimensional array with three indices x, y and z for the spatial dimensions, one index i for the velocity direction, and one index $t \in \{0, 1\}$ for the double buffering. The mapping between the five-dimensional array and the linear memory layout is defined by the order of the indices. The index t is typically the "slowest" index.[1] Intuitively, one would group together the 19 velocities of each lattice site which is achieved when the index i is the "fastest". This yields an array of structures, or the so-called *collision optimized* data layout [174] which is schematically depicted in figure A.2. During the collision phase, the populations $f_{0:18}(x, y, z)$ have to be fetched from memory which typically involves two cache lines of 16 doubles. The populations $f_{0:12}(x, y, z+1)$ are then automatically resident in the cache and can be used in the next iteration. However, in the propagation phase the populations are stored to non-contiguous locations and, in the worst case, 19 cache lines are stored of which only one entry was modified. This is a rather inefficient access pattern and, in addition, it is likely to generate many cache misses, especially if the cache size is small. The collision optimized data layout is therefore sub-optimal on many hardware architectures [174].

A better performance is achieved with structures of arrays, or the so-called *propagation optimized* data layout (see figure A.1). It groups the populations for one velocity direction together, i.e., the index i is the "next slowest" following the index t. The index z is now the "fastest" and should be iterated in the innermost loop. As the populations of a local lattice site are not contiguous in memory, the collision phase now requires 19 cache lines to be loaded. The same holds for the propagation phase, where also 19 cache lines are accessed. Given that the cache can keep at least 38 lines resident at the same time, the cached data can be reused in successive iterations over z. In this way, all cache entries are used once they have been loaded from main memory, or vice versa, they are modified before they are

[1] Since the order of indexing has different semantics in different programming languages, the terms "slow" and "fast" are used to describe the position of the indices. "Slowest" means that all other indices run through their whole range before the "slowest" index is increased by one. Thus, the "slowest" index corresponds to the first array index in C's row major order, while it corresponds to the last index in Fortran's column major order. In this terminology, the "fastest" index always addresses consecutive memory locations.

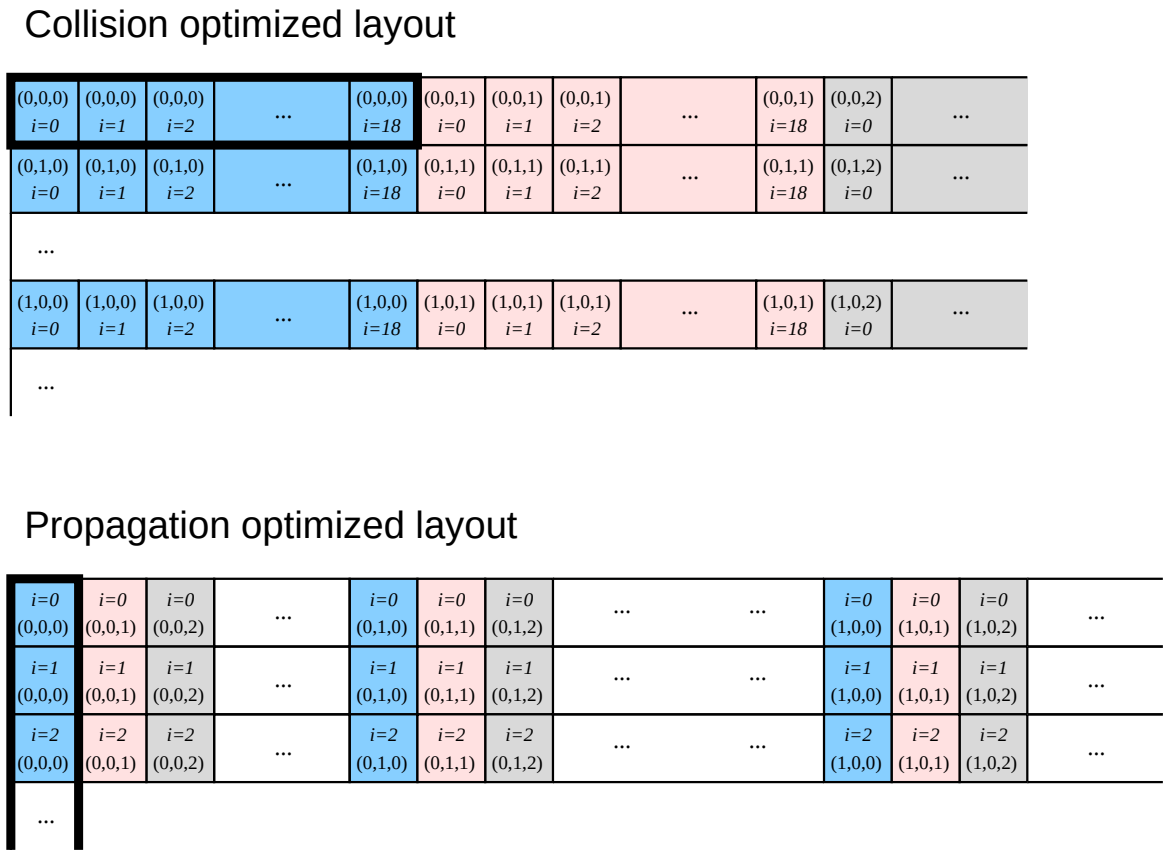


Figure A.2: Illustration of the collision optimized (top) and the propagation optimized data layout (bottom). Colors are used to distinguish different lattice sites. The black box indicates the populations that have to be fetched for the collision step. In the propagation optimized data layout, more cache lines have to be loaded in the collision phase, but they can be exploited during the propagation phase. In practice, the propagation optimized data layout is superior on many computer architectures.

stored back. The propagation optimized data layout is hence much more efficient and shows superior performance on many common hardware architectures [174].

A subtle problem with the propagation optimized data layout is that cache thrashing can occur if the "fastest" array dimension z is a power of two. This is because associative caches map physical memory locations to specific cache locations. With a power of two in the "fastest" array dimension, subsequent z-planes are mapped to the same cache address which causes many cache misses. The effect is a severe performance breakdown. A power of two in the z dimension should therefore be avoided, for example by array padding.

Further optimizations with respect to the memory access pattern are possible. For example one could use blocking techniques or loop splitting. However, these techniques are highly specific and tend to make the program code less extensible. They shall therefore not be discussed further here.

The lattice Boltzmann kernel of ESPResSo uses the propagation optimized data layout. The memory for the lattice Boltzmann populations is allocated during the initialization.

```
static void lb_realloc_fluid() {
  int i;

  lbfluid[0]    = realloc(*lbfluid,2*lbmodel.n_veloc*sizeof(double *));
  lbfluid[0][0] = realloc(**lbfluid,
                          2*lblattice.halo_grid_volume*lbmodel.n_veloc*sizeof(double));
  lbfluid[1]    = (double **)lbfluid[0] + lbmodel.n_veloc;
  lbfluid[1][0] = (double *)lbfluid[0][0] + lblattice.halo_grid_volume*lbmodel.n_veloc;

  for (i=0; i<lbmodel.n_veloc; ++i) {
    lbfluid[0][i] = lbfluid[0][0] + i*lblattice.halo_grid_volume;
    lbfluid[1][i] = lbfluid[1][0] + i*lblattice.halo_grid_volume;
  }

}
```

The first index of the array `lbfluid` is the index t for the double buffering. The second index is the index i for the velocities. `lbfluid[t][i]` is the spatial array for the velocity direction i. It is addressed with a single index which can be calculated from the spatial coordinates x, y and z via

```
index = get_linear_index(x,y,z,lblattice.halo_grid);
```

After a sweep through the whole lattice, the pointers for the source and destination arrays are swapped with

```
double **tmp;
tmp = lbfluid[0];
lbfluid[0] = lbfluid[1];
lbfluid[1] = tmp;
```

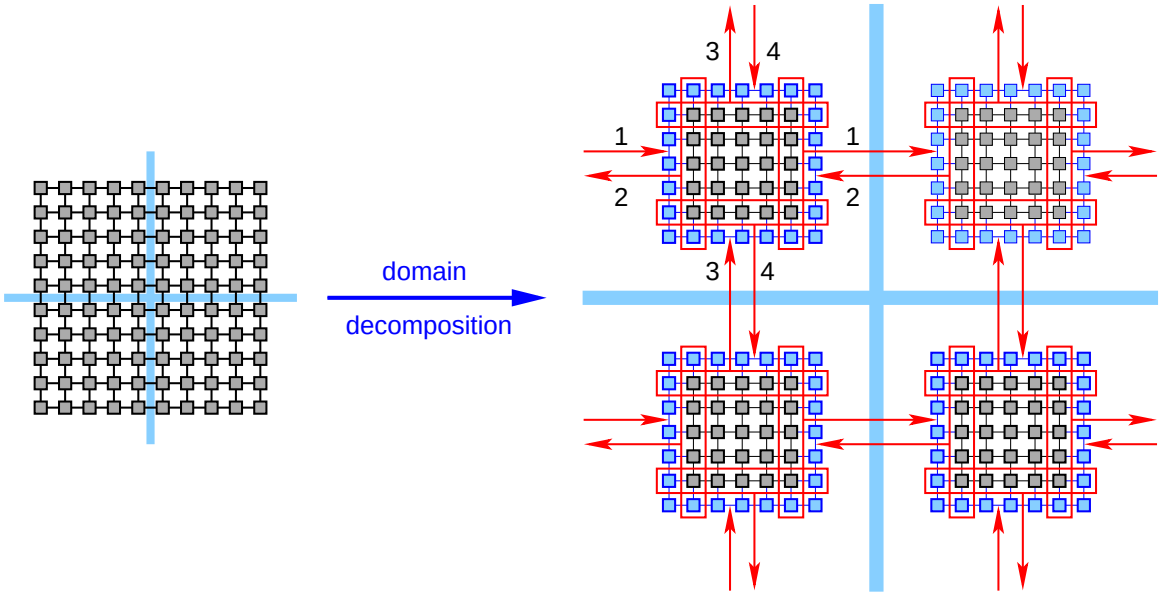


Figure A.3: Schematic illustration of the domain decomposition scheme used for parallelizing the lattice Boltzmann kernel. On each processor, the physical domain is surrounded by a halo region that serves to communicate the populations that cross the domain boundaries to the neighbor processors.

Accessing the populations is straightforward: the population $f_i(x, y, z)$ is read in the collision phase with `lbfluid[0][i][index]`, and in the streaming phase the population $f_i(x + c_{ix}, y + c_{iy}, z + c_{iz})$ is written with `lbfluid[1][i][index+shift]`, where `shift` is calculated from c_{ix}, c_{iy} and c_{iz}. The algorithm is basically independent of the data layout. The crucial point is the appropriate initialization of the array `lbfluid`, which has to be done in a way that leaves access to the populations transparent.

A.4 Parallelization

Parallelization can be a tricky issue and is certainly one of the most error-prone parts of any simulation software. The lattice Boltzmann method is comparatively easy to parallelize due to the specific structure of the update rule. The regular lattice suggests to use a domain decomposition scheme where the whole simulation box is divided into smaller rectangular subdomains. Each subdomain is taken care of by an individual processor. While the collision phase can be executed individually on every processor, the streaming phase requires the exchange of data between different processors. In ESPResSo, the message passing interface MPI is used for communication between the different processes. In order to facilitate the inter-processor data-exchange, an artificial halo region is introduced around the physical domain of each process, cf. figure A.3. The halo region contains the populations that leave or enter the local physical domain to or from another processor's domain. It has to be communicated in every time step, either before the collision loop over the internal lattice sites (pull scheme), or after the collision loop (push scheme). The halo is divided into planes that

direction	send populations	receive populations
positive x	f_1, f_7, f_9, f_{11}, f_{13}	f_2, f_8, f_{10}, f_{12}, f_{14}
negative x	f_2, f_8, f_{10}, f_{12}, f_{14}	f_1, f_7, f_9, f_{11}, f_{13}
positive y	f_3, f_7, f_{10}, f_{15}, f_{17}	f_4, f_8, f_9, f_{16}, f_{18}
negative y	f_4, f_8, f_9, f_{16}, f_{18}	f_3, f_7, f_{10}, f_{15}, f_{17}
positive z	f_5, f_{11}, f_{14}, f_{15}, f_{18}	f_6, f_{12}, f_{13}, f_{16}, f_{17}
negative z	f_6, f_{12}, f_{13}, f_{16}, f_{17}	f_5, f_{11}, f_{14}, f_{15}, f_{18}

Table A.2: The populations that have to be communicated between processes in the respective directions. In each of the six necessary communications, five populations are sent and five populations are received.

have to be communicated to the respective neighbor processes in the three spatial directions. In the domain decomposition scheme, the structure of the memory areas and the processes between which they are exchanged remain static. This information can hence be stored in a C-`struct`, the `HaloCommunicator`, which is set up during the initialization. It contains the information about the six different communications (two per spatial direction) and the respective halo planes. The structure of the data is represented using MPI datatypes. While an x-plane is contiguous in memory, the y- and z-planes are strided. By defining appropriate MPI datatypes, the exact layout is hidden in an abstraction layer. The communication procedure can then be implemented generically by exploiting the information stored in the `HaloCommunicator`. The actual data transfer is handled by the call

```
MPI_Sendrecv(s_buffer, 1, datatype, r_node, REQ_HALO_SPREAD,
             r_buffer, 1, datatype, s_node, REQ_HALO_SPREAD,
             MPI_COMM_WORLD, &status);
```

in the function `void halo_communication(HaloCommunicator *hc, void *base)`. The benefit of this abstract parallelization scheme is that the details of the implementation can be hidden in separate functions. Hence, in the lattice Boltzmann kernel, a single call to `halo_communcation` at the right place is sufficient. In this way, the algorithm and the parallelization are clearly separated, which reduces the risk of programming errors considerably and makes the code much more readable. Nonetheless, the parallelization scheme can be easily extended if necessary.

