

Lattice Boltzmann Method for the Simulation of Viscoelastic Fluid Flows

THÈSE N° 4505 (2009)

PRÉSENTÉE LE 23 OCTOBRE 2009

À LA FACULTÉ SCIENCES ET TECHNIQUES DE L'INGÉNIEUR
LABORATOIRE D'INGÉNIERIE NUMÉRIQUE
PROGRAMME DOCTORAL EN MÉCANIQUE

ÉCOLE POLYTECHNIQUE FÉDÉRALE DE LAUSANNE

POUR L'OBTENTION DU GRADE DE DOCTEUR ÈS SCIENCES

PAR

Orestis Pileas MALASPINAS

acceptée sur proposition du jury:

Prof. C. Ancey, président du jury
Prof. M. Deville, directeur de thèse
Prof. B. Chopard, rapporteur
Dr M. Picasso, rapporteur
Dr S. Succi, rapporteur



ÉCOLE POLYTECHNIQUE
FÉDÉRALE DE LAUSANNE

Suisse
2009

Remerciements

Avant tout je dédie cette thèse à *linpc4* qui nous a quitté un mois trop tôt et n'a pas pu voir l'achèvement de ces quatre ans de travail.

Je souhaiterais remercier un certain nombre de personnes qui m'ont aidé et supporté pendant les années de recherche qui ont mené à la rédaction de cette thèse.

Tout d'abord, je dis un grand merci à mon guide et chef, Michel Deville, à Nicolas Fiétier pour ses lumières viscoélastiques, ainsi qu'à Bastien Chopard qui a contribué à mon épanouissement dans le dur monde de Boltzmann et de son réseau. Je suis également reconnaissant aux membres de mon jury les Professeurs Christophe Ancey, Marco Picasso et Sauro Succi.

J'ai également été heureux d'être le colocataire du bureau ME A2 398 (merci à lui) au côtés d'abord de Roland puis de Benoit, qui avec Richard et Emeric ont encouragé mes visites régulières au "Satellite". Je ne saurais être complet sans également saluer Christoph et Andrea qui grâce à leur dévouement sans faille ont su être mon premier "étudiant" et mon premier "disciple" respectivement. Un énorme merci à Roxane et à "son mari" qui a rendu possible l'achèvement de l'an zéro de son plan quinquennal, tant du point de vue scientifique que moral (merci Jonas).

Tout cela n'aurait évidemment pas été possible sans le "Café du Lys" ainsi que de tous mes amis qui m'y ont accompagné pour décompresser (presque) chaque fin de semaine : Adriana, Aless, Caro, Mathilde, Nilay, Julien, Fogg, Navin, Stéphane et Tybalt (il y en a sûrement d'autres mais je sais pas s'ils méritent...).

Je remercie aussi mes parents, Maria et Andreas, pour leur soutien moral, technique et logistique (sans eux, c'est certain, rien n'aurait été possible), ainsi qu'Anna-**Sapfo**, Iliona et Orfeas pour avoir été de bons et attentionnés frères et soeurs. Un remerciement spécial aussi à Spitha, CDROM et Jack.

Finalement, une reconnaissance infinie à Céline, qui a dû et su me supporter pendant toutes ces années de "dur" labeur.

Contents

Remerciements	3
Contents	7
Résumé	13
Abstract	15
1 Introduction	19
1.1 State-of-the-art	19
1.2 Summary of the thesis	24
2 The Boltzmann equation and the hydrodynamic limit	27
2.1 The cons. eqs. derived from the Boltzmann equ.	27
2.2 Projection on the Hermite basis	31
2.3 Chapman–Enskog expansion	32
2.3.1 Order three truncation	37
2.3.2 Order two truncation	39
2.4 Discretization of velocity space	40
2.5 Time and space discretization	43
2.6 Summary	46
3 Non-Newtonian fluid dynamics	49
3.1 Generalized Newtonian fluids	49
3.2 Viscoelastic fluids	50
3.2.1 Action of the solvent on the polymers	50
3.2.2 Action of the polymers on the solvent	53
3.2.3 Jeffreys model	57
3.3 Summary	57
4 Velocity boundary conditions in the lattice Boltzmann method	59
4.1 Description of the problem	59
4.2 Proposed solutions	60
4.2.1 General properties of the studied BC	60

4.2.2	Computation of the density	61
4.2.3	Replacing only the missing populations or all of them	62
4.3	Description of the boundary conditions	63
4.3.1	The Inamuro method (<i>BC1</i>)	64
4.3.2	The Zou/He method (<i>BC2</i>)	66
4.3.3	The Regularized method (<i>BC3</i>)	68
4.3.4	The Finite-difference velocity gradient method (<i>BC4</i>)	68
4.4	Benchmarks	70
4.4.1	Steady plane channel flow (2D)	72
4.4.2	Oscillating plane channel flow (2D)	73
4.4.3	Periodic array of vortices (2D)	75
4.4.4	Dipole-wall collision (2D)	77
4.4.5	Rectangular steady channel flow (3D)	79
4.5	Discussion	81
5	Simulation of non-Newtonian fluids	85
5.1	Generalized Newtonian fluids	85
5.2	Viscoelastic fluids	86
5.2.1	The advection–diffusion with source scheme	87
5.2.2	Constitutive equation simulation	89
5.2.3	Conformation tensor boundary conditions	90
5.3	Description of the algorithm	93
6	Numerical results	97
6.1	Generalized Newtonian fluids	97
6.1.1	Steady planar Poiseuille flow	97
6.1.2	4:1 contraction	100
6.2	Viscoelastic fluids benchmarks	100
6.2.1	Taylor–Green vortex	101
6.2.2	The four–roll mill (simplified)	104
6.2.3	Steady planar Poiseuille flow	109
6.3	Parallelization of the viscoelastic model	111
6.4	Discussion	113
7	Conclusions and perspectives	119
A	Properties of the Hermite polynomials	121
B	Gauss–Hermite quadrature formulae	125

C Initial pressure field for the
Taylor–Green vortex

129

List of symbols

Acronyms & abbreviations

BC	Boundary Condition
l.h.s.	Left hand side
LBM	Lattice Boltzmann method
MRT	Multiple relaxation time
MSUPS	Mega Sites Updates per Second
r.h.s.	Right hand side

Adimensional numbers

Cu	Carreau number
De	Deborah number
D	Number of physical dimensions
Kn	Knudsen number
Ma	Mach number
Wi	Weissenberg number
Wi^{eff}	Effective Weissenberg number

Greek letters

$\dot{\gamma}$	Shear rate
$\dot{\epsilon}$	Elongation rate
$\rho\epsilon$	Internal energy density
ζ	Mobility
η	Vorticity
θ	$k_B T$

ξ	Particle velocity
ξ_i	Discretized microscopic velocity
κ	Thermal diffusivity
κ_p	Polymer diffusivity
λ	Polymer relaxation time
Λ	Deviatoric stress tensor
μ_0	Dynamic viscosity at zero shear rate
μ_∞	Dynamic viscosity at infinite shear rate
μ_p	Polymer dynamic viscosity
μ_s	Dynamic viscosity of the solvent
ν	Kinematic viscosity of the fluid
Π	Out of equilibrium polymer stress tensor
ρ	Density
$\bar{\rho}$	Density in terms of \bar{f}_i
ρ_p	Polymer density
Δt	Time increment
τ	Fluid relaxation time
$\bar{\tau}$	Relaxation time of the numerical scheme
Δx	Space increment
ψ	Configuration vector probability distribution function
$\omega(\xi)$	Weight function of the Hermite polynomial expansion
Roman letters	
$a_{0\alpha}^{(n)}$	Equilibrium Hermite expansion coefficient of order n
$a_\alpha^{(n)}$	Hermite expansion coefficient of order n

\mathbf{A}	Conformation tensor
\mathbf{c}	Microscopic velocity in the \mathbf{u} frame
\mathbf{c}_i	Discretized microscopic velocity in the \mathbf{u} frame
c_l	Lattice sound speed scaling factor
c_s	Sound speed
$E_{D,m}^Q$	Lattice of dimension D , quadrature order m with Q edges
\mathcal{E}	Total energy
\mathcal{E}_{kin}	Total kinetic energy
$\mathcal{E}_{\text{spring}}$	Total spring energy
\bar{f}_i	Discretized velocity distribution function of the numerical scheme
f_i^{eq}	“Effective” equilibrium distribution function
$f^{(1)}$	Discretized off-equilibrium velocity distribution function
$f_i^{(1)}$	Discretized off-equilibrium velocity distribution function
$f^{(0)}$	Equilibrium velocity distribution function
$f_i^{(0)}$	Discretized equilibrium velocity distribution function
f	Particle mass velocity distribution function
F	Force term in the Boltzmann equation
F_i	Discretized Boltzmann force term
f_i	Discretized velocity distribution function
$\rho\mathbf{g}$	Body force
$\mathcal{H}_{\alpha}^{(n)}$	Hermite polynomial of order n
$\mathcal{H}_{i\alpha}^{(n)}$	Discretized Hermite polynomial of degree n
H	Hooke constant
\mathbf{j}	Density flux
k_B	Boltzmann constant

l	Mean free path
L	Characteristic length scale of the system
m	Particle mass
\mathbf{x}	Position
\mathbf{P}	Stress tensor
\mathbf{P}_p	Stress tensor of the polymer
\mathbf{P}_s	Stress tensor of the solvent
p	Pressure
\mathbf{q}	Internal energy flux
\mathbf{r}	Configuration vector
\mathbf{r}_1	Position of bead one
\mathbf{r}_2	Position of bead two
\mathbf{r}_c	Position of the center of mass of the dumbbells
r_0	Norm of the conformation vector at equilibrium
r_e	Maximal length of the dumbbell
\mathbf{S}	Strain rate tensor (incompressible case)
\mathbf{s}	Connector force
T	Temperature
t	Time
\mathbf{u}	Macroscopic velocity field
$\bar{\mathbf{u}}$	Velocity in terms of \bar{f}_i
\mathbf{v}	Velocity of the dumbbell beads in the frame of the fluid velocity \mathbf{u}
w_i	Discrete weights of the Gauss–Hermite quadrature

Résumé

La simulation d'écoulements de fluides viscoélastiques est un domaine qui présente de grands défis tant du point de vue théorique que de celui de la modélisation numérique. En particulier, jusqu'ici, toutes les méthodes traditionnelles de résolution d'équations constitutives se sont heurtées au problème du nombre de Weissenberg élevé. Il est dès lors évident que de nouveaux outils sont nécessaires. Cette thèse propose d'appliquer la méthode de Boltzmann sur réseau (MBR) à la simulation de fluides viscoélastiques présentant des effets de mémoire, modélisés avec une équation constitutive faisant appel explicitement à une dérivée convective supérieure, ce qui est la première tentative du genre pour des écoulements non triviaux.