In practice, there is still potential for improvement. One point is the observation that only a subset of all velocity directions can cross the domain boundary in a given direction, namely those five with a positive projection on that direction. Hence, there is no need to communicate all 19 populations of the halo plane but it is sufficient to send the five populations that leave the domain and receive the five that enter. Table A.2 lists the populations that have to be communicated in the different directions. The set of populations that is sent to the right neighbor and the set received from the left neighbor is always the same. It is therefore convenient to combine the sent-to-right and receive-from-left into a single `MPI_Sendrecv` call. The optimized communication routine for the x-direction, for example, could look like the following:

```
/***************
 * X direction *
 ***************/
count = 5*lblattice.halo_grid[1]*lblattice.halo_grid[2];
sbuf = malloc(count*sizeof(double));
rbuf = malloc(count*sizeof(double));

/* send to right, recv from left i = 1, 7, 9, 11, 13 */
snode = node_neighbors[0];
rnode = node_neighbors[1];

buffer = sbuf;
index = get_linear_index(lblattice.grid[0]+1,0,0,lblattice.halo_grid);
for (z=0; z<lblattice.halo_grid[2]; z++) {
  for (y=0; y<lblattice.halo_grid[1]; y++) {

    buffer[0] = lbfluid[1][1][index];
    buffer[1] = lbfluid[1][7][index];
    buffer[2] = lbfluid[1][9][index];
    buffer[3] = lbfluid[1][11][index];
    buffer[4] = lbfluid[1][13][index];
    buffer += 5;

    index += yperiod;
  }
}

if (node_grid[0] > 1) {
  MPI_Sendrecv(sbuf, count, MPI_DOUBLE, snode, REQ_HALO_SPREAD,
        rbuf, count, MPI_DOUBLE, rnode, REQ_HALO_SPREAD,
        MPI_COMM_WORLD, &status);
} else {
  memcpy(rbuf,sbuf,count*sizeof(double));
}

buffer = rbuf;
index = get_linear_index(1,0,0,lblattice.halo_grid);
for (z=0; z<lblattice.halo_grid[2]; z++) {
  for (y=0; y<lblattice.halo_grid[1]; y++) {

    lbfluid[1][1][index] = buffer[0];
    lbfluid[1][7][index] = buffer[1];
    lbfluid[1][9][index] = buffer[2];
    lbfluid[1][11][index] = buffer[3];
    lbfluid[1][13][index] = buffer[4];
    buffer += 5;

    index += yperiod;
  }
}

/* send to left, recv from right i = 2, 8, 10, 12, 14 */
snode = node_neighbors[1];
rnode = node_neighbors[0];

buffer = sbuf;
index = get_linear_index(0,0,0,lblattice.halo_grid);
for (z=0; z<lblattice.halo_grid[2]; z++) {
  for (y=0; y<lblattice.halo_grid[1]; y++) {

    buffer[0] = lbfluid[1][2][index];
    buffer[1] = lbfluid[1][8][index];
    buffer[2] = lbfluid[1][10][index];
    buffer[3] = lbfluid[1][12][index];
    buffer[4] = lbfluid[1][14][index];
    buffer += 5;
```

```
    index += yperiod;
  }
}

if (node_grid[0] > 1) {
  MPI_Sendrecv(sbuf, count, MPI_DOUBLE, snode, REQ_HALO_SPREAD,
        rbuf, count, MPI_DOUBLE, rnode, REQ_HALO_SPREAD,
        MPI_COMM_WORLD, &status);
} else {
  memcpy(rbuf,sbuf,count*sizeof(double));
}

buffer = rbuf;
index = get_linear_index(lblattice.grid[0],0,0,lblattice.halo_grid);
for (z=0; z<lblattice.halo_grid[2]; z++) {
  for (y=0; y<lblattice.halo_grid[1]; y++) {

    lbfluid[1][2][index] = buffer[0];
    lbfluid[1][8][index] = buffer[1];
    lbfluid[1][10][index] = buffer[2];
    lbfluid[1][12][index] = buffer[3];
    lbfluid[1][14][index] = buffer[4];
    buffer += 5;

    index += yperiod;
  }
}

free(rbuf);
free(sbuf);
```

The above version is for the push scheme. It can easily be adopted to the pull scheme with minimal modifications. In this example, we have not used the MPI datatypes and instead, the data is packed into buffers manually. The scheme uses the minimal number of communications between processes (at least in case the neighbors in all directions are different processes) and no overhead data is transferred. It is in this sense an optimal scheme. Further optimizations might be possible with respect to the buffering strategy and for special process topologies. The latter cases are so far not addressed in the ESPResSo implementation.

A.5 Thermal fluctuations

The implementation of thermal fluctuations is straightforward. The amplitudes of the random fluctuations are calculated from the relaxation parameters during the initialization:

```
for (i=0; i<3; i++) lb_phi[i] = 0.0;
lb_phi[4] = sqrt(mu*e[19][4]*(1.-SQR(gamma_bulk)));
for (i=5; i<10; i++) lb_phi[i] = sqrt(mu*e[19][i]*(1.-SQR(gamma_shear)));
for (i=10; i<n_veloc; i++) lb_phi[i] = sqrt(mu*e[19][i]);
```

The random fluctuations are added to the modes after the relaxation.

```
double rootrho = sqrt(mode[0]+lbpar.rho);

/* stress modes */
mode[4] += rootrho*lb_phi[4]*gaussian_random();
mode[5] += rootrho*lb_phi[5]*gaussian_random();
mode[6] += rootrho*lb_phi[6]*gaussian_random();
mode[7] += rootrho*lb_phi[7]*gaussian_random();
mode[8] += rootrho*lb_phi[8]*gaussian_random();
mode[9] += rootrho*lb_phi[9]*gaussian_random();

/* ghost modes */
mode[10] += rootrho*lb_phi[10]*gaussian_random();
mode[11] += rootrho*lb_phi[11]*gaussian_random();
mode[12] += rootrho*lb_phi[12]*gaussian_random();
mode[13] += rootrho*lb_phi[13]*gaussian_random();
mode[14] += rootrho*lb_phi[14]*gaussian_random();
mode[15] += rootrho*lb_phi[15]*gaussian_random();
mode[16] += rootrho*lb_phi[16]*gaussian_random();
mode[17] += rootrho*lb_phi[17]*gaussian_random();
mode[18] += rootrho*lb_phi[18]*gaussian_random();
```

Note that both the stress modes and the kinetic modes have random fluctuations. For each fluctuating mode, a Gaussian random number has to be drawn. This is done in the function `double gaussian_random()` which implements a simple Box-Muller transformation [176].

A.6 Force coupling

The lattice Boltzmann implementation in ESPResSo includes the coupling of MD particles (polymers, colloids, etc.) to the LB fluid. For a theoretical description of the method, the reader is referred to the original publication of Ahlrichs and Dünweg [86]. The calculation of the coupling force is wrapped in the function `void calc_particle_lattice_ia()`. First, the random numbers for the fluctuating part of the force are pre-drawn for all particles and communicated.

```
/* draw random numbers for local particles */
for (c=0;c<local_cells.n;c++) {
  cell = local_cells.cell[c] ;
  p = cell->part ;
  np = cell->n ;
  for (i=0;i<np;i++) {
    p[i].lc.f_random[0] = lb_coupl_pref*gaussian_random();
    p[i].lc.f_random[1] = lb_coupl_pref*gaussian_random();
    p[i].lc.f_random[2] = lb_coupl_pref*gaussian_random();
  }
}

/* communicate the random numbers */
ghost_communicator(&cell_structure.ghost_lbcoupling_comm) ;
```

This is necessary to make sure that ghost particles use the same random number as their real counterparts. The synchronization of random numbers can lead to severe complications when strict positivity of the populations is required. The reason is that the order of ghost

and real particles is interchanged between neighboring processes. Consequently, a random force does not necessarily lead to negative populations on both processes. The revocation of a "bad" random number can therefore be triggered on different processes, which makes it very complicated to synchronize the redrawing of random numbers. In the worst case, the program can get stuck in a loop just redrawing random numbers. Therefore, the strict positivity of the populations is waived and small negative values are allowed. This should be a rare event if the simulation parameters have physically reasonable values.

The viscous drag force exerted by the fluid on the particle is calculated in the function `void lb_viscous_coupling(Particle *p, double force[3])`. First, the interpolated fluid velocity at the particle's position is determined.

```
/* calculate fluid velocity at particle's position
   this is done by linear interpolation
   (Eq. (11) Ahlrichs and Duenweg, JCP 111(17):8225 (1999)) */
interpolated_u[0] = interpolated_u[1] = interpolated_u[2] = 0.0 ;
for (z=0;z<2;z++) {
  for (y=0;y<2;y++) {
    for (x=0;x<2;x++) {

      local_node = &lbfields[node_index[(z*2+y)*2+x]];

      if (local_node->recalc_fields) {
        lb_calc_local_fields(node_index[(z*2+y)*2+x],local_node->rho,local_node->j,NULL);
        local_node->recalc_fields = 0;
        local_node->has_force = 1;
      }

      local_rho[0] = local_node->rho[0];
      local_j[0] = local_node->j[0];
      local_j[1] = local_node->j[1];
      local_j[2] = local_node->j[2];

      interpolated_u[0] += delta[3*x+0]*delta[3*y+1]*delta[3*z+2]*local_j[0]/(*local_rho);
      interpolated_u[1] += delta[3*x+0]*delta[3*y+1]*delta[3*z+2]*local_j[1]/(*local_rho);
      interpolated_u[2] += delta[3*x+0]*delta[3*y+1]*delta[3*z+2]*local_j[2]/(*local_rho);

    }
  }
}
```

Then the viscous drag is calculated and added to the random force.

```
/* calculate viscous force
 * take care to rescale velocities with time_step and transform to MD units
 * (Eq. (9) Ahlrichs and Duenweg, JCP 111(17):8225 (1999)) */
force[0] = - lbpar.friction * (p->m.v[0]/time_step - interpolated_u[0]*agrid/tau);
force[1] = - lbpar.friction * (p->m.v[1]/time_step - interpolated_u[1]*agrid/tau);
force[2] = - lbpar.friction * (p->m.v[2]/time_step - interpolated_u[2]*agrid/tau);

force[0] = force[0] + p->lc.f_random[0];
force[1] = force[1] + p->lc.f_random[1];
force[2] = force[2] + p->lc.f_random[2];
```

In the next step, the force is transformed to the momentum transfer in lattice units, cf. Equation (12) of [86].

```
/* transform momentum transfer to lattice units
   (Eq. (12) Ahlrichs and Duenweg, JCP 111(17):8225 (1999)) */
delta_j[0] = - force[0]*integrate_pref2/time_step*tau/agrid;
delta_j[1] = - force[1]*integrate_pref2/time_step*tau/agrid;
delta_j[2] = - force[2]*integrate_pref2/time_step*tau/agrid;
```

Finally, the momentum transfer is extrapolated to the lattice sites and added to the volumetric force acting on the fluid at that site.

```
for (z=0;z<2;z++) {
  for (y=0;y<2;y++) {
    for (x=0;x<2;x++) {

      local_f = lbfields[node_index[(z*2+y)*2+x]].force;

      local_f[0] += delta[3*x+0]*delta[3*y+1]*delta[3*z+2]*delta_j[0];
      local_f[1] += delta[3*x+0]*delta[3*y+1]*delta[3*z+2]*delta_j[1];
      local_f[2] += delta[3*x+0]*delta[3*y+1]*delta[3*z+2]*delta_j[2];

    }
  }
}
```

The force is applied in the function `void lb_apply_forces(index_t index, double* mode)` which is called from within the main collide-stream loop. It has to be mentioned that this implementation is not fully self-consistent, because the force enters the redefined hydrodynamic momentum density which itself determines the force. To deal with this problem one could use an iterative scheme. However, since the linear interpolation of the fluid velocity is only a first approximation, an iterative scheme is probably not very beneficial in view of the computational overhead.

A.7 Boundary conditions

The implementation of boundary conditions depends heavily on the specific type of boundary conditions, i.e., node based or link based boundary conditions. In ESPResSo, both types have been implemented. The link based implementation comprises bounce-back, specular reflections and slip-reflection. Link based boundary conditions affect the streaming of the populations at the boundary. Instead of following their velocity link, they are deflected to a different direction. This could in principle be implemented within the streaming step by redirecting the copy operation of the population to the correct target node. However, additional `if` statements would have to be introduced in the main loop which would deteriorate the efficiency. Furthermore, there is a danger to spoil the cache optimized data layout. Therefore, the streaming step is left unchanged such that the populations end up on nodes outside the fluid domain. They are then moved to the correct target node in a separate boundary loop. This works for any link based scheme. The specific steps for the different boundary conditions are outlined in the following.

A.7.1 Bounce-back

In the bounce-back rule, the populations that have moved out of the fluid domain have to be copied back to the reversed velocity direction on the node they came from. The index shift to the bounce-back node is pre-calculated and stored in the array `next`.

```
int yperiod = lblattice.halo_grid[0];
int zperiod = lblattice.halo_grid[0]*lblattice.halo_grid[1];
int next[19];
next[0]  =   0;                       // ( 0, 0, 0) =
next[1]  =   1;                       // ( 1, 0, 0) +
next[2]  = - 1;                       // (-1, 0, 0)
next[3]  =   yperiod;                 // ( 0, 1, 0) +
next[4]  = - yperiod;                 // ( 0,-1, 0)
next[5]  =   zperiod;                 // ( 0, 0, 1) +
next[6]  = - zperiod;                 // ( 0, 0,-1)
next[7]  =   (1+yperiod);             // ( 1, 1, 0) +
next[8]  = - (1+yperiod);             // (-1,-1, 0)
next[9]  =   (1-yperiod);             // ( 1,-1, 0)
next[10] = - (1-yperiod);             // (-1, 1, 0) +
next[11] =   (1+zperiod);             // ( 1, 0, 1) +
next[12] = - (1+zperiod);             // (-1, 0,-1)
next[13] =   (1-zperiod);             // ( 1, 0,-1)
next[14] = - (1-zperiod);             // (-1, 0, 1) +
next[15] =   (yperiod+zperiod);       // ( 0, 1, 1) +
next[16] = - (yperiod+zperiod);       // ( 0,-1,-1)
next[17] =   (yperiod-zperiod);       // ( 0, 1,-1)
next[18] = - (yperiod-zperiod);       // ( 0,-1, 1) +
```

The reversion of the velocity is achieved by using the following index map

```
int reverse[] = { 0, 2, 1, 4, 3, 6, 5, 8, 7, 10, 9, 12, 11, 14, 13, 16, 15, 18, 17 };
```

The bounce-back operation at the top wall then looks like

```
/* bottom-up sweep */
for (k=lblattice.halo_offset;k<lblattice.halo_grid_volume;k++) {

  if (lbfields[k].boundary) {

    /* bounce back to lower indices */
    lbfluid[1][reverse[5]][k-next[5]]   = lbfluid[1][5][k];
    lbfluid[1][reverse[11]][k-next[11]] = lbfluid[1][11][k];
    lbfluid[1][reverse[14]][k-next[14]] = lbfluid[1][14][k];
    lbfluid[1][reverse[15]][k-next[15]] = lbfluid[1][15][k];
    lbfluid[1][reverse[18]][k-next[18]] = lbfluid[1][18][k];

  }

}
```

The implementation for the bottom wall works analogously.

A.7.2 Specular reflections

Specular reflections differ from the bounce-back rule in the target nodes and the velocity mapping. The shift to the target nodes is again stored in an array `next`

```
int zperiod = lblattice.halo_grid[0]*lblattice.halo_grid[1];
int next[19];
next[0]  =   0;                      // ( 0, 0, 0)
next[1]  =   0;                      // ( 1, 0, 0)
next[2]  = - 0;                      // (-1, 0, 0)
next[3]  =   0;                      // ( 0, 1, 0)
next[4]  = - 0;                      // ( 0,-1, 0)
next[5]  =   zperiod;                // ( 0, 0, 1)
next[6]  = - zperiod;                // ( 0, 0,-1)
next[7]  =   0;                      // ( 1, 1, 0)
next[8]  = - 0;                      // (-1,-1, 0)
next[9]  =   0;                      // ( 1,-1, 0)
next[10]  = - 0;                     // (-1, 1, 0)
next[11] =   zperiod;                // ( 1, 0, 1)
next[12] = - zperiod;                // (-1, 0,-1)
next[13] = - zperiod;                // ( 1, 0,-1)
next[14] =   zperiod;                // (-1, 0, 1)
next[15] =   zperiod;                // ( 0, 1, 1)
next[16] = - zperiod;                // ( 0,-1,-1)
next[17] = - zperiod;                // ( 0, 1,-1)
next[18] =   zperiod;                // ( 0,-1, 1)
```

and the index mapping for specular reflections is given by

```
int reflect[] = { 0, 1, 2, 3, 4, 6, 5, 7, 8, 9, 10, 13, 14, 11, 12, 17, 18, 15, 16 };
```

The sweep through the nodes of the top wall is implemented as follows:

```
/* bottom-up sweep */
for (k=lblattice.halo_offset;k<lblattice.halo_grid_volume;k++) {

  if (lbfields[k].boundary) {

    /* reflect to lower indices */
    lbfluid[1][reflect[5]][k-next[5]] = lbfluid[1][5][k];
    lbfluid[1][reflect[11]][k-next[11]] = lbfluid[1][11][k];
    lbfluid[1][reflect[14]][k-next[14]] = lbfluid[1][14][k];
    lbfluid[1][reflect[15]][k-next[15]] = lbfluid[1][15][k];
    lbfluid[1][reflect[18]][k-next[18]] = lbfluid[1][18][k];

  }

}
```

A.7.3 Slip reflections

Slip reflections are a mixture of bounce-back and specular reflections. The implementation is therefore very similar to the above. We just quote the example of the top wall again:

```
double s = lb_boundary_par.slip_pref;
double r = 1.0 - s;

double **n = lbfluid[1];

/* bottom-up sweep */
for (k=lblattice.halo_offset; k<lblattice.halo_grid_volume; k++) {

  if (lbfields[k].boundary) {

    /* slip reflect to lower indices */
    n[6][k-zperiod]  = n[5][k];
    n[12][k-zperiod] = s*n[14][k] + r*n[11][k-zperiod+next[11]];
    n[13][k-zperiod] = s*n[11][k] + r*n[14][k-zperiod+next[14]];
    n[16][k-zperiod] = s*n[18][k] + r*n[15][k-zperiod+next[15]];
    n[17][k-zperiod] = s*n[15][k] + r*n[18][k-zperiod+next[18]];

  }

}
```

A.7.4 Local boundary collisions

In contrast to the link-based boundary conditions, local boundary conditions can be directly integrated into the main loop. Instead of the normal collisions and streaming, the corresponding sequence of boundary functions has to be called:

```
void lb_boundary_collisions(int index, double *modes) {

  double pi[6];

  lb_boundary_calc_modes(index, modes, pi);
  lb_boundary_relax_modes(index, modes, pi);
  lb_boundary_apply_forces(index, modes);
  lb_boundary_calc_n_push(index, modes);

}
```

For the special case of the bounce-back of non-equilibrium parts combined with BGK-relaxation, the whole processing of the boundary collisions is combined in the function `lb_boundary_bb_neq_BGK(index, modes)`. It mimics the same operations as in the bulk, but with modified values for the reduced symmetry at the boundary. For details, the reader is referred to chapter 6 of this thesis.

B Technical material

This appendix collects various technical details which have been skipped in the main text.

B.1 Hermite tensor polynomials and Gauss-Hermite quadratures

B.1.1 Hermite tensor polynomials

The Hermite tensor polynomials as used by Grad [177] are a complete basis set of Hilbert space with respect to the scalar product

$$\langle f|g\rangle = \int \omega(\mathbf{v}) f(\mathbf{v}) g(\mathbf{v})\, d\mathbf{v}. \tag{B.1}$$

The orthonormality relation reads

$$\int \omega(\mathbf{v}) \mathcal{H}^{(n)}_{\boldsymbol{\alpha}}(\mathbf{v}) \mathcal{H}^{(m)}_{\boldsymbol{\beta}}(\mathbf{v})\, d\mathbf{v} = \delta_{mn} \boldsymbol{\delta}^{(n)}_{\boldsymbol{\alpha\beta}}, \tag{B.2}$$

where $\boldsymbol{\delta}^{(n)}_{\boldsymbol{\alpha\beta}} = 1$ if $\boldsymbol{\alpha} = (\alpha_1, \ldots, \alpha_n)$ is a permutation of $\boldsymbol{\beta} = (\beta_1, \ldots, \beta_{m=n})$ and zero otherwise. The weight function associated with the Hermite polynomials is given by

$$\omega(\mathbf{v}) = (2\pi)^{-\frac{D}{2}} \exp\left[-\frac{\mathbf{v}^2}{2}\right]. \tag{B.3}$$

The latter can be used to define the Hermite polynomials explicitly

$$\mathcal{H}^{(n)}_{\alpha_1 \ldots \alpha_n} = \frac{(-1)^n}{\omega(\mathbf{v})} \frac{\partial}{\partial v_{\alpha_1}} \ldots \frac{\partial}{\partial v_{\alpha_n}} \omega(\mathbf{v}). \tag{B.4}$$

The first few polynomials are

$$\begin{aligned}
\mathcal{H}^{(0)}(\mathbf{v}) &= 1, \\
\mathcal{H}^{(1)}_{\alpha}(\mathbf{v}) &= v_\alpha, \\
\mathcal{H}^{(2)}_{\alpha\beta}(\mathbf{v}) &= v_\alpha v_\beta - \delta_{\alpha\beta}, \\
\mathcal{H}^{(3)}_{\alpha\beta\gamma}(\mathbf{v}) &= v_\alpha v_\beta v_\gamma - v_\alpha \delta_{\beta\gamma} - v_\beta \delta_{\alpha\gamma} - v_\gamma \delta_{\alpha\beta}.
\end{aligned} \tag{B.5}$$

The Hermite tensor polynomials satisfy the recurrence relation

$$v_\alpha \mathcal{H}^{(n)}_{\alpha_1 \dots \alpha_n} = \mathcal{H}^{(n+1)}_{\alpha\alpha_1 \dots \alpha_n} + \sum_{k=1}^{n} \delta_{\alpha\alpha_k} \mathcal{H}^{(n-1)}_{\alpha_1 \dots \alpha_{k-1}\alpha_{k+1} \dots \alpha_n}. \tag{B.6}$$

Any square integrable function in the Hilbert space can be expanded in the basis as

$$f(\mathbf{v}) = \omega(\mathbf{v}) \sum_{n=0}^{\infty} \frac{1}{n!} \mathsf{a}^{(n)}_{\boldsymbol{\alpha}} \mathcal{H}^{(n)}_{\boldsymbol{\alpha}}(\mathbf{v}), \tag{B.7}$$

where a contraction over the n-fold index $\boldsymbol{\alpha} = (\alpha_1, \cdots, \alpha_n)$ of the tensors $\mathsf{a}^{(n)}$ and $\mathcal{H}^{(n)}$ is to be understood. Since $\mathcal{H}^{(n)}$ is symmetric in $\boldsymbol{\alpha}$, we will assume that $\mathsf{a}^{(n)}$ is symmetric in $\boldsymbol{\alpha}$ as well. To obtain the expansion coefficients, we multiply by $\mathcal{H}^{(m)}(\mathbf{v})$ and integrate

$$\begin{aligned} \int f(\mathbf{v}) \mathcal{H}^{(m)}_{\boldsymbol{\beta}}(\mathbf{v}) d\mathbf{v} &= \int \omega(\mathbf{v}) \sum_{n=0}^{\infty} \frac{1}{n!} \mathsf{a}^{(n)}_{\boldsymbol{\alpha}} \mathcal{H}^{(n)}_{\boldsymbol{\alpha}}(\mathbf{v}) \mathcal{H}^{(m)}_{\boldsymbol{\beta}}(\mathbf{v}) d\mathbf{v} \\ &= \sum_{n=0}^{\infty} \frac{1}{n!} \mathsf{a}^{(n)}_{\boldsymbol{\alpha}} \, \delta_{mn} \delta_{\boldsymbol{\alpha}\boldsymbol{\beta}} \\ &= \mathsf{a}^{(m)}_{\boldsymbol{\beta}}, \end{aligned} \tag{B.8}$$

where we have exploited that there appear $n!$ permutations of $\boldsymbol{\alpha}$ in the contraction $\mathsf{a}^{(n)}_{\boldsymbol{\alpha}} \delta_{\boldsymbol{\alpha}\boldsymbol{\beta}}$. Thus we obtain the Hermite coefficients as

$$\mathsf{a}^{(n)} = \int f(\mathbf{v}) \mathcal{H}^{(n)}(\mathbf{v}) d\mathbf{v}. \tag{B.9}$$

B.1.2 Gauss-Hermite quadrature

The Gaussian quadrature is a means to approximate the integrals $\int \omega(v) f(v)\, dv$ for a given function $f(v)$ by

$$\int \omega(v) f(v)\, dv \approx \sum_{i=1}^{n} w_i f(c_i), \tag{B.10}$$

where w_i is a set of weights and c_i are called the nodes or abscissae of the quadrature. The aim is to find a choice of n nodes that maximizes the degree of precision m of the approximation, that is, the degree of a polynomial up to which (B.10) holds exactly.

For the weight function (B.3) and the integration interval $(-\infty, \infty)$ in $D = 1$ dimension, the Gauss-Hermite quadrature can be applied [178, 179]. The optimal nodes for an n-point quadrature are the roots of the one-dimensional Hermite polynomials $H_n(v)$ and the weights are given by

$$\begin{aligned} w_i &= \frac{1}{H_{n-1}(c_i) \frac{d}{dv} H_n(c_i)} \int \omega(v) H_{n-1}(v) H_{n-1}(v)\, dv \\ &= \frac{n!}{\left[n H_{n-1}(c_i)\right]^2}, \end{aligned} \tag{B.11}$$

Quadrature	c_i	w_i
$E^1_{1,1}$	0	1
$E^2_{1,3}$	± 1	$1/2$
$E^3_{1,5}$	0	$2/3$
	$\pm\sqrt{3}$	$1/6$
$E^4_{1,7}$	$\pm\sqrt{3-\sqrt{6}}$	$(3+\sqrt{6})/12$
	$\pm\sqrt{3+\sqrt{6}}$	$(3-\sqrt{6})/12$

Table B.1: Nodes and weights of some one-dimensional Gauss-Hermite quadratures.

where the relation $\frac{d}{dv}H_n = vH_n - H_{n+1} = nH_{n-1}$ was used and $\boldsymbol{\delta}^{(n)}_{\boldsymbol{\alpha\alpha}} = n!$ in one dimension. The degree of precision of the n-point quadrature in one dimension is $m = 2n-1$. The nodes and the weights for some one-dimensional Gauss-Hermite quadratures are listed in table B.1. The nomenclature $E^n_{D,m}$ is adopted from Shan et al. [47], where D is the dimension of space, n is number of nodes and m is the degree of precision.