Une dérivation théorique des modèles discrets correspondant aux équations de conservation de la masse, de la quantité de mouvement et de l'équation constitutive est présentée ainsi que le traitement particulier des conditions aux limites associées.

Nous commençons par présenter un algorithme permettant de simuler des écoulements de fluides Newtoniens généralisés (i.e. sans élasticité, rhéo-amincissants ou rhéo-épaississants) en résolvant l'écoulement plan de Poiseuille et dans une contraction 4:1 en 2D. Puis nous proposons une méthode de résolution d'équations constitutives présentant des effets de mémoire. En particulier, les équations basées sur des modèles d'haltères avec ressort à extensibilité infinie ou finie (modèles d'Oldroyd-B et de FENE-P respectivement) que nous couplons avec les équations de Navier–Stokes incompressibles. Nous appliquons notre modèle sur les cas tests du moulin à quatre rouleaux simplifié en 2D et sur la décroissance des vortex de Taylor–Green en 3D. Puis, finalement, nous proposons des conditions aux bords pour parois planes que nous validons sur le cas de l'écoulement Poiseuille bidimensionnel.

Deux avantages principaux de la méthode proposée sont la facilité d'implémentation de nouveaux modèles viscoélastiques et la parallélisation de l'algorithme.

Mots clés : dynamique des fluides numérique, méthode de Boltzmann sur réseau, fluides non-Newtoniens, fluides Newtoniens généralisés, fluides viscoélastiques, Navier–Stokes incompressible, conditions aux bords, développement de Chapman–Enskog.

Abstract

The simulation of flows of viscoelastic fluids is a very challenging domain from the theoretical as well as the numerical modelling point of view. In particular, all the existing methods have failed to solve the high Weissenberg number problem (HWNP). It is therefore clear that new tools are necessary. In this thesis we propose to tackle the problem of the simulation of viscoelastic fluids presenting memory effects, which is the first attempt of applying the lattice Boltzmann method (LBM) to this field for non-trivial flows.

A theoretical development of the discrete models corresponding to the equations of mass, momentum conservation and of the constitutive equation is presented as well as the particular treatment of the associated boundary conditions.

We start by presenting a simplified case where no memory but shear-thinning or shear-thickening effects are present : simulating the flow of generalized Newtonian fluids. We test the corresponding method against two-dimensional benchmarks : the 2D planar Poiseuille and the 4:1 contraction flows. Then we propose a new model consisting in solving the constitutive equations that account for memory effects, by explicitly including an upper-convected derivative, using the lattice Boltzmann method. In particular, we focus on the polymer dumbbell models, with infinite or finite spring extension (Oldroyd-B and FENE-P models). Using our model, we study the periodic (simplified) 2D four-roll mill and the 3D Taylor–Green decaying vortex cases. Finally, we propose an approach for simulating flat walls and show the applicability of the method on the 2D planar Poiseuille case.

Two of the advantages of the proposed method are the ease of implementation of new viscoelastic models and of an algorithm for parallel computing.

Keywords : computational fluid dynamics, lattice Boltzmann method, non-Newtonian fluids, generalized Newtonian fluids, viscoelastic fluids, incompressible Navier–Stokes, boundary conditions, Chapman–Enskog expansion.

Notations

We introduce here the mathematical notations that will be used throughout this thesis.

The scalar variables will be noted with lower case characters. The vectors will be in boldface lower case and the tensors will be in boldface upper case characters. The scalar product between two vectors will be noted by a dot “.” whereas the full index contraction is denoted by a colon “:”. The tensor product of two vectors \mathbf{a} and \mathbf{b} is denoted \mathbf{ab} .

For vectors and tensors we will also use the indexed notation. The Greek subscript indices range over physical dimensions. When dealing with higher dimensional spaces then the subscript indices are lower case Latin characters. We will use Einstein’s convention for summation. When a subscript letter is repeated twice in a product, a summation over all possible values of the index is implied. For example in a 3D space the scalar product $\boldsymbol{\xi} \cdot \mathbf{u}$ reads

$$\boldsymbol{\xi} \cdot \mathbf{u} = \xi_\alpha u_\alpha = \xi_x u_x + \xi_y u_y + \xi_z u_z.$$

The nabla operator, ∇ , is the vector containing the space derivatives in each spatial direction if not specified otherwise. If there is an possible ambiguity the quantity on which will depend the derivative will be underscored. In the table below the reader will find a summary of the notations, with some examples.

Object	Example	Index notation
Scalar	ρ, u	ρ, u
Vector	\mathbf{u}, \mathbf{a}	u_α, a_α
Tensor	$\mathbf{\Pi}, \mathbf{Q}$	$\Pi_{\alpha\beta}, Q_{\alpha\beta\gamma}$
Scalar product	$\boldsymbol{\xi} \cdot \mathbf{u}$	$\xi_\alpha u_\alpha$
Full tensor contraction	$\mathbf{\Pi} : \mathbf{S}$	$\Pi_{\alpha\beta} S_{\alpha\beta}$
Tensor product	$\mathbf{c}\boldsymbol{\xi}$	$c_\alpha \xi_\beta$
Nabla	$\nabla_{\mathbf{x}}, \nabla_{\mathbf{r}}$	$\partial_{x_\alpha}, \partial_{r_\alpha}$

Chapter 1

Introduction

1.1 State-of-the-art

Nowadays the lattice Boltzmann method (LBM) has established itself as a powerful tool for the simulation of a wide range of physical phenomena. One of its main applications is the field of computational fluid dynamics where it has proven successful to solve the weakly compressible Navier–Stokes equations (see Wolf-Gladrow [100], Succi [94], Chen and Doolen [18], Dellar [25, 26]) , but also much more complex situations as multiphase and multi-component flows (see e.g. Shan and Chen [91], Reis and Phillips [89]). This method is not solving directly the macroscopic conservation equations, but rather models the statistics of collision of particles and offers more modelling freedom. Therefore the LBM is a very good alternative to classical solvers like finite differences, finite volume, finite element, or spectral element methods (see respectively Griebel et al. [42], Versteeg and Malalasekera [97], Zienkiewicz et al. [103], Deville et al. [28] for example).

Historically, the lattice Boltzmann method was derived from the cellular automata and in particular the FHP model (named after Frisch, Hasslacher and Pomeau who published the model in [34]). It was shown, using the Chapman–Enskog multiscale expansion (see Chapman and Cowling [17]), that this model is able to reproduce the weakly compressible Navier–Stokes equations by modelling the flow with fictitious particles moving on a regular lattice and obeying very simple collision rules. One of the great advantages of this method is that it is unconditionally stable. However, it suffers from the statistical noise due to the discreteness of the particles. In order to try to correct this defect McNamara and Zanetti [74] proposed to replace the particles by their velocity distribution function. This idea gave birth to the lattice Boltzmann method in a paper by Higuera and Jimenez [52]. Eight years later, He and Luo [48] showed that the numerical scheme could be directly derived *a priori* from the continuous BGK (Bhatnagar, Gross and Krook proposed this model in [7]) Boltzmann equation by a proper discretiza-

tion of the velocity space¹. The discretization by He and Luo [48] contained a slight error since their derivation could only demonstrate first-order time accuracy of the method, although it was known since the beginnings of the lattice Boltzmann developments, that the scheme was second order accurate. In He et al. [49] the correct way of discretizing the space and time was proposed. The idea of these papers, to make use of the Hermite polynomials and of the Gauss–Hermite quadrature to discretize the microscopic velocity space, was used in Shan et al. [92] to construct models that can not only simulate the incompressible limit of the Navier–Stokes equations, but also isothermal compressible fluids and to incorporate also temperature dependence like in Nie et al. [76]. This idea pushed a step further could also lead to the simulation of the Grad’s 13-moments equations (see Grad [40]).

All the models discussed in the preceding paragraph are based on the BGK, or simple relaxation time, collision operator. Other collision operators were proposed to improve the stability and accuracy of the method and also to widen the range of physical phenomena that can be simulated. Two of them were the multiple relaxation time (MRT) model (see for example d’Humières [29], d’Humières et al. [30]) and the entropic model (see Ansumali and Karlin [3], Boghosian et al. [11] for example).

The MRT models modify the collision by projecting the distribution functions in the moments (of the velocity distribution) space and relaxing them towards a “moments” equilibrium with different relaxation times. Some of the moments are related to macroscopic quantities while others can be seen as “ghosts” which play no physical role, but can interfere numerically in the simulations. By tuning the relaxation times related with these “ghost modes”, one can improve the accuracy and the stability of the numerical scheme (see the paper by Dellar [26] for example). The entropic model approach consists of two distinct parts. A discrete Boltzmann’s H -function is defined. Then the equilibrium distribution function is computed by minimizing this “entropy” (in the BGK model the equilibrium is assumed to be Maxwellian). Then the relaxation towards this equilibrium is modified in order to impose that the H -function is never growing in order to satisfy the H theorem (see Huang [53] for example). This has as a consequence to impose the unconditional stability of the model. From the macroscopic point of view, the correction of the collision can be viewed as a modification of the viscosity depending on the shear rate tensor (see Malaspinas et al. [72]).

¹The main idea of this model is that the effect of the particles’ collision consists in driving the velocity distribution towards a prescribed (usually Maxwellian) local equilibrium with a characteristic constant (relaxation) time.