In higher dimensions $D > 1$, there is no unique quadrature procedure available. Nevertheless, one can construct quadratures in higher dimensions from the one-dimensional quadrature by writing

$$\begin{aligned}\int \omega(\mathbf{v}) \prod_{\alpha=1}^{D} v_\alpha^{n_\alpha}\, d\mathbf{v} &= \prod_{\alpha=1}^{D} \left(\int \omega(v_\alpha) v_\alpha^{n_\alpha} dv_\alpha \right) = \prod_{\alpha=1}^{D} \left(\sum_{k_\alpha} w_{k_\alpha} c_{k_\alpha}^{n_\alpha} \right) \\ &= \sum_{k_1} \cdots \sum_{k_D} w_{k_1} \cdots w_{k_D} c_{k_1}^{n_1} \cdots c_{k_D}^{n_D},\end{aligned} \tag{B.12}$$

where $n_1 + \cdots + n_D \leq n$. This means that a D-dimensional quadrature emerges from a combination of D one-dimensional quadratures:

$$\int \omega(\mathbf{v}) p(\mathbf{v})\, d\mathbf{v} = \sum w_{k_1 \ldots k_D} p(\mathbf{c}_{k_1 \ldots k_D}), \tag{B.13}$$

where $w_{k_1 \ldots k_D} = w_{k_1} \cdots w_{k_D}$ and $\mathbf{c}_{k_1 \ldots k_D} = (c_{k_1}, \ldots, c_{k_D})$, and $p(\mathbf{v})$ is a polynomial of degree n. For example, using the one-dimensional quadrature $E^3_{1,5}$, we obtain the quadratures $E^9_{2,5}$ and $E^{27}_{3,5}$ using the full set of abscissae in two and three dimensions, respectively (see table 2.1 on page 18).

In three dimensions, the number of nodes can be reduced without affecting the overall degree of the quadrature. The weights of the quadrature $E^{27}_{3,5}$ can be grouped into four symmetry classes w_q according to $q = \|\mathbf{c}_{k_1 \ldots k_3}\|^2/3$. For $D = 3$ and $n = 5$ we have $n_1 + n_2 + n_3 \leq 5$. We can assume $n_1 \leq n_2 \leq n_3$ without loss of generality. The $n_i \geq 0$ for $i = 1, 2, 3$ implies $n_1 \leq 1$. If $n_1 = 1$, the integral (B.12) vanishes for parity reasons. The sum on the right hand side also vanishes as the weights are symmetric with respect to $c_1 = 0$. Conversely, if $n_1 = 0$, $p(\mathbf{v})$ reduces to a two-dimensional polynomial and the quadrature retains its degree

if the weights of the three-dimensional quadrature match the weights of the two-dimensional quadrature $E_{2,5}^{9}$:

$$\begin{aligned} w_{q=0} + 2w_{q=1} &= \frac{2}{3} \cdot \frac{2}{3} = \frac{4}{9}, \\ w_{q=1} + 2w_{q=2} &= \frac{2}{3} \cdot \frac{1}{6} = \frac{1}{9}, \\ w_{q=2} + 2w_{q=3} &= \frac{1}{6} \cdot \frac{1}{6} = \frac{1}{36}. \end{aligned} \tag{B.14}$$

This is an under-determined system that can be rewritten in the parametric form[1]

$$\begin{aligned} w_{q=0} &= \frac{1}{9}(2+t), \\ w_{q=1} &= \frac{1}{18}(2-t), \\ w_{q=2} &= \frac{1}{36}t, \\ w_{q=3} &= \frac{1}{72}(1-t). \end{aligned} \tag{B.15}$$

The original quadrature $E_{3,5}^{27}$ is recovered for $t = 2/3$. By choosing $t = 0$ or $t = 1$ we can omit either the $q = 2$ or the $q = 3$ symmetry class and thus effectively reduce the number of abscissae. Note that this does not affect the accuracy of the quadrature. The result are the quadratures $E_{3,5}^{15}$ and $E_{3,5}^{19}$ that correspond to the D3Q15 and D3Q19 models, respectively.

Besides the production formulas we shall demonstrate that quadratures can also be constructed from a predefined set of abscissae $\mathbf{c}_i$. We note that the orthonormal relation implies

$$\int \omega(\mathbf{v}) \mathcal{H}^{(n)} \, d\mathbf{v} = \delta_{0,n}. \tag{B.16}$$

The quadrature (B.13) is of degree m, if and only if

$$\sum_{i=1}^{n} w_i \mathcal{H}^{(n)}(\mathbf{c}_i) = \delta_{0,n}. \tag{B.17}$$

For a set of abscissae that obeys parity symmetry, this relation is automatically satisfied for the odd tensor polynomials. The even tensor polynomials up to $n = 4$ yield the following conditions on the weights w_i:

$$\begin{aligned} &\sum_i w_i = 1, \\ &\sum_i w_i c_{i\alpha} c_{i\beta} = \delta_{\alpha\beta}, \\ &\sum_i w_i c_{i\alpha} c_{i\beta} c_{i\gamma} c_{i\delta} = \delta_{\alpha\beta}\delta_{\gamma\delta} + \delta_{\alpha\gamma}\delta_{\beta\delta} + \delta_{\alpha\delta}\delta_{\beta\gamma}. \end{aligned} \tag{B.18}$$

[1] This was previously derived by Shan et al. [47], but equation (A 20) in their paper contains typographic errors.

We note that this is closely related to the conditions on the weights that follow from the physical requirements within the Chapman-Enskog equation. This shows the intimate relationship between the Hermite expansion and the symmetry properties of the velocity set and the lattice sums.

Finally, we remark that quadratures can also be obtained on different routes [179]. The smallest number of nodes among the known quadratures is used by the 13-point quadrature $E_{3,5}^{13}$, which is of degree 5.[2] In general, it is yet an open problem, how the minimal quadrature for a given geometry, weight function and degree of accuracy can be found.

B.2 Lattice sums and isotropic lattice models

The lattice sums

$$T^{(n)} = \sum_i w_i \mathbf{c}_i \dots \mathbf{c}_i \tag{B.19}$$

play a pivotal role in the derivation and analysis of the lattice Boltzmann method. In this section, we treat some important properties of the lattice sums with respect to the isotropy of the lattice model.

B.2.1 Lattice sums for discrete velocity sets

Lattice Boltzmann models are often constructed from a given discrete velocity set. This set then has to satisfy certain symmetry properties in order to guarantee Galilean invariance and isotropy of the macroscopic equations. The requirements can be conveniently formulated in terms of the lattice sums, which have to take the form of certain invariant tensors. Here we derive the general form of the lattice sums for simple cubic lattices.

The n-th rank lattice sum for a given set of discrete velocities is

$$T^{(n)}_{\alpha_1 \dots \alpha_n} = \sum_i w_i c_{i\alpha_1} \dots c_{i\alpha_n}. \tag{B.20}$$

The w_i are the weights from the equilibrium distribution which we consider as free parameters here. For symmetry reasons, w_i depends on the length of $\mathbf{c}_i$ but not its direction. The lattice sum can therefore be split into contributions from the different subshells indexed by $q = \mathbf{c}_i^2$

$$T^{(n)}_{\alpha_1 \dots \alpha_n} = \sum_q w_q \sum_{i=1}^{b_q} c_{qi\alpha_1} \dots c_{qi\alpha_n}, \tag{B.21}$$

[2] The abscissae of $E_{3,5}^{13}$ correspond to the vertices of an icosahedron. It is therefore *not* related to the D3Q13 model [24] which uses the next-nearest neighbors on a cubic grid.

where b_q is the coordination number of the q-subshell. On a simple cubic lattice, the velocity set is invariant under parity transformations. Hence the odd lattice sums vanish

$$T^{(n)}_{\alpha_1 \ldots \alpha_n} = 0 \qquad \text{if } n \text{ is odd.} \tag{B.22}$$

We proceed to evaluate the sums for $n = 0$, $n = 2$ and $n = 4$ explicitly. For $n = 0$, we get

$$T^{(0)} = \sum_q w_q \sum_{i=1}^{b_q} 1 = \sum_q b_q w_q = 1. \tag{B.23}$$

The second-rank lattice sum on the cubic lattice is an isotropic tensor. We obtain

$$T^{(2)}_{\alpha\beta} = \sum_q w_q \sum_{i=1}^{b_q} c_{qi\alpha} c_{qi\beta} = \sum_q \frac{q b_q}{D} w_q \delta_{\alpha\beta} = \sigma_2 \, \delta_{\alpha\beta}, \tag{B.24}$$

where

$$\sigma_2 = \sum_q \frac{q b_q}{D} w_q. \tag{B.25}$$

To evaluate the fourth-order lattice sum, we note that a fourth rank tensor that is invariant under transformations in the symmetry group of the cubic lattice can be written as a linear combination of a fourth-rank isotropic tensor and a cubic anisotropy $\delta_{\alpha\beta\gamma\delta}$, which is one if all four indices are equal and zero otherwise:

$$T^{(4)}_{\alpha\beta\gamma\delta} = \kappa_4 \delta_{\alpha\beta\gamma\delta} + \sigma_4 \left(\delta_{\alpha\beta}\delta_{\gamma\delta} + \delta_{\alpha\gamma}\delta_{\beta\delta} + \delta_{\alpha\delta}\delta_{\beta\gamma} \right). \tag{B.26}$$

For a lattice with $c_{qi\alpha} \leq 1$, we can evaluate

$$T^{(4)}_{xxxx} = \sum_q w_q \sum_{i=1}^{b_q} c_{qix} c_{qix} c_{qix} c_{qix} = \sum_q w_q \frac{q b_q}{D} = \kappa_4 + 3\sigma_4, \tag{B.27}$$

and

$$T^{(4)}_{\alpha\alpha\beta\beta} = \sum_q w_q \sum_{i=1}^{b_q} c_{qi\alpha} c_{qi\alpha} c_{qi\beta} c_{qi\beta} = \sum_q w_q b_q q^2 = D\kappa_4 + D(D+2)\sigma_4. \tag{B.28}$$

From this it follows

$$\begin{aligned} \kappa_4 &= \sum_q w_q \frac{q b_q (D + 2 - 3q)}{D(D-1)}, \\ \sigma_4 &= \sum_q w_q \frac{q b_q (q-1)}{D(D-1)}. \end{aligned} \tag{B.29}$$

For the D3Q19 model with $q = 0, 1, 2$, we get the following equation system for the weights

$$\begin{aligned} 1 &= w_0 + 6w_1 + 12w_2, \\ \sigma_2 &= 2w_1 + 8w_2, \\ \sigma_4 &= 4w_2, \\ \kappa_4 &= 2w_1 - 4w_2. \end{aligned} \tag{B.30}$$

This admits a solution only if

$$\sigma_2 = 3\sigma_4 + \kappa_4. \tag{B.31}$$

Then the solution is

$$\begin{aligned} w_0 &= 1 - 2\sigma_2 - \kappa_4 = 1 - 6\sigma_4 - 3\kappa_4, \\ w_1 &= \frac{1}{6}\left(\sigma_2 + 2\kappa_4\right) = \frac{1}{2}\left(\sigma_4 + \kappa_4\right), \\ w_2 &= \frac{1}{4}\sigma_4 = \frac{1}{12}\left(\sigma_2 - \kappa_4\right), \end{aligned} \tag{B.32}$$

which is used in the main text to obtain the values (2.54).

B.2.2 Lattice sums and Gauss-Hermite quadrature

Next we prove equation (2.40) of the main text. It states that, for a given lattice with links $\mathbf{c}_i$ and corresponding weights w_i, isotropy of the lattice sums

$$T^{(n)} = \sum_i w_i \mathbf{c}_i \dots \mathbf{c}_i = \begin{cases} 0 & n \text{ odd} \\ \boldsymbol{\delta}^{(n)} & n \text{ even,} \end{cases} \tag{B.33}$$

is equivalent to w_i and $\mathbf{c}_i$ being the weights and nodes of a Gauss-Hermite quadrature of degree $m \geq n$. Let us define $\mathsf{p}_n(\mathbf{v}) \equiv \underbrace{\mathbf{v} \dots \mathbf{v}}_{n \text{ times}}$, then it holds

$$\int \omega(\mathbf{v}) \mathsf{p}_n(\mathbf{v}) d\mathbf{v} = \begin{cases} 0 & n \text{ odd} \\ \boldsymbol{\delta}^{(n)} & n \text{ even.} \end{cases} \tag{B.34}$$

Since $\mathsf{p}_n(\mathbf{v})$ is a tensor polynomial of degree n, it is exactly evaluated by a Gauss-Hermite quadrature of degree $m \geq n$, i.e.,

$$\sum_i w_i \mathbf{c}_i \dots \mathbf{c}_i = \int \omega(\mathbf{v}) \mathsf{p}_n(\mathbf{v}) d\mathbf{v}, \tag{B.35}$$

which proves one direction of the equivalence. For the other direction we note that any polynomial $\mathsf{p}(\mathbf{v})$ of degree m is a linear combination of $\mathsf{p}_n(\mathbf{v})$, $n \leq m$, hence

$$\begin{aligned} \int \omega(\mathbf{v}) \mathsf{p}(\mathbf{v}) \, d\mathbf{v} &= \int \omega(\mathbf{v}) \sum_{k=0}^{m} a_k \mathsf{p}_k(\mathbf{v}) \, d\mathbf{v} \\ &= \sum_{k=0}^{m} a_k \int \omega(\mathbf{v}) \mathsf{p}_k(\mathbf{v}) \, d\mathbf{v} = \sum_{k=0}^{\lfloor \frac{m}{2} \rfloor} a_{2k} \boldsymbol{\delta}^{(2k)}. \end{aligned} \tag{B.36}$$

Then, if the lattice sums are isotropic, we can insert (B.33) for the $\boldsymbol{\delta}$'s in (B.36) and use the definition of $\mathsf{p}_n(\mathbf{c}_i)$ to obtain

$$\int \omega(\mathbf{v})\mathsf{p}(\mathbf{v})\,d\mathbf{v} = \sum_{k=0}^{m} a_k \sum_i w_i \mathbf{c}_i \dots \mathbf{c}_i = \sum_i w_i \sum_{k=0}^{m} a_k \mathsf{p}_k(\mathbf{c}_i) = \sum_i w_i \mathsf{p}(\mathbf{c}_i). \tag{B.37}$$

The equality of the leftmost with the rightmost side implies that w_i and $\mathbf{c}_i$ are the weights and nodes of a quadrature of degree m. This completes the proof of the equivalence. □

B.3 Theoretical analysis of the slip boundary condition

The stationary solution of the lattice Boltzmann equation for the a one-dimensional flow profile can be obtained as follows. For simplicity, we use the single relaxation time approximation for which the stationary solution of the lattice Boltzmann equation reads

$$f_i^{\alpha+\hat{c}_{iz}} = (1+\lambda) f_i^{\alpha} - \lambda f_j^{\mathrm{eq}}(\rho, u_x^{\alpha}) + \Delta_i^{g,\alpha}. \tag{B.38}$$

In the specific case of the D3Q19 model, the stationary solution for the 19 populations is explicitly given by

$$\begin{aligned}
f_0^{\alpha} &= \frac{\rho}{3}\left(1 - \frac{(u_x^{\alpha})^2}{2c_s^2}\right),\\
f_1^{\alpha} &= \frac{\rho}{18}\left(1 + \frac{u_x^{\alpha} a}{c_s^2 \tau} + \frac{(u_x^{\alpha})^2}{c_s^2}\right) - \frac{a f_{\text{ext}}}{18\lambda c_s^2} + \frac{\zeta}{18\lambda c_s^2 a^2} u_x^{\alpha}(\delta_{1,\alpha} + \delta_{\alpha,N}),\\
f_2^{\alpha} &= \frac{\rho}{18}\left(1 - \frac{u_x^{\alpha} a}{c_s^2 \tau} + \frac{(u_x^{\alpha})^2}{c_s^2}\right) + \frac{a f_{\text{ext}}}{18\lambda c_s^2} - \frac{\zeta}{18\lambda c_s^2 a^2} u_x^{\alpha}(\delta_{1,\alpha} + \delta_{\alpha,N}),\\
f_3^{\alpha} &= \frac{\rho}{18},\\
f_4^{\alpha} &= \frac{\rho}{18},\\
f_5^{\alpha} &= \frac{\rho}{18},\\
f_6^{\alpha} &= \frac{\rho}{18},\\
f_7^{\alpha} &= \frac{\rho}{36}\left(1 + \frac{u_x^{\alpha} a}{c_s^2 \tau} + \frac{(u_x^{\alpha})^2}{c_s^2}\right) - \frac{a f_{\text{ext}}}{36\lambda c_s^2} + \frac{\zeta}{36\lambda c_s^2 a^2} u_x^{\alpha}(\delta_{1,\alpha} + \delta_{\alpha,N}),\\
f_8^{\alpha} &= \frac{\rho}{36}\left(1 - \frac{u_x^{\alpha} a}{c_s^2 \tau} + \frac{(u_x^{\alpha})^2}{c_s^2}\right) + \frac{a f_{\text{ext}}}{36\lambda c_s^2} - \frac{\zeta}{36\lambda c_s^2 a^2} u_x^{\alpha}(\delta_{1,\alpha} + \delta_{\alpha,N}),\\
f_9^{\alpha} &= \frac{\rho}{36}\left(1 + \frac{u_x^{\alpha} a}{c_s^2 \tau} + \frac{(u_x^{\alpha})^2}{c_s^2}\right) - \frac{a f_{\text{ext}}}{36\lambda c_s^2} + \frac{\zeta}{36\lambda c_s^2 a^2} u_x^{\alpha}(\delta_{1,\alpha} + \delta_{\alpha,N}),\\
f_{10}^{\alpha} &= \frac{\rho}{36}\left(1 - \frac{u_x^{\alpha} a}{c_s^2 \tau} + \frac{(u_x^{\alpha})^2}{c_s^2}\right) + \frac{a f_{\text{ext}}}{36\lambda c_s^2} - \frac{\zeta}{36\lambda c_s^2 a^2} u_x^{\alpha}(\delta_{1,\alpha} + \delta_{\alpha,N}),
\end{aligned} \tag{B.39}$$

$$
\begin{aligned}
f_{11}^{\alpha} &= -\lambda\frac{\rho}{36}\left(1+\frac{u_x^{\alpha-1}a}{c_s^2\tau}+\frac{(u_x^{\alpha-1})^2}{c_s^2}\right)+(1+\lambda)f_i^{\alpha-1}+\frac{af_{\text{ext}}}{36c_s^2}-\frac{\zeta}{36c_s^2a^2}u_x^{\alpha-1}\delta_{1,\alpha-1},\\
f_{12}^{\alpha} &= -\lambda\frac{\rho}{36}\left(1-\frac{u_x^{\alpha+1}a}{c_s^2\tau}+\frac{(u_x^{\alpha+1})^2}{c_s^2}\right)+(1+\lambda)f_i^{\alpha+1}-\frac{af_{\text{ext}}}{36c_s^2}+\frac{\zeta}{36c_s^2a^2}u_x^{\alpha+1}\delta_{\alpha+1,N},\\
f_{13}^{\alpha} &= -\lambda\frac{\rho}{36}\left(1+\frac{u_x^{\alpha+1}a}{c_s^2\tau}+\frac{(u_x^{\alpha+1})^2}{c_s^2}\right)+(1+\lambda)f_i^{\alpha+1}+\frac{af_{\text{ext}}}{36c_s^2}-\frac{\zeta}{36c_s^2a^2}u_x^{\alpha+1}\delta_{\alpha+1,N},\\
f_{14}^{\alpha} &= -\lambda\frac{\rho}{36}\left(1-\frac{u_x^{\alpha-1}a}{c_s^2\tau}+\frac{(u_x^{\alpha-1})^2}{c_s^2}\right)+(1+\lambda)f_i^{\alpha-1}-\frac{af_{\text{ext}}}{36c_s^2}+\frac{\zeta}{36c_s^2a^2}u_x^{\alpha-1}\delta_{1,\alpha-1},\\
f_{15}^{\alpha} &= -\lambda\frac{\rho}{36}+(1+\lambda)f_j^{\alpha-1},\\
f_{16}^{\alpha} &= -\lambda\frac{\rho}{36}+(1+\lambda)f_j^{\alpha+1},\\
f_{17}^{\alpha} &= -\lambda\frac{\rho}{36}+(1+\lambda)f_j^{\alpha+1},\\
f_{18}^{\alpha} &= -\lambda\frac{\rho}{36}+(1+\lambda)f_j^{\alpha-1}.
\end{aligned}
$$

Let us assume that the direction of the flow is the x-direction. The flow velocity in this direction is obtained as

$$
\begin{aligned}
\rho u_x^{\alpha} &= \sum_i f_i^{\alpha}c_{ix}\\
&= (f_1^{\alpha}-f_2^{\alpha}+f_7^{\alpha}-f_8^{\alpha}+f_9^{\alpha}-f_{10}^{\alpha})\frac{a}{\tau}+(f_{11}^{\alpha}-f_{14}^{\alpha})\frac{a}{\tau}+(f_{13}^{\alpha}-f_{12}^{\alpha})\frac{a}{\tau},
\end{aligned}
\tag{B.40}
$$

where we have grouped the populations in a practically convenient way. The different subexpressions can be evaluated by plugging in the explicit solutions for the populations

$$
(f_1^{\alpha}-f_2^{\alpha}+f_7^{\alpha}-f_8^{\alpha}+f_9^{\alpha}-f_{10}^{\alpha}) = \frac{2a}{9c_s^2\tau}\rho u_x^{\alpha}-\frac{2f_{\text{ext}}a}{9\lambda c_s^2}+\frac{2\zeta}{9\lambda c_s^2a^2}u_x^{\alpha}(\delta_{1,\alpha}+\delta_{\alpha,N}),
\tag{B.41a}
$$

$$
(f_{11}^{\alpha}-f_{14}^{\alpha}) = (1+\lambda)(f_{11}^{\alpha-1}-f_{14}^{\alpha-1})-\frac{\lambda a}{18c_s^2\tau}\rho u_x^{\alpha-1}+\frac{f_{\text{ext}}a}{18c_s^2}-\frac{\zeta}{18c_s^2a^2}u_x^{\alpha-1}\delta_{1,\alpha-1},
\tag{B.41b}
$$

$$
(f_{13}^{\alpha}-f_{12}^{\alpha}) = (1+\lambda)(f_{13}^{\alpha+1}-f_{12}^{\alpha+1})-\frac{\lambda a}{18c_s^2\tau}\rho u_x^{\alpha+1}+\frac{f_{\text{ext}}a}{18c_s^2}-\frac{\zeta}{18c_s^2a^2}u_x^{\alpha+1}\delta_{\alpha+1,N},
\tag{B.41c}
$$

which leads to

$$
\begin{aligned}
\rho u_x^{\alpha} &= (f_1^{\alpha}-f_2^{\alpha}+f_7^{\alpha}-f_8^{\alpha}+f_9^{\alpha}-f_{10}^{\alpha})\frac{a}{\tau}+(f_{11}^{\alpha}-f_{14}^{\alpha})\frac{a}{\tau}+(f_{13}^{\alpha}-f_{12}^{\alpha})\frac{a}{\tau}\\
&= \frac{2}{3}\rho u_x^{\alpha}-\frac{\lambda}{6}\rho\left(u_x^{\alpha-1}+u_x^{\alpha+1}\right)+\frac{\lambda-2}{3\lambda}f_{\text{ext}}\tau\\
&\quad+(1+\lambda)\frac{a}{\tau}\left(f_{11}^{\alpha-1}-f_{14}^{\alpha-1}+f_{13}^{\alpha+1}-f_{12}^{\alpha+1}\right)\\
&\quad+\frac{2\zeta\tau}{3\lambda a^3}u_x^{\alpha}(\delta_{1,\alpha}+\delta_{\alpha,N})-\frac{\zeta\tau}{6a^3}\left(u_x^{\alpha-1}\delta_{1,\alpha-1}+u_x^{\alpha+1}\delta_{\alpha+1,N}\right).
\end{aligned}
\tag{B.42}
$$

Furthermore, we can write

$$(f_{11}^{\alpha} - f_{14}^{\alpha} + f_{13}^{\alpha} - f_{12}^{\alpha})\frac{a}{\tau} = \frac{1}{3}\rho u_x^{\alpha} + \frac{2f_{\text{ext}}\tau}{3\lambda} - \frac{2\zeta\tau}{3\lambda a^3}u_x^{\alpha}(\delta_{1,\alpha} + \delta_{\alpha,N}). \tag{B.43}$$

Using (B.42) and plugging in (B.41a) to (B.41c) we get the following recurrence relations

$$\begin{aligned}
(f_{11}^{\alpha} - f_{14}^{\alpha})\frac{a}{\tau} &= \frac{1}{3}\rho u_x^{\alpha} + \frac{2f_{\text{ext}}\tau}{3\lambda} - \frac{2\zeta\tau}{3\lambda a^3}u_x^{\alpha}(\delta_{1,\alpha} + \delta_{\alpha,N}) - (f_{13}^{\alpha} - f_{12}^{\alpha})\frac{a}{\tau} \\
&= \frac{1}{3}\rho u_x^{\alpha} + \frac{\lambda}{6}\rho u_x^{\alpha+1} + \frac{4-\lambda}{6\lambda}f_{\text{ext}}\tau - \frac{2\zeta\tau}{3\lambda a^3}u_x^{\alpha}(\delta_{1,\alpha} + \delta_{\alpha,N}) \\
&\quad + \frac{\zeta\tau}{6a^3}u_x^{\alpha+1}\delta_{\alpha+1,N} - (1+\lambda)\frac{a}{\tau}(f_{13}^{\alpha+1} - f_{12}^{\alpha+1}), \\
(f_{13}^{\alpha} - f_{12}^{\alpha})\frac{a}{\tau} &= \frac{1}{3}\rho u_x^{\alpha} + \frac{2f_{\text{ext}}\tau}{3\lambda} - \frac{2\zeta\tau}{3\lambda a^3}u_x^{\alpha}(\delta_{1,\alpha} + \delta_{\alpha,N}) - (f_{11}^{\alpha} - f_{14}^{\alpha})\frac{a}{\tau} \\
&= \frac{1}{3}\rho u_x^{\alpha} + \frac{\lambda}{6}\rho u_x^{\alpha-1} + \frac{4-\lambda}{6\lambda}f_{\text{ext}}\tau - \frac{2\zeta\tau}{3\lambda a^3}u_x^{\alpha}(\delta_{1,\alpha} + \delta_{\alpha,N}) \\
&\quad + \frac{\zeta\tau}{6a^3}u_x^{\alpha-1}\delta_{1,\alpha-1} - (1+\lambda)\frac{a}{\tau}(f_{11}^{\alpha-1} - f_{14}^{\alpha-1}).
\end{aligned} \tag{B.44}$$