The lattice Boltzmann community was also very quickly interested in the boundary conditions to be used in their simulations, and especially in the behavior on solid walls. In the macroscopic world, the fluid “sticks” on solid boundaries (no-slip condition) and this is reflected by a zero velocity field. However, it is not straightforward to translate this information to the velocity distribution function. This topic is of crucial importance for the numerical simulations, since the numerical stability and accuracy are usually determined by the quality of the boundary conditions. The first (and still the most popular) boundary condition that was developed was the so-called bounce-back, in which all the velocities of particles that hit the walls are reversed. Its simplicity, unconditional stability and the fact that it is orientation independent, makes it particularly appealing. Nevertheless, it is only able to mimic a motionless wall, and therefore it does not allow to impose any flow in a simulation. Furthermore, its accuracy is rather low and reduces the overall exactness of a simulation. In order to correct these defects, a large amount of boundary conditions were proposed to impose Dirichlet conditions (prescribed velocity field) on walls. We give here a non exhaustive list of references for some of them that are used for flat walls (see Skordos [93], Inamuro et al. [54], Zou and He [104], Ansumali and Karlin [4], Latt and Chopard [68]) and for curved boundaries in Bouzidi et al. [14] and in Ginzburg and d’Humières [36].

In this thesis we apply the LBM to the simulation of viscoelastic fluids. This class of fluids is characterized by the fact that the deformation of an element of fluid because of stress does not only depend on the strain itself (as it is the case for Newtonian fluids) but also on the history of the deformation (memory effect). The polymer melts or solutions, which are present in industrial and everyday life products, are a good example of such fluids. The viscoelastic fluids are very challenging from the theoretical as well as from the numerical point of view. There exist numerous very different models, that range from molecular descriptions which consider all the microscopic interactions between particles, to macroscopic continuum mechanics models that ignore the small constitutive elements of the fluids (see the books by Bird et al. [8, 9], Larson [64] and Lin [70] for example).

The continuum based theories, like the Oldroyd-B model, are based on invariance properties of the deformation of continuous media that have been investigated in many details by classical methods, from finite differences to spectral element methods (see Tomé et al. [96], Bonito et al. [12], Keunings [59], Fiétier [33] for example). Unfortunately they all suffer from the same problem of representing correctly the fluids with strong viscoelastic interactions, which are characterized by a high Weissenberg (Wi) or Deborah (De)

number² ($Wi = 0$ or $De = 0$ numbers correspond to Newtonian fluids). It is indeed reported in the literature as the HWNP (high Weissenberg number problem) (e. g. [82]) that all these methods fail to converge as Wi increases. There was a need for novel modelling techniques. However it is not clear which one of the physical model or of the numerical method, used for the resolution of the viscoelastic simulations, is responsible for the lack of accuracy and numerical stability in the high Weissenberg number problem. One possible issue would be an inappropriate treatment of the boundaries. For the Newtonian fluids we know by experience that the fluid is sticking on the walls (no-slip boundary condition). There exists no equivalent condition for the stress experienced by the viscoelastic fluids and in practice one of the few analytical results must be used to impose the behavior of the viscoelastic stress on the boundaries of a flow.

In order to improve the modelling of viscoelastic flows, the community has followed different approaches that depart from the continuum mechanics. The molecular dynamics models give a very good description of the microscopic phenomena. But their prohibitive computational cost is incompatible with the study of macroscopic interactions of polymers with a solvent for non-trivial geometries. An intermediate scale approach is more affordable from this point of view. The development of kinetic theory approaches, that provide a coarse grained description of the molecules (see the book by Bird et al. [8] or the review by Keunings [60] more recently) represent therefore a good alternative. These models are describing the evolution of the probability distribution function of the molecule configurations. These polymer molecules are seen as chains of beads connected by rigid segments or by springs, a quite good approximation for dilute solutions. For polymer concentrated solutions or melts, a kinetic theory of entangled systems has been used. One of the first models was the reptation theory by Edwards, de Gennes, and Doi (see de Gennes [24], Doi and Edwards [31]). Suffering from different defects this model has been improved to include more physical processes, such that contour length fluctuations, tube stretch and convective constraint release. A review of these models can be found in McLeish [73] for example. However these mathematical models are difficult to solve on a computer, and usually macroscopic constitutive equations are derived from them and then solved using classical techniques (see Keunings [59] for example). The step from the mesoscopic to the macroscopic world is made using averages which are done provided some simplifying mathematical and

²The Weissenberg number, $Wi \equiv \lambda \dot{\gamma}$, is defined as the product of the relaxation time λ of the fluid and the shear rate, while, the Deborah number, $De = \lambda/t_c$, is the ratio of the relaxation time over characteristic time scale of the physical process, t_c .

physical assumptions which may filter out some of the underlying physics. Furthermore the macroscopic constitutive equations are usually equivalent to the ones obtained with the continuum approaches and therefore do not offer really an alternative from the numerical point of view. The development of the micro-macro methods that couple the dynamics of the kinetic theory approach with the macroscopic continuum mechanics are a good alternative. These methods, which became really popular thanks to Laso and Öttinger [65], are well described in the book by Öttinger [80], are now a well established tool (see for example Grande et al. [41] and Bonito et al. [12]).

In this thesis we will try to tackle the problem of the simulation of viscoelastic fluids, with a completely different methodology : the lattice Boltzmann method. The hope is that the mesoscopic kinetic theory approach will provide some progress in the field. Solutions have already been proposed for generalized Newtonian fluids³ by different authors. The main idea of all these models, based on the BGK collision, is to recompute the relaxation time (which is linearly related to viscosity) at each cell and for each time step depending on the constitutive equation relating the shear rate with the viscosity. In Aharonov and Rothman [2], the problem of generalized Newtonian fluids in diffusion problems is tackled by simulating a power-law fluid. This approach is then used for the simulation of 2D porous media. In Ouared and Chopard [81] the blood flow is approximated using the Casson model and applied on the 2D Poiseuille flow as well as a 2D aneurysm geometry. In these two papers the relaxation time is recomputed locally for each timestep and position. In Gabbanelli et al. [35] the problem of the simulation of a truncated power-law model fluid is addressed, but here in contrast to what has been done in Aharonov and Rothman [2] and Ouared and Chopard [81], the relaxation time is not recomputed locally, but first-order accurate finite differences are used. This model is then used for the 2D Poiseuille flow and for the reentrant corner problem. In Kehrwald [58], the simulation of shear thinning fluids is performed using a MRT model and applied to both 2D Poiseuille and single-screw extruder geometries. In Boyd et al. [15], the power-law problem for a LB BGK model is solved using a second-order local scheme, but only in the 2D Poiseuille case. In Malaspina et al. [71] the cases of power-law and Carreau–Yasuda fluids were simulated in the cases of steady Poiseuille and 4:1 contractions flows. Finally in Vikhansky [98] the LBM is used for the simulation of Bingham fluids for flows in a cylinder and

³The viscosity of the fluid is depending on the shear rate, but there is no memory effects as in viscoelastic fluids. The viscosity is either decreasing or increasing with growing shear rate. These two cases correspond to shear-thinning or shear-thickening effects.

a periodic mesh of cylinders.

For the viscoelastic fluids, that exhibit memory effects, there exist only very few papers in the literature, which only partly take into account elastic effects in the flow. In Qian and Deng [88] the elastic effect is obtained by modifying the equilibrium distribution, whereas in Ispolatov and Grant [55] the elastic effect is obtained by adding a Maxwell-like (exponentially decaying) force to the system. In Giraud et al. [37, 38], and in Lallemand et al. [62] a scheme for solving the Jeffreys model is proposed, but it fails to exhibit some important elastic effects since the stress tensor is assumed traceless. In recent works by Onishi et al. [78, 79] the Fokker–Plank counterpart for the Oldroyd-B and FENE-P models were simulated with the help of the LBM. The numerical results presented were obtained for very simple shear flows and did not allow us to assert that the viscoelastic effects were very strong. These two approaches are the state-of-the-art lattice Boltzmann models for viscoelastic fluids. In Denniston et al. [27] the LBM is used for the simulation of liquid crystals with an approach that shares some similarities with ours although the constitutive equation is much more complex, which limits the type of geometries that can be handled. There is therefore still room for improvements in this domain, that we will discuss in this work. This thesis is to our knowledge the first attempt to solve the full set of equations including conservation and macroscopic constitutive equations of the extended Oldroyd-B type of non-linear effects for non-trivial flows. Only isothermal and incompressible fluids are considered here.

1.2 Summary of the thesis

The outline of this thesis is the following. In Chapter 2 we present the *a priori* construction of the lattice Boltzmann method from the continuous Boltzmann equation, and give different macroscopic regimes of validity of the method. We also give the explicit implementation formulas and show how to, *a priori*, incorporate an external force to the scheme in a coherent way, which to our knowledge was never presented before. In Chapter 3 we quickly introduce the basics of non-Newtonian fluid dynamics, by describing the generalized Newtonian case and the dumbbell model for viscoelastic fluids. In Chapter 4, we discuss the issue of the velocity boundary conditions in the lattice Boltzmann method by analyzing theoretically and numerically some of the most popular treatments described in the literature. We use the Chapman–Enskog expansion to discuss their validity and run several benchmarks, on steady and unsteady flows, and compare them with existing analytical and numerical solutions. In Chapter 5 we describe our models for

the simulation of non-Newtonian fluids. As a first step we describe the procedure to implement any constitutive model, and then apply the method for a Carreau fluid. Then we describe our model for incorporating viscoelasticity through an Oldroyd-B constitutive equation. We also discuss the crucial point of boundary conditions for the configuration tensor of the viscoelastic model. In Chapter 6, we make a numerical validation of the models presented in Chapter 5. We then compute the accuracy of the method by comparing our solution with a semi-analytical solution for the Poiseuille case and then, in a more complicated geometry, with a commercial solver for the 4:1 planar contraction. In the viscoelastic case, we implement the Oldroyd-B constitutive equation and validate our models by comparing simulations with the analytical solution for a planar Poiseuille flow and for the flow in a 2D simplified four-roll mill, and with a high accuracy Fourier pseudospectral algorithm for the Taylor–Green vortex case. As the lattice Boltzmann method is intrinsically efficient for parallel computing, we present the corresponding results at the end of Chapter 6. Finally we conclude this dissertation in Chapter 7.

Chapter 2

The Boltzmann equation and the hydrodynamic limit

In this chapter we will summarize briefly the theory of the continuous Boltzmann equation and the macroscopic limits one can consider. For more details one should refer to Huang [53] or Zwanzig [105] for example.

2.1 The conservation equations derived from the Boltzmann equation

In the kinetic theory framework, we are interested in the dynamics of large numbers of particles that are interacting through collisions. Since it is essentially impossible to track them individually and since one is in practice more interested in the behavior of macroscopic states rather than in the state of each particle, we will concentrate on the particle mass distribution function $f(\mathbf{x}, \boldsymbol{\xi}, t)$ which is defined such that

$$f(\mathbf{x}, \boldsymbol{\xi}, t) d^3x d^3\xi, \quad (2.1)$$

is the total mass of particles which are located in the volume d^3x around position \mathbf{x} and which have a velocity located in the volume $d^3\xi$ around $\boldsymbol{\xi}$ at time t . This approximation is valid whenever small volume elements $d^3x d^3\xi$ contain a large number of particles¹. We will consider the different moments of f

$$\rho(\mathbf{x}, t) = \int d\boldsymbol{\xi} f(\mathbf{x}, \boldsymbol{\xi}, t), \quad (2.2)$$

$$\mathbf{j}(\mathbf{x}, t) = \rho \mathbf{u} = \int d\boldsymbol{\xi} \boldsymbol{\xi} f(\mathbf{x}, \boldsymbol{\xi}, t), \quad (2.3)$$

where the integration is carried out on the whole velocity space. In these two equations, we can easily identify ρ as the density, and \mathbf{u} as the mean

¹Typically a volume of $d^3x = 10^{-9}\text{cm}^3$ and $d^3\xi = 10^{-2}\xi$ ($\xi \sim \sqrt{k_B T}$, $T = 1^\circ\text{C}$) still contains 10^4 particles.

velocity field. The higher order moments are more conveniently computed with respect to the mean velocity. Accordingly we introduce the microscopic velocity in the mean velocity frame

$$\mathbf{c}(\boldsymbol{\xi}, \mathbf{x}, t) = \boldsymbol{\xi} - \mathbf{u}(\mathbf{x}, t). \quad (2.4)$$

The internal energy density is then given by

$$\rho\epsilon = \frac{1}{2} \int d\mathbf{c} \, c^2 f(\mathbf{x}, \boldsymbol{\xi}, t), \quad (2.5)$$

where $c^2 = \|\mathbf{c}\|^2$ is the norm of \mathbf{c} squared. We can relate the internal energy density, for a mono-atomic gas, to the temperature by the equipartition of energy

$$\rho\epsilon = \frac{D}{2} \rho\theta, \quad (2.6)$$

where D is the physical dimension, $\theta = k_B T/m$ with k_B the Boltzmann constant, m the particles mass and T the temperature. We are now interested in the time evolution of f . In the presence of an external force field \mathbf{g} one can write a balance equation namely, the Boltzmann equation,

$$\left(\partial_t + \boldsymbol{\xi} \cdot \nabla_{\mathbf{x}} + \frac{\mathbf{g}}{m} \cdot \nabla_{\boldsymbol{\xi}} \right) f(\mathbf{x}, \boldsymbol{\xi}, t) = \Omega, \quad (2.7)$$

where the collision integral (or collision operator) Ω , represents the changes in f due to the inter-particle collisions. The Boltzmann equation is a non-linear integro-differential equation, a notoriously intractable equation (except for very small volumes where direct Monte Carlo simulations can be used). Without any loss of generality, we will assume that $m = 1$ in the sequel.

Usually Ω is given in terms of the scattering cross section and is quadratic in f (since only binary collisions are considered). Bhatnagar, Gross and Krook (see Bhatnagar et al. [7]) proposed a simplified model where the assumption is made that the effect of the collision between particles is to drive the gas towards a local equilibrium state. This model, known under the name of BGK, is given by

$$(\partial_t + \boldsymbol{\xi} \cdot \nabla_{\mathbf{x}} + \mathbf{g} \cdot \nabla_{\boldsymbol{\xi}}) f = -\frac{1}{\tau} (f - f^{(0)}), \quad (2.8)$$

where τ is the relaxation time and the equilibrium distribution $f^{(0)}$ is just the Maxwell–Boltzmann distribution function

$$f^{(0)}(\mathbf{x}, \boldsymbol{\xi}, t) = \frac{\rho(\mathbf{x}, t)}{(2\pi\theta(\mathbf{x}, t))^{D/2}} \exp\left(-\frac{(\mathbf{u}(\mathbf{x}, t) - \boldsymbol{\xi})^2}{2\theta(\mathbf{x}, t)}\right). \quad (2.9)$$

We are now interested in the macroscopic physics that can be obtained thanks to the BGK equation. Defining the velocity average of an observable $\mathcal{A}(\mathbf{x}, \boldsymbol{\xi}, t)$ by

$$\langle \mathcal{A} \rangle(\mathbf{x}, t) = \frac{1}{\rho} \int d\boldsymbol{\xi} \mathcal{A} f, \quad (2.10)$$

and multiplying (2.8) by \mathcal{A} and integrating over the velocity space, one can show that the equation of motion of the quantity $\langle \mathcal{A} \rangle$ is given by

$$\begin{aligned} -\frac{\rho}{\tau} (\langle \mathcal{A} \rangle - \langle \mathcal{A} \rangle^{\text{eq}}) &= \partial_t (\rho \langle \mathcal{A} \rangle) + \nabla_{\mathbf{x}} \cdot (\rho \langle \boldsymbol{\xi} \mathcal{A} \rangle) - \rho \langle \boldsymbol{\xi} \cdot \nabla_{\mathbf{x}} \mathcal{A} \rangle \\ &\quad - \rho \langle \mathcal{A} \nabla_{\boldsymbol{\xi}} \cdot \mathbf{g} \rangle - \rho \langle \mathbf{g} \cdot \nabla_{\boldsymbol{\xi}} \mathcal{A} \rangle, \end{aligned} \quad (2.11)$$

where $\langle \mathcal{A} \rangle^{\text{eq}}$ is

$$\langle \mathcal{A} \rangle^{\text{eq}} = \frac{1}{\rho} \int d\boldsymbol{\xi} \mathcal{A} f^{(0)}. \quad (2.12)$$

By definition, the expectation value of a conserved observable is the same if computed with the equilibrium, $f^{(0)}$, or the complete, f , distribution functions. In our case, these fields are the density ρ , the momentum, $\rho \mathbf{u}$, and the internal energy density $\rho \epsilon$, which are given by

$$\rho = \rho \langle 1 \rangle = \rho \langle 1 \rangle^{\text{eq}}, \quad (2.13)$$

$$\rho \mathbf{u} = \rho \langle \boldsymbol{\xi} \rangle = \rho \langle \boldsymbol{\xi} \rangle^{\text{eq}}, \quad (2.14)$$

$$\rho \epsilon = \frac{\rho}{2} \langle \mathbf{c}^2 \rangle = \frac{\rho}{2} \langle \mathbf{c}^2 \rangle^{\text{eq}}. \quad (2.15)$$

Replacing \mathcal{A} in Eq. (2.11) by those three moments, we see that the l.h.s. vanishes, and we get the general conservation laws. For $\mathcal{A} = 1$ we have the continuity equation

$$\partial_t \rho + \nabla \cdot (\rho \mathbf{u}) = 0. \quad (2.16)$$

For $\mathcal{A} = \boldsymbol{\xi}$ we obtain the momentum conservation equation

$$\partial_t (\rho \mathbf{u}) + \nabla \cdot \langle \boldsymbol{\xi} \boldsymbol{\xi} \rangle = \rho \mathbf{g}.$$

By defining the stress or pressure tensor²

$$\mathbf{P} = \int d\mathbf{c} \mathbf{c} \mathbf{c} f, \quad (2.17)$$

² \mathbf{P} is called either stress tensor in the mechanics community or pressure tensor in the kinetic theory community.

where \mathbf{cc} is the tensor product of \mathbf{c} with \mathbf{c} , we can rewrite this equation in a more familiar way

$$\begin{aligned}\partial_t(\rho\mathbf{u}) + \nabla \cdot (\rho\mathbf{u}\mathbf{u}) - \nabla \cdot \mathbf{P} &= \rho\mathbf{g}, \\ \rho \frac{d\mathbf{u}}{dt} - \nabla \cdot \mathbf{P} &= \rho\mathbf{g},\end{aligned}\tag{2.18}$$

where $\frac{d}{dt}$ represents the total derivative with respect to t and where we used the continuity equation to simplify the result. The trace of the stress tensor gives us the hydrostatic pressure, which is related to the internal energy density by

$$p = \frac{P_{\alpha\alpha}}{D} = \frac{2\rho\epsilon}{D}.\tag{2.19}$$

The use of the equipartition theorem (Eq. (2.6)) leads to the perfect gas law

$$p = \rho\theta.\tag{2.20}$$

Finally if $\mathcal{A} = \xi^2/2$ we have the energy conservation equation

$$\partial_t \left(\rho\epsilon + \frac{1}{2}\rho\mathbf{u}^2 \right) + \nabla \cdot \left\langle \frac{1}{2}\xi^2\xi \right\rangle = 0.\tag{2.21}$$

In order to rewrite this equation in a more convenient notation, we define the internal energy flux \mathbf{q}

$$\mathbf{q} = \frac{1}{2} \int d\mathbf{c} \mathbf{c}\mathbf{c}^2 f,\tag{2.22}$$

and we get

$$\begin{aligned}\partial_t \left(\rho\epsilon + \frac{1}{2}\rho\mathbf{u}^2 \right) + \nabla \cdot \left[\mathbf{u} \left(\rho\epsilon + \frac{1}{2}\rho\mathbf{u}^2 \right) + \mathbf{P} \cdot \mathbf{u} + \mathbf{q} \right] &= 0, \\ \rho \frac{d\epsilon}{dt} + \mathbf{P} : (\nabla\mathbf{u}) + \nabla \cdot \mathbf{q} &= 0,\end{aligned}\tag{2.23}$$

where the colon “:” denotes the full index contraction and where we used twice the continuity and once the momentum conservation equation. At this point we have the mass, momentum and energy conservation equations in general forms. To write down a closed set of equations, one has to impose some constitutive equations for the pressure tensor and for the internal energy flux.