Adding these two relations leads to

$$\begin{aligned}
&\left(f_{11}^{\alpha-1} - f_{14}^{\alpha-1} + f_{13}^{\alpha+1} - f_{12}^{\alpha+1}\right)\frac{a}{\tau} \\
&\quad= \frac{1}{3}\rho u_x^{\alpha-1} + \frac{\lambda}{3}\rho u_x^{\alpha} + \frac{1}{3}\rho u_x^{\alpha+1} + \frac{4-\lambda}{3\lambda}f_{\text{ext}}\tau \\
&\qquad - \frac{2\zeta\tau}{3\lambda a^3}u_x^{\alpha-1}\delta_{1,\alpha-1} + \frac{\zeta\tau}{6a^3}u_x^{\alpha}(\delta_{1,\alpha} + \delta_{\alpha,N}) - \frac{2\zeta\tau}{3\lambda a^3}u_x^{\alpha+1}\delta_{\alpha+1,N} \\
&\qquad - (1+\lambda)\frac{a}{\tau}(f_{11}^{\alpha} - f_{14}^{\alpha} + f_{13}^{\alpha} - f_{12}^{\alpha}) \\
&\quad= \frac{1}{3}\rho u_x^{\alpha-1} + \frac{\lambda}{3}\rho u_x^{\alpha} + \frac{1}{3}\rho u_x^{\alpha+1} + \frac{4-\lambda}{3\lambda}f_{\text{ext}}\tau \\
&\qquad - \frac{2\zeta\tau}{3\lambda a^3}u_x^{\alpha-1}\delta_{1,\alpha-1} + \frac{\zeta\tau}{6a^3}u_x^{\alpha}(\delta_{1,\alpha} + \delta_{\alpha,N}) - \frac{2\zeta\tau}{3\lambda a^3}u_x^{\alpha+1}\delta_{\alpha+1,N} \\
&\qquad - (1+\lambda)\left(\frac{1}{3}\rho u_x^{\alpha} + \frac{2f_{\text{ext}}\tau}{3\lambda} - \frac{2\zeta\tau}{3\lambda a^3}u_x^{\alpha}(\delta_{1,\alpha} + \delta_{\alpha,N})\right) \\
&\quad= \frac{1}{3}\rho\left(u_x^{\alpha-1} - u_x^{\alpha} + u_x^{\alpha+1}\right) + \frac{2-3\lambda}{3\lambda}f_{\text{ext}}\tau - \frac{2\zeta\tau}{3\lambda a^3}u_x^{\alpha-1}\delta_{1,\alpha-1} \\
&\qquad + \frac{5\lambda+4}{6\lambda}\frac{\zeta\tau}{a^3}u_x^{\alpha}(\delta_{1\alpha} + \delta_{\alpha,N}) - \frac{2\zeta\tau}{3\lambda a^3}u_x^{\alpha+1}\delta_{\alpha+1,N},
\end{aligned} \tag{B.45}$$

which we can finally use in (B.42) to obtain

$$\begin{aligned}\rho u_x^\alpha &= \frac{2}{3}\rho u_x^\alpha - \frac{\lambda}{6}\rho\left(u_x^{\alpha-1} + u_x^{\alpha+1}\right) + \frac{\lambda-2}{3\lambda} f_{\text{ext}}\tau \\ &\quad + (1+\lambda)\frac{a}{\tau}\left(f_{11}^{\alpha-1} - f_{14}^{\alpha-1} + f_{13}^{\alpha+1} - f_{12}^{\alpha+1}\right) \\ &\quad + \frac{2\zeta\tau}{3\lambda a^3} u_x^\alpha(\delta_{1,\alpha} + \delta_{\alpha,N}) - \frac{\zeta\tau}{6a^3}\left(u_x^{\alpha-1}\delta_{1,\alpha-1} + u_x^{\alpha+1}\delta_{\alpha+1,N}\right) \\ &= \frac{1-\lambda}{3}\rho u_x^\alpha + \frac{2+\lambda}{6}\rho(u_x^{\alpha-1} + u_x^{\alpha+1}) - \lambda f_{\text{ext}}\tau \\ &\quad - \frac{5\lambda+4}{6\lambda}\frac{\zeta\tau}{a^3}(u_x^{\alpha-1}\delta_{1,\alpha-1} + u_x^{\alpha+1}\delta_{\alpha+1,N}) \\ &\quad + \frac{8+9\lambda+5\lambda^2}{6\lambda}\frac{\zeta\tau}{a^3} u_x^\alpha(\delta_{1,\alpha} + \delta_{\alpha,N}).\end{aligned} \tag{B.46}$$

Reordering terms and plugging in the expression (4.29) for the viscosity η_s, we finally arrive at the finite difference equation

$$\begin{aligned}\eta_s\frac{u_x^{\alpha-1} - 2u_x^\alpha + u_x^{\alpha+1}}{a^2} &= -f_{\text{ext}} + \frac{8+9\lambda+5\lambda^2}{6\lambda^2}\frac{\zeta}{a^3}u_x^\alpha(\delta_{1,\alpha} + \delta_{\alpha,N}) \\ &\quad - \frac{5\lambda+4}{6\lambda^2}\frac{\zeta}{a^3}(u_x^{\alpha-1}\delta_{1,\alpha-1} + u_x^{\alpha+1}\delta_{\alpha+1,N}).\end{aligned} \tag{B.47}$$

This is equation (5.59) of the main text. □

B.4 Functional derivation of the bulk equilibrium distribution

Here, we rederive the bulk equilibrium distribution as the minimizer of the quadratic functional (6.11). The stationary distribution of the functional is

$$f_i^{\text{eq}} = w_i\left(\lambda_\rho + \lambda_{\mathbf{j},\alpha}c_{i\alpha} + \lambda_{\Pi,\alpha\beta}c_{i\alpha}c_{i\beta}\right). \tag{B.48}$$

The equilibrium distribution has to satisfy the constraints

$$\begin{aligned}\rho &= \sum_i f_i^{\text{eq}}, \\ j_\alpha &= \sum_i f_i^{\text{eq}} c_{i\alpha}, \\ \Pi_{\alpha\beta}^{\text{eq}} &= \sum_i f_i^{\text{eq}} c_{i\alpha}c_{i\beta}.\end{aligned} \tag{B.49}$$

Plugging in (B.48) and using the lattice sums from section B.2 in this appendix, we get

$$\begin{aligned}\rho &= \lambda_\rho + \sigma_2\lambda_{\Pi,\alpha\alpha}, \\ \rho u_\alpha &= \sigma_2\lambda_{\mathbf{j},\alpha}, \\ 3\rho c_s^2 + \rho u_\alpha u_\alpha &= 3\sigma_2\lambda_\rho + 5\sigma_4\lambda_{\Pi,\alpha\alpha}, \\ \rho u_\alpha u_\beta - \frac{1}{3}\rho u_\gamma u_\gamma\delta_{\alpha\beta} &= 2\sigma_4\overline{\lambda_\Pi}_{,\alpha\beta},\end{aligned} \tag{B.50}$$

where have decomposed the stress tensor and the corresponding Lagrange multiplier into their trace and traceless part. Moreover, we have assumed $\kappa_4 = 0$ to ensure isotropy of fourth-rank tensors. The solution of the above equation system is

$$\begin{aligned}
\lambda_\rho &= \frac{1}{5\sigma_4 - 3\sigma_2^2}\left(5\sigma_4\rho - 3\sigma_2\rho c_s^2 - \sigma_2\rho u_\alpha u_\alpha\right), \\
\lambda_{\mathbf{j},\alpha} &= \frac{1}{\sigma_2}\rho u_\alpha, \\
\lambda_{\Pi,\alpha\alpha} &= \frac{1}{5\sigma_4 - 3\sigma_2^2}\left(3\rho c_s^2 - 3\sigma_2\rho + \rho u_\alpha u_\alpha\right), \\
\overline{\lambda_\Pi}_{,\alpha\beta} &= \frac{1}{2\sigma_4}\left(\rho u_\alpha u_\beta - \frac{1}{3}\rho u_\gamma u_\gamma \delta_{\alpha\beta}\right).
\end{aligned} \tag{B.51}$$

For convenience, we continue with the explicit values for the D3Q19 model, where $\sigma_4 = \sigma_2^2 = c_s^4$ and $c_s^2 = 1/3$. Then the Lagrange multipliers are

$$\begin{aligned}
\lambda_\rho &= \rho\left(1 - \frac{1}{2c_s^2}u_\alpha u_\alpha\right), \\
\lambda_{\mathbf{j},\alpha} &= \frac{1}{c_s^2}\rho u_\alpha, \\
\lambda_{\Pi,\alpha\beta} &= \frac{1}{2c_s^4}\rho u_\alpha u_\beta,
\end{aligned} \tag{B.52}$$

and we finally arrive at the equilibrium distribution

$$\begin{aligned}
f_i^{\text{eq}} &= w_i\rho\left[1 - \frac{1}{2c_s^2}u_\alpha\, u_\alpha + \frac{u_\alpha c_{i\alpha}}{c_s^2} + \frac{u_\alpha u_\beta c_{i\alpha}c_{i\beta}}{2c_s^4}\right] \\
&= w_i\rho\left[1 + \frac{u_\alpha c_{i\alpha}}{c_s^2} + \frac{u_\alpha u_\beta\left(c_{i\alpha}c_{i\beta} - c_s^2\delta_{\alpha\beta}\right)}{2c_s^4}\right].
\end{aligned} \tag{B.53}$$

This is the familiar expression for the bulk equilibrium distribution. □

C Sourcecode

The lattice Boltzmann implementation that was developed during this work has been integrated in the ESPResSo software package [3, 4]. The sourcecode can be found on the accompanying CD-ROM. It contains the full ESPResSo package which is distributed under the GNU General Public License (GPL) [180]. The parts whose development has been initiated by this author comprise, inter alia, the following files:

`lattice.h` `lattice.c`	Data structures for lattices and mapping functions
`halo.h` `halo.c`	Data structures for halo regions and parallelization routines
`lb-d3q18.h`	Data structures for D3Q18 model
`lb-d3q19.h`	Data structures for D3Q19 model
`lb.h` `lb.c`	The lattice Boltzmann kernel, including fluctuations and force coupling
`statistics_fluid.h` `statistics_fluid.h`	Data structures for fluid observables and analysis routines
`lb-boundaries.h` `lb-boundaries.c`	Lattice Boltzmann boundary conditions

The author has further contributed various bug-fixes and several extensions to ESPResSo, for example a second order accurate Langevin integrator. A full list is set aside in order to put emphasis on those parts that are relevant for this work.

Bibliography

[1] R. Adhikari, K. Stratford, M. E. Cates and A. J. Wagner. Fluctuating lattice Boltzmann. *Europhys. Lett.* **71**, 473–479 (2005).

[2] B. Dünweg, U. D. Schiller and A. J. C. Ladd. Statistical mechanics of the fluctuating lattice Boltzmann equation. *Phys. Rev. E* **76**, 036704 (2007).

[3] H.-J. Limbach, A. Arnold, B. A. Mann and C. Holm. ESPResSo – An Extensible Simulation Package for Research on Soft Matter Systems. *Comput. Phys. Commun.* **174**, 704–727 (2006).

[4] `http://www.espresso.mpg.de/`

[5] S. Succi. *The Lattice Boltzmann Equation for Fluid Dynamics and Beyond*. Clarendon Press (2001).

[6] D. Raabe. Overview of the lattice Boltzmann method for nano- and microscale fluid dynamics in materials science and engineering. *Modelling and Simulation in Material Sciences and Engineering* **12**, R13–R46 (2004).

[7] J. M. Yeomans. Mesoscale simulations: Lattice Boltzmann and particle algorithms. *Physica A* **369**, 159–184 (2006).

[8] B. Dünweg and A. J. C. Ladd. Lattice Boltzmann simulations of soft matter systems (2008). arXiv:0803.2826v1.

[9] D. A. Wolf-Gladrow. *Lattice-Gas Cellular Automata and Lattice Boltzmann Models*. Springer (2000).

[10] U. Frisch, D. d'Humières, B. Hasslacher et al. Lattice gas hydrodynamics in two and three dimensions. *Complex Systems* **1**, 649 (1987).

[11] L. P. Kadanoff and J. Swift. Transport coefficients near the critical point: A master-equation approach. *Phys. Rev.* **165**, 310–322 (1968).

[12] J. Hardy, Y. Pomeau and O. de Pazzis. Time evolution of a two-dimensional model system. I. Invariant states and time correlation functions. *J. Math. Phys.* **14**, 1746–1759 (1973).

[13] U. Frisch, B. Hasslacher and Y. Pomeau. Lattice-gas automata for the Navier-Stokes equation. *Phys. Rev. Lett.* **56**, 1505–1508 (1986).

[14] D. d'Humières, P. Lallemand and U. Frisch. Lattice gas models for 3D hydrodynamics. *Europhys. Lett.* **2**, 291–297 (1986).

[15] S. Wolfram. Cellular automaton fluids 1: Basic theory. *J. Stat. Phys.* **45**, 471–526 (1986).

[16] R. Benzi, S. Succi and M. Vergalossa. The lattice Boltzmann equation: theory and applications. *Phys. Rep.* **222**, 145–197 (1992).

[17] G. McNamara and G. Zanetti. Use of the Boltzmann equation to simulate lattice-gas automata. *Phys. Rev. Lett.* **61**, 2332–2335 (1988).

[18] F. J. Higuera and J. Jimenez. Boltzmann approach to lattice gas simulations. *Europhys. Lett.* **9**, 663–668 (1989).

[19] P. L. Bhatnagar, E. P. Gross and M. Krook. A model for collision processes in gases. I. Small amplitude processes in charged and neutral one-component systems. *Phys. Rev.* **94**, 511–525 (1954).

[20] J. M. Koelman. A simple lattice Boltzmann scheme for Navier-Stokes fluid flow. *Europhys. Lett.* **15**, 603 (1991).

[21] S. Chen, H. Chen, D. Martinez and W. Matthaeus. Lattice Boltzmann model for simulation of magnetohydrodynamics. *Phys. Rev. Lett.* **67**, 3776 (1991).

[22] Y. H. Qian, D. d'Humières and P. Lallemand. Lattice BGK models for Navier-Stokes equation. *Europys. Lett.* **17**, 479–484 (1992).

[23] D. d'Humières. Generalized lattice-Boltzmann equations. In *Rarefied gas dynamics: theory and simulations*, edited by B. D. Shizgal and D. P. Weaver, volume 159 of *Prog. Aeronaut. Astronaut.*, pp. 450–458 (1992).

[24] D. d'Humières, M. Bouzidi and P. Lallemand. Thirteen-velocity three-dimensional lattice Boltzmann model. *Phys. Rev. E* **63**, 066702 (2001).

[25] D. d'Humières, I. Ginzburg, M. Krafczyk, P. Lallemand and L.-S. Luo. Multiple-relaxation time lattice Boltzmann models in three dimensions. *Philos. Trans. R. Soc. Lond. A* **360**, 437–451 (2002).

[26] P. Lallemand and L.-S. Luo. Theory of the lattice Boltzmann method: Dispersion, dissipation, isotropy, Galilean invariance, and stability. *Phys. Rev. E* **61**, 6546–6562 (2000).

[27] A. J. Wagner. An H-theorem for the lattice Boltzmann approach to hydrodynamics. *Europhys. Lett.* **44**, 144–149 (1998).

[28] I. V. Karlin, A. N. Gorban, S. Succi and V. Boffi. Maximum entropy principle for lattice kinetic equations. *Phys. Rev. Lett.* **81**, 6–9 (1998).