The idea proposed in Grad [40] is to seek for solutions of the BGK equation (Eq. (2.8)) projected on an Hermite basis (see Grad [39] for a detailed study

of the Hermite polynomials basis). Grad's method leads to two additional sets of equations for the pressure tensor and for the energy flux. The whole system is known under the name of Grad's 13-moments equations. We will not enter the details of the derivation of Grad, but we will use his idea of projecting the distribution functions on Hermite polynomials as done in Shan et al. [92], Nie et al. [76]. We will then truncate the expansion in order to recover the desired hydrodynamic behavior.

2.2 Projection on the Hermite basis

The projection of the velocity distribution function on the Hermite polynomial basis is given by

$$f(\mathbf{x}, \boldsymbol{\xi}, t) = \omega(\boldsymbol{\xi}) \sum_{n=0}^{\infty} \frac{1}{n!} \mathcal{H}_{\boldsymbol{\alpha}}^{(n)}(\boldsymbol{\xi}) a_{\boldsymbol{\alpha}}^{(n)}(\mathbf{x}, t), \quad (2.24)$$

where $\omega(\boldsymbol{\xi})$ is a weight function, $\mathcal{H}_{\boldsymbol{\alpha}}^{(n)}$ and $a_{\boldsymbol{\alpha}}^{(n)}$ denote respectively the Hermite polynomial of degree n (see Appendix A for the definition of $\mathcal{H}_{\boldsymbol{\alpha}}^{(n)}$) and the expansion coefficient of degree n , the vector $\boldsymbol{\alpha}$ is a multi-index $\alpha_1 \dots \alpha_n$, and where we used Einstein's convention for the summation over all indices in $\boldsymbol{\alpha}$. The expansion coefficients are given by

$$a_{\boldsymbol{\alpha}}^{(n)}(\mathbf{x}, t) = \int d\boldsymbol{\xi} f(\mathbf{x}, \boldsymbol{\xi}, t) \mathcal{H}_{\boldsymbol{\alpha}}^{(n)}(\boldsymbol{\xi}). \quad (2.25)$$

The projection on the Hermite basis is natural because the expansion coefficients are given by the moments of a distribution function. Using the definitions of the moments of f (see Eqs. (2.2), (2.3) and (2.17)) one gets for $a^{(n)}$ ($n = 0, \dots, 4$)

$$a^{(0)} = \rho, \quad (2.26)$$

$$a_{\alpha}^{(1)} = \rho u_{\alpha}, \quad (2.27)$$

$$a_{\alpha\beta}^{(2)} = P_{\alpha\beta} + \rho (u_{\alpha} u_{\beta} - \delta_{\alpha\beta}), \quad (2.28)$$

$$a_{\alpha\beta\gamma}^{(3)} = Q_{\alpha\beta\gamma} + u_{\alpha} a_{\beta\gamma}^{(2)} + u_{\beta} a_{\alpha\gamma}^{(2)} + u_{\gamma} a_{\alpha\beta}^{(2)} + (1 - D) \rho u_{\alpha} u_{\beta} u_{\gamma}, \quad (2.29)$$

$$a_{\alpha\beta\gamma\delta}^{(4)} = R_{\alpha\beta\gamma\delta} - (P_{\alpha\beta} \delta_{\gamma\delta} + P_{\alpha\gamma} \delta_{\beta\delta} + P_{\alpha\delta} \delta_{\beta\gamma} + P_{\beta\gamma} \delta_{\alpha\delta} + P_{\beta\delta} \delta_{\gamma\alpha} + P_{\gamma\delta} \delta_{\alpha\beta}) \\ + (\delta_{\alpha\beta} \delta_{\gamma\delta} + \delta_{\alpha\gamma} \delta_{\beta\delta} + \delta_{\alpha\delta} \delta_{\beta\gamma}), \quad (2.30)$$

where \mathbf{Q} and \mathbf{R} are given by

$$\mathbf{Q} = \int d\mathbf{c} \mathbf{c} \mathbf{c} \mathbf{c} f, \quad \mathbf{R} = \int d\mathbf{c} \mathbf{c} \mathbf{c} \mathbf{c} \mathbf{c} f, \quad (2.31)$$

where \mathbf{ccc} and \mathbf{cccc} are the tensor products of \mathbf{c} with themselves. The internal energy tensor flux, \mathbf{Q} , is related to the energy flux \mathbf{q} by

$$q_\alpha = Q_{\alpha\beta\beta}.$$

The macroscopic equations obtained by Grad are only depending on five moments, ρ , $\rho\mathbf{u}$, $\rho\epsilon$, \mathbf{P} and \mathbf{q} . Therefore one can use an approximation of the distribution function f which would have the same moments and the macroscopic equation that would be recovered would be identical. We can therefore truncate the infinite series of Eq. (2.24) up to the desired order (four in the case of the Grad's 13-moments equations) to obtain the macroscopic equations without keeping all the information contained in the original distribution function

$$f(\mathbf{x}, \boldsymbol{\xi}, t) \cong f^N(\mathbf{x}, \boldsymbol{\xi}, t) = \omega(\boldsymbol{\xi}) \sum_{n=0}^N \frac{1}{n!} \mathcal{H}_\alpha^{(n)}(\boldsymbol{\xi}) a_\alpha^{(n)}(\mathbf{x}, t), \quad (2.32)$$

where N is the order of truncation. In order to analyze the effect of the loss of information due to the truncation at a certain order we will use the Chapman–Enskog expansion (see Chapman and Cowling [17]) that is presented in the next section.

2.3 Chapman–Enskog expansion

In this section we will first project the BGK equation (2.8) on the Hermite basis and then use a perturbative expansion to find expressions for the pressure tensor and the heat flux. This expansion follows the path used in the book by Huang [53] and is not the standard way it is done in the literature (e.g. Wolf-Gladrow [100] or Chopard et al. [20]). Nevertheless it keeps the basic ideas and is simpler to use. Before proceeding further we need to compute the expansion of the equilibrium distribution

$$f^{(0)} = \omega \sum_{n=0}^{\infty} \frac{1}{n!} \mathcal{H}_\alpha^{(n)} a_{0\alpha}^{(n)}, \quad \text{where } a_{0\alpha}^{(n)} = \int d\boldsymbol{\xi} f^{(0)} \mathcal{H}_\alpha^{(n)}. \quad (2.33)$$

The coefficients $a_0^{(n)}$ up to order four can be computed straightforwardly using Gaussian integration as

$$a_0^{(0)} = \rho, \quad (2.34)$$

$$a_{0\alpha}^{(1)} = \rho u_\alpha, \quad (2.35)$$

$$a_{0\alpha\beta}^{(2)} = \rho u_\alpha u_\beta + \rho(\theta - 1)\delta_{\alpha\beta}, \quad (2.36)$$

$$a_{0\alpha\beta\gamma}^{(3)} = \rho u_\alpha u_\beta u_\gamma + \rho(\theta - 1)(\delta_{\alpha\beta}u_\gamma + \delta_{\alpha\gamma}u_\beta + \delta_{\beta\gamma}u_\alpha), \quad (2.37)$$

$$\begin{aligned} a_{0\alpha\beta\gamma\delta}^{(4)} &= \rho u_\alpha u_\beta u_\gamma u_\delta + \rho(\theta - 1)^2(\delta_{\alpha\beta}\delta_{\gamma\delta} + \delta_{\alpha\gamma}\delta_{\beta\delta} + \delta_{\alpha\delta}\delta_{\beta\gamma}) \\ &\quad + \rho(\theta - 1)(\delta_{\alpha\beta}u_\gamma u_\delta + \delta_{\alpha\gamma}u_\beta u_\delta + \delta_{\alpha\delta}u_\beta u_\gamma \\ &\quad + \delta_{\beta\gamma}u_\alpha u_\delta + \delta_{\beta\delta}u_\alpha u_\gamma + \delta_{\gamma\delta}u_\alpha u_\beta). \end{aligned} \quad (2.38)$$

Let us first recall the BGK equation (2.8)

$$\partial_t f + \boldsymbol{\xi} \cdot \nabla_{\mathbf{x}} f + \mathbf{g} \cdot \nabla_{\boldsymbol{\xi}} f = -\frac{1}{\tau} (f - f^{(0)}). \quad (2.39)$$

The idea will be to project this equation on the Hermite basis. Using the relations of Appendix A, we can rewrite the space derivative and velocity derivative terms, projected on the polynomial of degree n , like

$$\int d\boldsymbol{\xi} \mathcal{H}_\alpha^{(n)} \boldsymbol{\xi} \cdot \nabla_{\mathbf{x}} f = \nabla_{\mathbf{x}} \cdot a_\alpha^{(n)} + (\nabla_{\mathbf{x}} a_\alpha^{(n-1)} + \text{perm}), \quad (2.40)$$

$$F(\boldsymbol{\xi}) \equiv \int d\boldsymbol{\xi} (\mathbf{g} \cdot \nabla_{\boldsymbol{\xi}})(\mathcal{H}_\alpha^{(n)} f) = (\mathbf{g} a_\alpha^{(n-1)} + \text{perm}), \quad (2.41)$$

where F represents the force term, “perm” represents all the cyclic index permutations and $\nabla_{\mathbf{x}}$ stands for the divergence. We assume in a first approximation that the distribution function is given by its equilibrium value. We therefore can rewrite the BGK equation (2.39) as

$$\partial_t f^{(0)} + \boldsymbol{\xi} \cdot \nabla_{\mathbf{x}} f^{(0)} - F^{\text{eq}} = 0, \quad (2.42)$$

where the F^{eq} is

$$F^{\text{eq}} = (\mathbf{g} a_{0\alpha}^{(n-1)} + \text{perm}). \quad (2.43)$$

Projecting this equation on the Hermite polynomial of degree n , and using Eqs. (2.40) and (2.41) we get

$$\partial_t a_0^{(n)} + \nabla \cdot a_0^{(n+1)} + (\nabla a_0^{(n-1)} + \text{perm}) - (\mathbf{g} a_0^{(n-1)} + \text{perm}) = 0, \quad (2.44)$$

For $n = 0, 1$ one finds

$$\partial_t \rho + \nabla \cdot (\rho \mathbf{u}) = 0, \quad (2.45)$$

$$\partial_t (\rho \mathbf{u}) + \nabla \cdot (\rho \theta \mathbf{I} + \rho \mathbf{u} \mathbf{u}) = \rho \mathbf{g}. \quad (2.46)$$

For $n = 2$, we take the trace of Eq. (2.44) and divide the result by two, and we find

$$\partial_t(\rho\epsilon) - \mathbf{u} \cdot \nabla(\rho\theta) + \left(\frac{D+2}{2}\right) \nabla \cdot (\rho\theta\mathbf{u}) = 0. \quad (2.47)$$

At this order of approximation one obtains the Euler equations for inviscid fluids. We now want to refine our approximation. We will assume that f is no longer exactly at equilibrium but close to it

$$f = f^{(0)} + f^{(1)}, \quad f^{(1)} \ll f^{(0)}. \quad (2.48)$$

Before proceeding further let us evaluate the order of magnitude of the “off-equilibrium” part $f^{(1)}$. Assuming that the characteristic length-scale of our system is L , and using that the characteristic speed of our particles is the speed of sound, c_s , we can approximate the l.h.s. of the BGK equation (2.39) as

$$\frac{df}{dt} \sim \frac{c_s}{L} f = -\frac{1}{\tau} (f - f^{(0)}), \quad (2.49)$$

Replacing f by the “perturbative” *Ansatz*, Eq. (2.48), in this last equation one has

$$f^{(0)} \sim -\frac{L}{c_s\tau} f^{(1)}, \quad (2.50)$$

where we used that $f^{(1)} \ll f^{(0)}$. Since $l = c_s\tau$ is the mean free path we immediately see that $f^{(1)}$ must scale like the Knudsen number³. Our perturbative approach is therefore valid in the limit of the small Knudsen numbers. We now can proceed and compute the off-equilibrium part of f ,

$$\partial_t f^{(0)} + \boldsymbol{\xi} \cdot \nabla_{\mathbf{x}} f^{(0)} - F^{\text{eq}} = -\frac{1}{\tau} f^{(1)}. \quad (2.51)$$

We will now reuse the Hermite polynomial representation, in order to be able to compute $f^{(1)}$

$$-\frac{1}{\tau} a_1^{(n)} = \partial_t a_0^{(n)} + \nabla \cdot a_0^{(n+1)} + \left(\nabla a_0^{(n-1)} + \text{perm} \right) - \left(\mathbf{g} a_0^{(n-1)} + \text{perm} \right), \quad (2.52)$$

where $a_{1\boldsymbol{\alpha}}^{(n)}$ is given by

$$a_{1\boldsymbol{\alpha}}^{(n)} = \int d\boldsymbol{\xi} f^{(1)} \mathcal{H}_{\boldsymbol{\alpha}}^{(n)}. \quad (2.53)$$

We know that the Hermite coefficients of the equilibrium distribution are given in terms of the conserved moments (mass ρ , momentum $\mathbf{j} = \rho\mathbf{u}$ and

³The Knudsen number, $\text{Kn} = l/L$, is the ratio between the mean free path and the characteristic lengthscale of the considered system.

internal energy $e = \rho\epsilon$). We can therefore rewrite the terms derived with respect to time in this last equation as

$$\partial_t a_0^{(n)} = \partial_\rho a_0^{(n)} \partial_t \rho + \partial_j a_0^{(n)} \cdot \partial_t (\rho \mathbf{u}) + \partial_e a_0^{(n)} \partial_t (\rho \epsilon), \quad (2.54)$$

$$\begin{aligned} &= -\partial_\rho a_0^{(n)} \nabla \cdot (\rho \mathbf{u}) + \partial_j a_0^{(n)} \cdot (\rho \mathbf{g} - \nabla \cdot (\rho \theta \mathbf{I} + \rho \mathbf{u} \mathbf{u})) \\ &\quad + \partial_e a_0^{(n)} \left((\mathbf{u} \cdot \nabla) \rho \theta - \left(\frac{D+2}{2} \right) \nabla \cdot (\rho \theta \mathbf{u}) \right), \end{aligned} \quad (2.55)$$

where in the last line we used the ‘‘conservation laws’’ computed above (Eqs. (2.45)-(2.47)) to replace the time derivatives with space derivatives. With Eqs. (2.34)-(2.38) one can compute the different ‘‘conserved moments derivatives’’ for $a_0^{(n)}$ for $n = 0, \dots, 3$

$$\partial_\rho a_0^{(0)} = 1, \quad \partial_\rho a_{0\alpha}^{(1)} = 0, \quad \partial_\rho a_{0\alpha\beta}^{(2)} = -u_\alpha u_\beta - \delta_{\alpha\beta}, \quad (2.56)$$

$$\partial_\rho a_{0\alpha\beta\gamma}^{(3)} = -2u_\alpha u_\beta u_\gamma - \theta(\delta_{\alpha\beta} u_\gamma + \delta_{\alpha\gamma} u_\beta + \delta_{\beta\gamma} u_\alpha), \quad (2.57)$$

$$\partial_{j_\alpha} a_0^{(0)} = 0, \quad \partial_{j_\beta} a_{0\alpha}^{(1)} = \delta_{\alpha\beta}, \quad \partial_{j_\gamma} a_{0\alpha\beta}^{(2)} = \delta_{\alpha\gamma} u_\beta + \delta_{\beta\gamma} u_\alpha, \quad (2.58)$$

$$\begin{aligned} \partial_{j_\delta} a_{0\alpha\beta\gamma}^{(3)} &= u_\alpha u_\beta \delta_{\gamma\delta} + u_\alpha u_\gamma \delta_{\beta\delta} + u_\beta u_\gamma \delta_{\alpha\delta} \\ &\quad + (\theta - 1)(\delta_{\alpha\beta} \delta_{\gamma\delta} + \delta_{\alpha\gamma} \delta_{\beta\delta} + \delta_{\beta\gamma} \delta_{\alpha\delta}), \end{aligned} \quad (2.59)$$

$$\partial_e a_0^{(0)} = \partial_{\rho\epsilon} a_{0\alpha}^{(1)} = 0, \quad \partial_e a_{0\alpha\beta}^{(2)} = \frac{2}{D} \delta_{\alpha\beta}, \quad (2.60)$$

$$\partial_e a_{0\alpha\beta\gamma}^{(3)} = \frac{2}{D} (\delta_{\alpha\beta} u_\gamma + \delta_{\alpha\gamma} u_\beta + \delta_{\beta\gamma} u_\alpha). \quad (2.61)$$

It is therefore straightforward to compute all the first-order coefficients

$$a_1^{(0)} = a_{1\alpha}^{(1)} = a_{1\alpha\alpha}^{(2)} = 0, \quad (2.62)$$

$$a_{1\alpha\beta}^{(2)} = -\tau \rho \theta \Lambda_{\alpha\beta}, \quad (2.63)$$

$$\begin{aligned} a_{1\alpha\beta\gamma}^{(3)} &= -\tau \rho \theta \left(\Lambda_{\alpha\beta} u_\gamma + \Lambda_{\alpha\gamma} u_\beta + \Lambda_{\beta\gamma} u_\alpha \right. \\ &\quad \left. + (\delta_{\alpha\beta} \partial_\gamma \theta + \delta_{\alpha\gamma} \partial_\beta \theta + \delta_{\beta\gamma} \partial_\alpha \theta) \right), \end{aligned} \quad (2.64)$$

where

$$\Lambda_{\alpha\beta} = \partial_\alpha u_\beta + \partial_\beta u_\alpha - \frac{2}{D} \partial_\gamma u_\gamma \delta_{\alpha\beta}. \quad (2.65)$$

Note that Eq. (2.62) is a consequence of the mass, momentum and energy conservation. We therefore can reconstruct the off-equilibrium distribution

function

$$f^{(1)} = \sum_{n=0}^3 \frac{1}{n!} \mathcal{H}_{\alpha}^{(n)} a_{1\alpha}^{(n)} \quad (2.66)$$

$$\begin{aligned} &= \left(\frac{1}{2} \mathcal{H}_{i\alpha\beta}^{(2)} a_{1\alpha\beta}^{(2)} + \frac{1}{6} \mathcal{H}_{i\alpha\beta\gamma}^{(3)} a_{1\alpha\beta\gamma}^{(3)} \right), \\ &= -\tau\rho\theta \left[\frac{1}{2} \mathcal{H}_{i\alpha\beta}^{(2)} \Lambda_{\alpha\beta} + \frac{1}{6} \mathcal{H}_{i\alpha\beta\gamma}^{(3)} \left(\Lambda_{\alpha\beta} u_{\gamma} + \Lambda_{\alpha\gamma} u_{\beta} \right. \right. \\ &\quad \left. \left. + \Lambda_{\beta\gamma} u_{\alpha} + (\delta_{\alpha\beta} \partial_{\gamma} \theta + \delta_{\alpha\gamma} \partial_{\beta} \theta + \delta_{\beta\gamma} \partial_{\alpha} \theta) \right) \right]. \quad (2.67) \end{aligned}$$

In order to close Eqs. (2.18) and (2.23) we have to compute the stress tensor and the heat flux, \mathbf{P} and \mathbf{q} . Using the perturbative expansion we rewrite them as

$$\mathbf{P} = \mathbf{P}^{(0)} + \mathbf{P}^{(1)}, \quad \mathbf{q} = \mathbf{q}^{(0)} + \mathbf{q}^{(1)}.$$