[29] I. V. Karlin and S. Succi. Equilibria for discrete kinetic equations. *Phys. Rev. E* **58**, R4053–R4056 (1998).

[30] I. V. Karlin, A. Ferrante and H. C. Öttinger. Perfect entropy functions of the lattice Boltzmann method. *Europhys. Lett.* **47**, 182–188 (1999).

[31] S. Ansumali and I. V. Karlin. Stabilization of the lattice Boltzmann method by the H theorem: A numerical test. *Phys. Rev. E* **62**, 7999–8003 (2000).

[32] S. Succi, I. V. Karlin and H. Chen. *Colloquium*: Role of the H theorem in lattice Boltzmann hydrodynamic simulations. *Rev. Mod. Phys.* **74**, 1203–1220 (2002).

[33] X. He and L.-S. Luo. *A priori* derivation of the lattice Boltzmann equation. *Phys. Rev. E* **55**, R6333–R6336 (1997).

[34] X. He and L.-S. Luo. Theory of the lattice Boltzmann method: From the Boltzmann equation to the lattice Boltzmann equation. *Phys. Rev. E* **56**, 6811–6817 (1997).

[35] X. Shan and X. He. Discretization of the Velocity Space in the Solution of the Boltzmann equation. *Phys. Rev. Lett.* **80**, 65–68 (1998).

[36] A. J. C. Ladd. Short-time motion of colloidal particles - Numerical simulation via a fluctuating lattice-Boltzmann equation. *Phys. Rev. Lett.* **70**, 1339–1342 (1993).

[37] A. J. C. Ladd. Numerical simulations of particulate suspensions via a discretized Boltzmann equation. Part 1. Theoretical foundation. *J. Fluid Mech.* **271**, 285–309 (1994).

[38] A. J. C. Ladd. Numerical simulations of particulate suspensions via a discretized Boltzmann equation. Part 2. Numerical results. *J. Fluid Mech.* **271**, 311–339 (1994).

[39] X. W. Shan and H. D. Chen. Lattice Boltzmann model for simulating flows with multiple phases and components. *Phys. Rev. E* **47**, 1815–1819 (1993).

[40] X. W. Shan and H. D. Chen. Simulation of nonideal gases and liquid-gas phase-transitions by the lattice Boltzmann-equation. *Phys. Rev. E* **49**, 2941–2948 (1994).

[41] M. R. Swift, W. R. Osborn and J. M. Yeomans. Lattice Boltzmann simulation of nonideal fluids. *Phys. Rev. Lett.* **75**, 830–833 (1995).

[42] M. R. Swift, E. Orlandini, W. R. Osborn and J. M. Yeomans. Lattice Boltzmann simulations of liquid-gas and binary fluid systems. *Phys. Rev. E* **54**, 5041–5052 (1996).

[43] F. J. Alexander, S. Chen and J. D. Sterlig. Lattice Boltzmann thermohydrodynamics. *Phys. Rev. E* **47**, R2249–R2252 (1993).

[44] Y. H. Qian. Simulating thermohydrodynamics with lattice BGK models. *J. Sci. Comp.* **8**, 231–241 (1993).

[45] Y. Chen, H. Ohashi and M. Akiyama. Thermal lattice Bhatnagar-Gross-Krook model without nonlinear deviations in macrodynamic equations. *Phys. Rev. E* **50**, 2776–2783 (1994).

[46] M. Sbragaglia and S. Succi. A note on the lattice Boltzmann method beyond Chapman-Enskog limits. *Europhys. Lett.* **73**, 370–376 (2006).

[47] X. Shan, X.-F. Yuan and H. Chen. Kinetic theory representation of hydrodynamics: a way beyond the Navier-Stokes equation. *J. Fluid. Mech.* **550**, 413–441 (2006).

[48] S. Ansumali, I. V. Karlin, S. Arcidiacono, A. Abbas and N. I. Prasianakis. Hydrodynamics beyond Navier-Stokes: Exact solution to the lattice Boltzmann hierarchy. *Phys. Rev. Lett.* **98**, 124502 (2007).

[49] L. D. Landau and E. M. Lifshitz. *Fluid Mechanics*, volume 6 of *Course of Theoretical Physics*. 2nd rev. repr. edition. Butterworth-Heinemann (2002).

[50] P. Résibois and M. de Leener. *Classical Kinetic Theory of Fluids*. John Wiley & Sons (1977).

[51] C. Cercignani. *The Boltzmann Equation and Its Applications*, volume 67 of *Applied Mathematical Sciences*. Springer (1988).

[52] S. Chapman and T. G. Cowling. *The mathematical theory of non-uniform gases*. 3rd edition. Cambridge Mathematical Library Series, Cambridge University Press (1990).

[53] H. Grad. On the kinetic theory of rarified gases. *Comm. Pure Appl. Math.* **2**, 331–407 (1949).

[54] C. D. Levermore. Moment closure hierarchies for kinetic theories. *J. Stat. Phys.* **83**, 1021–1065 (1996).

[55] J. E. Broadwell. Shock structure in a simple discrete velocity gas. *Phys. Fluids* **7**, 1243–1247 (1964).

[56] J. E. Broadwell. Study of rarefied shear flow by the discrete velocity method. *J. Fluid Mech.* **19**, 401–414 (1964).

[57] L.-S. Luo. Some recent results on discrete velocity models and ramifications for lattice Boltzmann equation. *Comp. Phys. Comm.* **129**, 63–74 (2000).

[58] Y. H. Qian and S. A. Orszag. Lattice BGK models for the Navier-Stokes equation: Nonlinear deviation in compressible regimes. *Europhys. Lett.* **21**, 255–259 (1993).

[59] X. B. Nie, X. Shan and H. Chen. Galilean invariance of lattice Boltzmann models. *Europhys. Lett.* **81**, 34005 (2008).

[60] R. Rubinstein and L.-S. Luo. Theory of the lattice Boltzmann equation: Symmetry properties of discrete velocity sets. *Phys. Rev. E* **77**, 036709 (2008).

[61] X. Y. He, L. S. Luo and M. Dembo. Some progress in lattice Boltzmann method. 1. Nonuniform mesh grids. *J. Comp. Phys.* **129**, 357–363 (1996).

[62] H. D. Chen. Volumetric formulation of the lattice Boltzmann method for fluid dynamics: Basic concept. *Phys. Rev. E* **58**, 3955–3963 (1998).

[63] P. Ahlrichs. *Computersimulationen zur Dynamik von verdünnten und halbverdünnten Polymerlösungen*. Ph.D. thesis, Johannes Gutenberg-Universität Mainz (2000).

[64] I. Ginzburg and D. d'Humières. Multireflection boundary conditions for lattice Boltzmann models. *Phys. Rev. E* **68**, 066614 (2003).

[65] A. J. C. Ladd and R. Verberg. Lattice-Boltzmann simulations of particle-fluid suspensions. *J. Stat. Phys.* **104**, 1191–1251 (2001).

[66] O. B. Usta, A. J. C. Ladd and J. E. Butler. Lattice-Boltzmann simulations of the dynamics of polymer solutions in periodic and confined geometries. *J. Chem. Phys.* **122**, 094902 (2005).

[67] P. J. Hoogerbrugge and J. M. V. A. Koelman. Simulating Microscopic Hydrodynamic Phenomena with Dissipative Particle Dynamics. *Europhys. Lett.* **19**, 155–160 (1992).

[68] J. M. V. A. Koelman and P. J. Hoogerbrugge. Dynamic simulation of hard-sphere suspensions under steady shear. *Europhys. Lett.* **21**, 363 (1993).

[69] P. Español and P. Warren. Statistical Mechanics of Dissipative Particle Dynamics. *Europhys. Lett.* **30**, 191–196 (1995).

[70] P. Español. Hydrodynamics from dissipative particle dynamics. *Phys. Rev. E* **52**, 1734–1742 (1995).

[71] R. D. Groot and P. B. Warren. Dissipative particle dynamics: Bridging the gap between atomistic and mesoscopic simulation. *J. Chem. Phys.* **107**, 4423–4435 (1997).

[72] A. Malevanets and R. Kapral. Mesoscopic model for solvent dynamics. *J. Chem. Phys.* **110**, 8605–8613 (1999).

[73] T. Ihle and D. M. Kroll. Stochastic rotation dynamics: A Galilean-invariant mesoscopic model for fluid flow. *Phys. Rev. E* **6302**, 020201 (2001).

[74] T. Ihle and D. M. Kroll. Stochastic rotation dynamics. I. Formalism, Galilean invariance, and Green-Kubo relations. *Phys. Rev. E* **67**, 066705 (2003).

[75] T. Ihle and D. M. Kroll. Stochastic rotation dynamics. II. Transport coefficients, numerics, and long-time tails. *Phys. Rev. E* **67**, 066706 (2003).

[76] R. G. Winkler, A. Ripoll, K. Mussawisade and G. Gompper. Simulation of complex fluids by multi-particle-collision dynamics. *Comp. Phys. Comm.* **169**, 326–330 (2005).

[77] H. Noguchi, N. Kikuchi and G. Gompper. Particle-based mesoscale hydrodynamic techniques. *Europhys. Lett.* **78**, 10005 (2007).

[78] I. O. Gotze, H. Noguchi and G. Gompper. Relevance of angular momentum conservation in mesoscale hydrodynamics simulations. *Phys. Rev. E* **76**, 046705 (2007).

[79] H. Noguchi and G. Gompper. Transport coefficients of off-lattice mesoscale-hydrodynamics simulation techniques. *Phys. Rev. E* **78**, 016706 (2008).

[80] C. M. Pooley and J. M. Yeomans. Kinetic theory derivation of the transport coefficients of stochastic rotation dynamics. *J. Phys. Chem. B* **109**, 6505–6513 (2005).

[81] L. D. Landau and E. M. Lifshitz. *Statistical Physics. Theory of the condensed state*, volume 9 of *Course of Theoretical Physics*. Repr. edition. Butterworth-Heinemann (2002).

[82] R. F. Fox and G. E. Uhlenbeck. Contributions to non-equilibrium thermodynamics. I. Theory of hydrodynamical fluctuations. *Phys. Fluids* **13**, 1893–1902 (1970).

[83] E. J. Hinch. Application of the Langevin equation to fluid suspensions. *J. Fluid Mech.* **72**, 499–511 (1975).

[84] N. van Kampen. *Stochastic Processes in Physics and Chemistry*. North-Holland (2001).

[85] P. Ahlrichs and B. Dünweg. Lattice Boltzmann simulation of polymer-solvent systems. *Int. J. Mod. Phys. C* **9**, 1429–1438 (1998).

[86] P. Ahlrichs and B. Dünweg. Simulation of a single polymer chain in solution by combining lattice Boltzmann and molecular dynamics. *J. Chem. Phys.* **111**, 8225–8239 (1999).

[87] V. Lobaskin and B. Dünweg. A new model for simulating colloidal dynamics. *New Journal Of Physics* **6**, 54 (2004).

[88] V. Lobaskin, B. Dünweg, M. Medebach, T. Palberg and C. Holm. Electrophoresis of colloidal dispersions in the low-salt regime. *Phys. Rev. Lett.* **98**, 176105 (2007).

[89] E. M. Lishitz and L. P. Pitaevski. *Physical Kinetics*. Pergamon Press (1981).

[90] S. Ansumali. *Minimal Kinetic Modeling of Hydrodynamics*. Ph.D. thesis, Swiss Federal Institute of Technology, Zurich (2004).

[91] S. Ansumali, I. V. Karlin and H. C. Öttinger. Minimal entropic kinetic models for hydrodynamics. *Europhys. Lett.* **63**, 798–804 (2003).

[92] M. Junk, A. Klar and L. S. Luo. Asymptotic analysis of the lattice Boltzmann equation. *J. Comp. Phys.* **210**, 676–704 (2005).

[93] S. Chen and G. D. Doolen. Lattice Boltzmann method for fluid flows. *Annu. Rev. Fluid Mech.* **30**, 329–364 (1998).

[94] G. McNamara and B. Alder. Analysis of the lattice Boltzmann treatment of hydrodynamics. *Physica A* **194**, 218–228 (1993).

[95] R. W. Mei, L. S. Luo, P. Lallemand and D. d'Humieres. Consistent initial conditions for lattice Boltzmann simulations. *Computers & Fluids* **35**, 855–862 (2006).

[96] A. Caiazzo. Analysis of lattice Boltzmann nodes initialisation in moving boundary problems. *Prog. Comp. Fluid Dyn.* **8**, 3–10 (2008).

[97] M. Junk and Z. Yang. Asymptotic analsysis of lattice Boltzmann boundary conditions. *J. Stat. Phys.* **121**, 3–35 (2005).

[98] Z. Guo, C. Zheng and B. Shi. Discrete lattice effects on the forcing term in the lattice Boltzmann method. *Phys. Rev. E* **65**, 046308 (2002).

[99] `http://www.ladhyx.polytechnique.fr/activities/micro_en.html`

[100] A. Groisman, M. Enzelberger and S. R. Quake. Microfluidic memory and control devices. *Science* **300**, 955–958 (2003).

[101] J. Latt, B. Chopard, O. Malaspinas, M. Deville and A. Michler. Straight velocity boundaries in the lattice Boltzmann method. *Phys. Rev .E* **77**, 056703 (2008).

[102] M. Fichman and G. Hetsroni. Viscosity and slip velocity in gas flow in microchannels. *Phys. Fluids* **17**, 123102 (2005).

[103] Y.-H. Zhang, X.-J. Gu, R. W. Barber and D. R. Emerson. Capturing Knudsen layer phenomena using a lattice Boltzmann model. *Phys. Rev. E* **74**, 046704 (2006).

[104] Y. Zhang, R. Qin and D. R. Emerson. Lattice Boltzmann simulation of rarefied gas flows in microchannels. *Phys. Rev. E* **71**, 044702 (2005).