From Eqs. (2.17) and (2.22) we can compute $\mathbf{P}^{(n)}$ and $\mathbf{q}^{(n)}$

$$P_{\alpha\beta}^{(n)} = \int d\xi \left(\mathcal{H}_{\alpha\beta}^{(2)} - \mathcal{H}_{\alpha}^{(1)} u_{\beta} - \mathcal{H}_{\beta}^{(1)} u_{\alpha} + (\delta_{\alpha\beta} + u_{\alpha} u_{\beta}) \mathcal{H}^{(0)} \right) f^{(n)}, \quad (2.68)$$

$$\begin{aligned} q_{\alpha}^{(n)} &= \frac{1}{2} \int d\xi \left(\mathcal{H}_{\alpha\beta\beta}^{(3)} - 2\mathcal{H}_{\alpha\beta}^{(2)} u_{\beta} - \mathcal{H}_{\beta\beta}^{(2)} u_{\alpha} \right. \\ &\quad \left. + \mathcal{H}_{\alpha}^{(1)} u_{\beta} u_{\beta} + 2\mathcal{H}_{\beta}^{(1)} u_{\alpha} u_{\beta} \right. \\ &\quad \left. - \mathcal{H}^{(0)} (u_{\alpha} u_{\beta} u_{\beta} + (D+2)u_{\alpha}) \right) f^{(n)}. \quad (2.69) \end{aligned}$$

As a consequence $\mathbf{q}^{(0)} = 0$ and we have for $\mathbf{P}^{(0)}$

$$P_{\alpha\beta}^{(0)} = \rho\theta\delta_{\alpha\beta}. \quad (2.70)$$

We still have to compute $\mathbf{P}^{(1)}$ and $\mathbf{q}^{(1)}$, which are respectively given by

$$P_{\alpha\beta}^{(1)} = \int d\xi \mathcal{H}_{\alpha\beta}^{(2)} f^{(1)}, \quad (2.71)$$

$$q_{\alpha}^{(1)} = \frac{1}{2} \int d\xi \left(\mathcal{H}_{\alpha\beta\beta}^{(3)} - 2u_{\beta} \mathcal{H}_{\alpha\beta}^{(2)} \right) f^{(1)}. \quad (2.72)$$

Thanks to the orthogonality properties of the Hermite polynomials (see Appendix A) it is straightforward to compute these integrals

$$\mathbf{P}^{(1)} = -\tau\rho\theta\mathbf{\Lambda}, \quad (2.73)$$

$$\mathbf{q}^{(1)} = -\tau\rho\theta \left(\frac{D+2}{2} \right) \nabla\theta. \quad (2.74)$$

We therefore deduce the Navier–Stokes equations

$$\partial_t \rho + \nabla \cdot (\rho \mathbf{u}) = 0, \quad (2.75)$$

$$\rho \frac{d\mathbf{u}}{dt} + \nabla \cdot (p\mathbf{I} - \mu_s \mathbf{\Lambda}) = \rho \mathbf{g}, \quad (2.76)$$

$$\rho \frac{d\epsilon}{dt} + (p\mathbf{I} - \mu_s \mathbf{\Lambda}) : (\nabla \mathbf{u}) - \left(\frac{D+2}{2} \right) \nabla \cdot (\kappa \nabla \theta) = 0, \quad (2.77)$$

where we introduced the pressure p through the perfect gas law $p = \rho\theta$, and where the dynamic viscosity μ_s and thermal diffusivity κ are given by

$$\mu_s = \kappa = \rho\theta\tau. \quad (2.78)$$

We recognize above the thermal compressible Navier–Stokes equations. Nevertheless this model is a little bit too restrictive, since it only allows the viscosity to be equal to the diffusivity, and furthermore the bulk viscosity is null. These flaws are easily understandable, since the model only contains one transport parameter : the relaxation time τ . It is therefore natural that it cannot represent correctly all the macroscopic transport coefficients. A multiple relaxation time model has been proposed by Shan and Chen [90] that allows the viscosity to be different from the diffusivity coefficient.

We just showed that the BGK Boltzmann equation, taken in the small Knudsen number limit, leads to the compressible Navier–Stokes equations. We must also point out that in order to recover this macroscopic limit we only needed the truncation of the velocity distribution function up to order four in Hermite polynomials. We show now which limit we would have recovered if one had truncated this series down to a lower order.

2.3.1 Order three truncation

We now assume that f and $f^{(0)}$ are given by their third order expansion in Hermite polynomials

$$f = \omega \sum_{n=0}^3 \frac{1}{n!} \mathcal{H}_{\alpha}^{(n)} a_{\alpha}^{(n)}, \quad f^{(0)} = \omega \sum_{n=0}^3 \frac{1}{n!} \mathcal{H}_{\alpha}^{(n)} a_{0\alpha}^{(n)},$$

where the $a^{(n)}$ and $a_0^{(n)}$ are still given by Eqs. (2.26)-(2.29) and (2.34)-(2.37). Applying the same procedure as in the previous subsection, we want to compute $f^{(1)}$, which is given by Eq. (2.66). In order to compute it we therefore need the expansion coefficients up to order $n = 3$. The expansion coefficients for $n = 0, 1, 2$ remain unchanged and are given by Eqs. (2.62), (2.63).

Because of the orthogonality properties of the Hermite polynomials we have that $a_{0\alpha}^{(4)} = 0$, and Eq. (2.52) becomes

$$-\frac{1}{\tau}a_{1\alpha\beta\gamma}^{(3)} = \partial_t a_{0\alpha\beta\gamma}^{(3)} + \left(\partial_\alpha a_{0\beta\gamma}^{(2)} + \partial_\beta a_{0\alpha\gamma}^{(2)} + \partial_\gamma a_{0\alpha\beta}^{(2)} \right) - \left(g_\alpha a_{0\beta\gamma}^{(2)} + g_\beta a_{0\alpha\gamma}^{(2)} + g_\gamma a_{0\alpha\beta}^{(2)} \right). \quad (2.79)$$

Since all the terms on the r.h.s. side of this equation have been computed in the preceding subsection, it is convenient to rewrite this equation as

$$-\frac{1}{\tau} \left\{ a_{1\alpha\beta\gamma}^{(3)} \right\}^3 = -\frac{1}{\tau} \left\{ a_{1\alpha\beta\gamma}^{(3)} \right\}^4 - \partial_\delta \left\{ a_{0\alpha\beta\gamma\delta}^{(4)} \right\}^4, \quad (2.80)$$

where the notation $\left\{ \cdot \right\}^n$ is the Hermite coefficient computed from $f^{(0)}$ truncated at order n . This leads to

$$\begin{aligned} -\frac{1}{\tau} \left\{ a_{1\alpha\beta\gamma}^{(3)} \right\}^3 &= -\partial_\delta (\rho u_\alpha u_\beta u_\gamma u_\delta) \\ &+ \left\{ \rho\theta (u_\alpha \Lambda_{\beta\gamma} + u_\beta \Lambda_{\alpha\gamma} + u_\gamma \Lambda_{\alpha\beta}) \right. \\ &\quad - \partial_\alpha (\rho(\theta - 1)u_\beta u_\gamma) - \partial_\beta (\rho(\theta - 1)u_\alpha u_\gamma) \\ &\quad - \partial_\gamma (\rho(\theta - 1)u_\alpha u_\beta) - \delta_{\alpha\beta} \partial_\delta (\rho(\theta - 1)u_\gamma u_\delta) \\ &\quad \left. - \delta_{\alpha\gamma} \partial_\delta (\rho(\theta - 1)u_\beta u_\delta) - \delta_{\beta\gamma} \partial_\delta (\rho(\theta - 1)u_\alpha u_\delta) \right\} \\ &+ \rho\theta \left\{ \delta_{\alpha\beta} \partial_\gamma (\theta - \rho(\theta - 1)^2) + \delta_{\alpha\gamma} \partial_\beta (\theta - \rho(\theta - 1)^2) \right. \\ &\quad \left. + \delta_{\beta\gamma} \partial_\alpha (\theta - \rho(\theta - 1)^2) \right\}. \end{aligned} \quad (2.81)$$

Since the $a_1^{(n)}$ for $n = 0, 1, 2$ are the same as in the previous subsection, we conclude that the mass and momentum equations (Eqs. (2.75), (2.76)) are still valid. The differences appear in the energy conservation equation and in particular in the energy flux, \mathbf{q} . By substituting Eq. (2.80) in Eq. (2.66) and then computing the internal energy flux (see Eq. (2.72)), denoted $\mathbf{q}^{(1)'}$ we get

$$\begin{aligned} q_\alpha^{(1)'} &= q_\alpha^{(1)} + \frac{\tau}{2} \left\{ \partial_\delta (\rho u_\alpha u_\beta u_\gamma u_\delta) + (D + 2) \partial_\alpha (\rho(\theta - 1)^2) \right. \\ &\quad \left. + ((D + 4) \partial_\beta (\rho(\theta - 1)u_\alpha u_\beta) + \partial_\alpha (\rho(\theta - 1)u_\beta u_\beta)) \right\}, \end{aligned} \quad (2.82)$$

where $\mathbf{q}^{(1)}$ is given by Eq. (2.74). This result shows that the energy equation, although having a correct form does not reproduce accurately enough the heat flux and cannot be used for thermal fluids. Nevertheless, the formulation with order three truncation represents isothermal compressible fluids since the continuity and momentum conservation equations are still exact.

2.3.2 Order two truncation

The order two truncation is the most commonly used lattice Boltzmann model and has been discussed extensively in many works during the last twenty years (see among other Chen and Doolen [18] or Latt [66]). We will not enter into the details of the modifications and only give the major results that we will reuse in the sequel.