[105] C. Y. Lim, C. Shu, X. D. Niu and Y. T. Chew. Application of lattice Boltzmann method to simulate microchannel flows. *Phys. Fluids* **14**, 2299–2308 (2002).

[106] L. Ren, K.-Q. Zhu and X.-L. Wang. Effects of the slip velocity boundary condition on the characteristics of microbearings. *J. Micromech. Microeng.* **14**, 116–124 (2004).

[107] J. Maurer, P. Tabeling, P. Joseph and H. Willaime. Second-order slip laws in microchannels for helium and nitrogen. *Phys. Fluids* **15**, 2613–2621 (2003).

[108] H. A. Stone, A. D. Stroock and A. Ajdari. Engineering flows in small devices: Microfluidics towards a lab-on-a-chip. *Annu. Rev. Fluid Mech.* **36**, 381–411 (2004).

[109] E. Lauga, M. P. Brenner and H. A. Stone. Microfluidics: The No-Slip Boundary Condition. In *Handbook of Experimental Fluid Dynamics*, edited by C. Tropea, A. Yarin and J. F. Foss, chapter 19, pp. 1219–1240. Springer, New York (2005).

[110] R. Benzi, L. Biferale, M. Sbragaglia, S. Succi and F. Toschi. Mesoscopic modelling of heterogenous boundary conditions for microchannel flows. *J. Fluid Mech.* **548**, 257–280 (2006).

[111] L. Bocquet and J.-L. Barrat. Flow boundary conditions from nano- to micro-scales. *Soft Matter* **3**, 685–693 (2007).

[112] C. L. M. H. Navier. Mémoire sur les lois du mouvement des fluides. *Mémoire de l'Académie Royale des Sciences de l'Institut de France* **6**, 389–440 (1823).

[113] J. C. Maxwell. On stresses in rarified gases arising from inequalities of temperature. *Philos. Trans. R. Soc. London* **170**, 231–256 (1879).

[114] E. P. Gross, E. A. Jackson and S. Ziering. Boundary value problems in kinetic theory of gases. *Ann. Phys.* **1**, 141–167 (1957).

[115] E. P. Gross and E. A. Jackson. Kinetic models and the linearized Boltzmann equation. *Phys. Fluids* **2**, 432–441 (1959).

[116] A. J. Kainz and U. M. Titulaer. An accurate two-stream moment method for kinetic boundary layer problems of linear kinetic equations. *J. Phys. A: Math. Gen.* **25**, 1855–1874 (1992).

[117] S. Ansumali and I. V. Karlin. Kinetic boundary conditions in the lattice Boltzmann method. *Phys. Rev. E* **66**, 026311 (2002).

[118] N. G. Hadjiconstantinou. Comment on Cercignani's second-order slip coefficient. *Phys. Fluids* **15**, 2352–2354 (2003).

[119] H. Helmholtz and G. v. Piotrowski. Üeber die Reibung tropfbarer Flüssigkeiten. *Sitz.-Ber. der k. Akad. d. Wiss. in Wien. Math. nat. Kl.* **XL**, 607–658 (1860).

[120] W. P. Yudistiawan, S. Ansumali and I. V. Karlin. Hydrodynamics beyond Navier-Stokes: The slip flow model. *Phys. Rev. E* **78**, 016705 (2008).

[121] R. Cornubert, D. d'Humières and D. Levermore. A Knudsen layer theory for lattice gases. *Physica D* **47**, 241–259 (1991).

[122] I. Ginzbourg and P. M. Adler. Boundary flow condition analysis for three-dimensional lattice Boltzmann model. *J. Phys. II France* **4**, 191–214 (1994).

[123] O. Behrend. Solid-fluid boundaries in particle suspension simulations via the lattice Boltzmann method. *Phys. Rev. E* **52**, 1164–1175 (1995).

[124] Q. Zou, S. Hou and G. D. Doolen. Analytical solutions of the lattice Boltzmann BGK model. *J. Stat. Phys.* **81**, 319–334 (1995).

[125] L.-S. Luo. Analytic solutions of linearized lattice Boltzmann equation for simple flows. *J. Stat. Phys.* **88**, 913–926 (1997).

[126] X. He, Q. Zou, L.-S. Luo and M. Dembo. Analytic solutions of simple flows and analysis of nonslip boundary conditions for the lattice Boltzmann BGK model. *J. Stat. Phys.* **87**, 115–136 (1997).

[127] M. Sbragaglia and S. Succi. Analytical calculation of slip flow in lattice Boltzmann models with kinetic boundary conditions. *Phys. Fluids* **17**, 093602 (2005).

[128] S. Succi. Mesoscopic modeling of slip motion at fluid-solid interfaces with heterogeneous catalysis. *Phys. Rev. Lett.* **89**, 064502 (2002).

[129] L. Szalmás. Slip-flow boundary condition for straight walls in the lattice Boltzmann model. *Phys. Rev. E* **73**, 066710 (2006).

[130] G. H. Tang, W. Q. Tao and Y. L. He. Lattice Boltzmann method for gaseous microflows using kinetic theory boundary conditions. *Phys. Fluids* **17**, 058101 (2005).

[131] B. Chun and A. J. C. Ladd. Interpolated boundary condition for lattice Boltzmann simulations of flows in narrow gaps. *Phys. Rev. E* **75**, 066705 (2007).

[132] D. P. Ziegler. Boundary conditions for lattice Boltzmann simulations. *J. Stat. Phys.* **71**, 1171–1177 (1993).

[133] P. A. Skordos. Initial and boundary conditions for the lattice Boltzmann method. *Phys. Rev. E* **48**, 4823–4841 (1993).

[134] D. R. Noble, S. Chen, J. G. Georgiadis and R. O. Buckius. A consistent hydrodynamic boundary condition for the lattice Boltzmann method. *Phys. Fluids* **7**, 203–209 (1995).

[135] D. R. Noble, J. G. Georgiadis and O. Buckius, Richard. Direct assessment of lattice Boltzmann hydrodynamics and boundary conditions for recirculating flows. *J. Stat. Phys.* **81**, 17–33 (1995).

[136] R. S. Maier, R. S. Bernard and D. W. Grunau. Boundary conditions for the lattice Boltzmann method. *Phys. Fluids* **8**, 1788–1801 (1996).

[137] T. Inamuro, M. Yoshino and F. Ogino. A non-slip boundary condition for lattice Boltzmann simulations. *Phys. Fluids* **7**, 2928–2930 (1995).

[138] T. Inamuro, M. Yoshino and F. Ogino. Erratum: "A non-slip boundary condition for lattice Boltzmann simulations". *Phys. Fluids* **8**, 1124 (1996).

[139] Q. Zou and X. He. On pressure and velocity boundary conditions for the lattice Boltzmann BGK model. *Phys. Fluids* **9**, 1591–1598 (1997).

[140] J. Lätt. *Hydrodynamic Limit of Lattice Boltzmann Equations*. Ph.D. thesis, Faculté des sciences de l'Université de Genève (2007).

[141] I. Ginzbourg and D. d'Humières. Local second-order boundary methods for lattice Boltzmann models. *J. Stat. Phys.* **84**, 927–971 (1996).

[142] I. Halliday, L. A. Hammond and C. M. Care. Enhanced closure scheme for lattice Boltzmann equation hydrodynamics. *J. Phys. A.: Math. Gen.* **35**, L157–L166 (2002).

[143] S. Chen, D. Martínez and R. Mei. On boundary conditions in lattice Boltzmann methods. *Phys. Fluids* **8**, 2527–2536 (1996).

[144] O. Filippova and D. Hänel. Lattice Boltzmann simulation of gas-particle flow in filters. *Comput. Fluids* **26**, 697 (1997).

[145] O. Filippova and D. Hänel. Grid refinement for Lattice-BGK models. *J. Comp. Phys.* **147**, 219–228 (1998).

[146] O. Filippova and D. Hänel. Boundary fitting and local grid refinement for lattice-BGK models. *Int. J. Mod. Phys. C* **9**, 1271–1279 (1998).

[147] R. Mei, L.-S. Luo and W. Shyy. An accurate curved boundary treatment in the lattice Boltzmann method. *J. Comp. Phys.* **155**, 307–330 (1999).

[148] D. Yu, R. Mei, L.-S. Luo and W. Shyy. Viscous flow computations with the method of lattice Boltzmann equation. *Prog. Aerosp. Sci.* **39**, 329–367 (2003).

[149] Z. Guo, C. Zheng and B. Shi. An extrapolation method for boundary conditions in lattice Boltzmann method. *Phys. Fluids* **14**, 2007–2010 (2002).

[150] R. Verberg and A. J. C. Ladd. Lattice-Boltzmann model with sub-grid-scale boundary conditions. *Phys. Rev. Lett.* **84**, 2148–2151 (2000).

[151] R. Verberg and A. J. C. Ladd. Accuracy and stability of a lattice-Boltzmann model with subgrid scale boundary conditions. *Phys. Rev. E* **65**, 016701 (2001).

[152] R. Y. Zhang, H. D. Chen, Y. H. Qian and S. Chen. Effective volumetric lattice Boltzmann scheme. *Phys. Rev. E* **63**, 056705 (2001).

[153] H. L. Fan, R. Y. Zhang and H. D. Chen. Extended volumetric scheme for lattice Boltzmann models. *Phys. Rev. E* **73**, 066708 (2006).

[154] M. Bouzidi, M. Firdaouss and P. Lallemand. Momentum transfer of a Boltzmann-lattice fluid with boundaries. *Phys. Fluids* **13**, 3452–3459 (2001).

[155] P. Lallemand and L.-S. Luo. Lattice Boltzmann method for moving boundaries. *J. Comp. Phys.* **184**, 406–421 (2003).

[156] M. Rohde, D. Kandhai, J. J. Derksen and H. E. A. van den Akker. Improved bounce-back methods for no-slip walls in lattice-Boltzmann schemes: Theory and simulations. *Phys. Rev. E* **67**, 066703 (2003).

[157] L. Szalmás. Knudsen layer theory for high-order lattice Boltzmann models. *Europhys. Lett.* **80**, 24003 (2007).

[158] L.-S. Luo. Comment on "Discrete Boltzmann equation for microfluidics". *Phys. Rev. Lett.* **92**, 139401 (2004).

[159] J.-L. Barrat and L. Bocquet. Large slip effect at a nonwetting fluid-solid interface. *Phys. Rev. Lett.* **82**, 4671–4674 (1999).

[160] R. Benzi, L. Biferale, M. Sbragaglia, S. Succi and F. Toschi. Mesoscopic two-phase model for describing apparent slip in micro-channel flows. *Europhys. Lett.* **74**, 651–657 (2006).

[161] R. Benzi, L. Biferale, M. Sbragaglia, S. Succi and F. Toschi. Mesoscopic modeling of a two-phase flow in the presence of boundaries: The contact angle. *Phys. Rev. E* **74**, 021509 (2006).

[162] J. Hyväluoma and J. Harting. Slip flow over structured surfaces with entrapped microbubbles. *Phys. Rev. Lett.* **100**, 246001 (2008).

[163] C. Kunert and J. Harting. Roughness induced boundary slip in microchannel flows. *Phys. Rev. Lett.* **99**, 176001 (2007).

[164] C. Kunert and J. Harting. Simulation of fluid flow in hydrophobic rough microchannels. *Int. J. Comp. Fluid Dyn.* **22**, 475–480 (2008).

[165] J. Smiatek, M. P. Allen and F. Schmid. Tunable-slip boundaries for coarse-grained simulations of fluid flow. *Eur. Phys. J. E* **26**, 115–122 (2008).

[166] N. S. Martys, X. Shan and H. Chen. Evaluation of the external force term in the discrete Boltzmann equation. *Phys. Rev. E* **58**, 6855–6857 (1998).

[167] J. M. Buick and C. A. Greated. Gravity in a lattice Boltzmann model. *Phys. Rev. E* **61**, 5307–5320 (2000).

[168] R. Mei, D. Yu, W. Shyy and L.-S. Luo. Force evaluation in the lattice Boltzmann method involving curved geometry. *Phys. Rev. E* **65**, 041203 (2002).

[169] R. W. Nash, R. Adhikari and M. E. Cates. Singular forces and pointlike colloids in lattice Boltzmann hydrodynamics. *Phys. Rev. E* **77**, 026709 (2008).

[170] H. Struchtrup. Kinetic schemes and boundary conditions for moment equations. *Zeitschrift für Angewandte Mathematik und Physik (ZAMP)* **51**, 346–365 (2000).

[171] M. Torrilhon and H. Struchtrup. Boundary conditions for regularized 13-moment-equations for micro-channel-flows. *J. Comp. Phys.* **227**, 1982–2011 (2008).

[172] M. Z. Bazant and O. I. Vinogradova. Tensorial hydrodynamic slip (2008). arXiv:0808.0765v1.

[173] W. Gropp, E. Lusk and A. Skjellum. *Using MPI*. MIT Press (1996).

[174] G. Wellein, T. Zeiser, G. Hager and S. Donath. On the single processor performance of simple lattice Boltzmann kernels. *Computers & Fluids* **35**, 910–919 (2006).

[175] T. Pohl, N. Thürey, F. Deserno et al. Performance evaluation of parallel large-scale lattice Boltzmann applications on three supercomputing architectures. Technical Report, Lehrstuhl für Informatik, Friedrich-Alexander-Universität Erlangen-Nürnberg (2004).

[176] W. H. Press, S. A. Teukolsky, W. T. Vetterling and B. P. Flannery. *Numerical recipes in C*. 2nd edition. Cambridge University Press (1992).

[177] H. Grad. Note on N-dimensional Hermite polynomials. *Comm. Pure Appl. Math.* **2**, 325–330 (1949).

[178] I. N. Bronstein and K. A. Semendjajew. *Taschenbuch der Mathematik*. 3rd edition. Verlag Harri Deutsch (1997).

[179] A. H. Stroud. *Approximate Calculation of Multiple Integrals*. Prentice Hall (1971).

[180] `http://www.gnu.org/licenses/`