This model is only valid for an a-thermal and weakly compressible flows (low Mach number⁴). The equilibrium distribution function is given by

$$f^{(0)} = w_i \rho \left(1 + \boldsymbol{\xi} \cdot \mathbf{u} + \frac{1}{2} \mathcal{H}_{\alpha\beta}^{(2)} u_\alpha u_\beta \right). \quad (2.83)$$

Since there is no temperature in this model, the energy conservation is irrelevant and $\theta = 1$. As a consequence the time derivatives of the coefficients become

$$\partial_t a_0^{(n)} = -\partial_\rho a_0^{(n)} \nabla \cdot (\rho \mathbf{u}) + \partial_j a_0^{(n)} \cdot (\rho \mathbf{g} - \nabla \cdot (\rho \mathbf{I} + \rho \mathbf{u} \mathbf{u})). \quad (2.84)$$

And therefore from equation (2.52) one gets

$$a_{1\alpha\beta}^{(2)} = -2\tau \rho S_{\alpha\beta}, \quad (2.85)$$

where the strain rate tensor \mathbf{S} is given by

$$S_{\alpha\beta} = \frac{1}{2} (\partial_\alpha u_\beta + \partial_\beta u_\alpha). \quad (2.86)$$

We can apply the same procedure as in the third order truncation limit, but this time the $a_0^{(n)}$, $n = 3, 4$, coefficients will be zero and we will have to compute modifications in $a_1^{(2)}$. As done in (2.80) we can relate the second order truncated Hermite coefficient to the one of the third order

$$\begin{aligned} -\frac{1}{\tau} \left\{ a_{1\alpha\beta}^{(2)} \right\}^2 &= -\frac{1}{\tau} \left\{ a_{1\alpha\beta}^{(2)} \right\}^3 - \partial_\gamma \left\{ a_{0\alpha\beta\gamma}^{(3)} \right\}^3 \\ &= 2\rho S_{\alpha\beta} - \partial_\gamma (\rho u_\alpha u_\beta u_\gamma), \\ &= 2\rho S_{\alpha\beta} + \mathcal{O}(\text{Ma}^3). \end{aligned} \quad (2.87)$$

Using the low Ma number approximation, the non-equilibrium distribution is given by

$$f^{(1)} = -\tau \rho \mathcal{H}_{i\alpha\beta}^{(2)} S_{\alpha\beta}, \quad (2.88)$$

⁴The Mach number, $\text{Ma} = u/c_s$, is the ratio between the characteristic velocity of the fluid and the speed of sound.

which allows us to compute $\mathbf{P}^{(1)}$

$$P_{\alpha\beta}^{(1)} = -2\tau\rho S_{\alpha\beta}. \quad (2.89)$$

This gives us the following pressure tensor

$$P_{\alpha\beta} = \rho\delta_{\alpha\beta} - 2\mu_s S_{\alpha\beta}, \quad (2.90)$$

where $\mu_s = \rho\tau$ is the dynamic viscosity. Finally we recover the following macroscopic equations

$$\begin{aligned} \partial_t \rho + \nabla \cdot (\rho \mathbf{u}) &= 0, \\ \rho \frac{d\mathbf{u}}{dt} + \nabla \cdot (p\mathbf{I} - 2\mu_s \mathbf{S}) &= \rho \mathbf{g}. \end{aligned} \quad (2.91)$$

2.4 Discretization of velocity space

In the preceding subsection, we derived the macroscopic equations obtained from the BGK Boltzmann equation for different truncation orders. We will now use these truncated distribution function descriptions to discretize the velocity space in a convenient manner. Since the macroscopic equations only depend on the Hermite expansion coefficients $a_{\alpha}^{(n)}$ and can be exactly recovered with the truncated f^N (N is the truncation order) instead of the complete f , it follows that Eq. (2.25) can be rewritten as

$$a_{\alpha}^{(n)}(\mathbf{x}, t) = \int d\boldsymbol{\xi} f^N(\mathbf{x}, \boldsymbol{\xi}, t) \mathcal{H}_{\alpha}^{(n)}(\boldsymbol{\xi}) = \int d\boldsymbol{\xi} \omega(\boldsymbol{\xi}) r(\mathbf{x}, \boldsymbol{\xi}, t), \quad (2.92)$$

where $r(\mathbf{x}, \boldsymbol{\xi}, t)$ is a polynomial function of $\boldsymbol{\xi}$ of degree not greater than $2N$. This integral can be *exactly* evaluated as a weighted sum using the Gauss–Hermite quadrature (see the book by Krylov [61] and Appendix B)

$$a_{\alpha}^{(n)}(\mathbf{x}, t) = \sum_{i=0}^{q-1} w_i r(\mathbf{x}, \boldsymbol{\xi}_i, t) = \sum_{i=0}^{q-1} \frac{w_i}{\omega(\boldsymbol{\xi}_i)} f^N(\mathbf{x}, \boldsymbol{\xi}_i, t) \mathcal{H}_{\alpha}^{(n)}(\boldsymbol{\xi}_i), \quad (2.93)$$

where w_i , $\boldsymbol{\xi}_i$ ($i = 0, \dots, q-1$) are the weights and abscissae of the Gauss–Hermite quadrature of a degree $\leq 2N$ (see Appendix B). Therefore the discretized $f^N(\mathbf{x}, \boldsymbol{\xi}_i, t)$, ($i = 0, \dots, q-1$) completely determine $f^N(\mathbf{x}, \boldsymbol{\xi}, t)$ and its moments. As can be seen in Appendix B the abscissae are not of unit length as in the standard lattice Boltzmann method. In order to be consistent with these models we have to rescale all the velocities by a factor c_l , which can be seen as a lattice sound velocity scaling factor. This will have

as an effect to give slightly different Hermite polynomials and expansion coefficients. In the sequel we will use this rescaling in order to give the right implementation formulas (see Appendix A).

In order to finish the discretization of Eq. (2.39) one still has to project $f^{(0)}$ on the truncated Hermite basis. The expansion coefficients are given by

$$a_0^{(0)} = \rho, \quad (2.94)$$

$$a_{0\alpha}^{(1)} = \rho u_\alpha, \quad (2.95)$$

$$a_{0\alpha\beta}^{(2)} = \rho u_\alpha u_\beta + c_l^2 \rho (\theta - 1) \delta_{\alpha\beta}, \quad (2.96)$$

$$a_{0\alpha\beta\gamma}^{(3)} = \rho u_\alpha u_\beta u_\gamma + c_l^2 \rho (\theta - 1) (\delta_{\alpha\beta} u_\gamma + \delta_{\alpha\gamma} u_\beta + \delta_{\beta\gamma} u_\alpha), \quad (2.97)$$

$$\begin{aligned} a_{0\alpha\beta\gamma\delta}^{(4)} = & \rho u_\alpha u_\beta u_\gamma u_\delta + c_l^4 \rho (\theta - 1)^2 (\delta_{\alpha\beta} \delta_{\gamma\delta} + \delta_{\alpha\gamma} \delta_{\beta\delta} + \delta_{\alpha\delta} \delta_{\beta\gamma}) \\ & + c_l^2 \rho (\theta - 1) (\delta_{\alpha\beta} u_\gamma u_\delta + \delta_{\alpha\gamma} u_\beta u_\delta + \delta_{\alpha\delta} u_\beta u_\gamma \\ & + \delta_{\beta\gamma} u_\alpha u_\delta + \delta_{\beta\delta} u_\alpha u_\gamma + \delta_{\gamma\delta} u_\alpha u_\beta). \end{aligned} \quad (2.98)$$

Defining f_i as

$$f_i(\mathbf{x}, t) \equiv \frac{w_i}{\omega(\boldsymbol{\xi}_i)} f(\mathbf{x}, \boldsymbol{\xi}_i, t). \quad (2.99)$$

we can rewrite the discretized equilibrium distribution function, truncated at order four, as

$$\begin{aligned} f_i^{(0)} &= w_i \sum_{n=0}^4 \frac{1}{c_l^{2n} n!} a_{0\alpha}^{(n)} \mathcal{H}_{i\alpha}^{(n)}, \\ &= w_i \rho \left(1 + \frac{\boldsymbol{\xi}_i \cdot \mathbf{u}}{c_l^2} + \frac{1}{2c_l^4} [(\boldsymbol{\xi}_i \cdot \mathbf{u})^2 - c_l^2 \mathbf{u}^2 + c_l^2 (\theta - 1) (\xi_i^2 - c_l^2 D)] \right. \\ &\quad + \frac{\boldsymbol{\xi}_i \cdot \mathbf{u}}{6c_l^6} [(\boldsymbol{\xi}_i \cdot \mathbf{u})^2 - 3c_l^2 \mathbf{u}^2 + 3c_l^2 (\theta - 1) (\xi_i^2 - c_l^2 (D + 2))] \\ &\quad + \frac{1}{24c_l^8} [(\boldsymbol{\xi}_i \cdot \mathbf{u})^4 - 6c_l^2 \mathbf{u}^2 (\boldsymbol{\xi}_i \cdot \mathbf{u})^2 + 3c_l^4 \mathbf{u}^4 \\ &\quad + 6c_l^2 (\theta - 1) ((\boldsymbol{\xi}_i \cdot \mathbf{u})^2 (\xi_i^2 - c_l^2 (D + 4)) + c_l^2 \mathbf{u}^2 (c_l^2 (D + 2) - \xi_i^2)) \\ &\quad \left. + 3c_l^4 (\theta - 1)^2 (\boldsymbol{\xi}_i^4 - 2c_l^2 (D + 2) \xi_i^2 + c_l^4 D (D + 2)) \right], \end{aligned} \quad (2.100)$$

where $\mathcal{H}_{i\alpha}^{(n)}$ are the discretized Hermite polynomials. With these definitions

we can also write the discretized force term, F_i (see Eq. (2.41))

$$\begin{aligned}
F_i &= w_i \sum_{n=1}^3 \frac{1}{n! c_l^{2n}} \mathcal{H}_{i\alpha}^{(n)} a_{0\alpha}^{(n)}, \\
&= w_i \rho \left(\frac{\xi_{i\alpha} g_\alpha}{c_l^2} + \frac{\mathcal{H}_{i\alpha\beta}^{(2)}}{c_l^4} g_\alpha u_\beta + \frac{\mathcal{H}_{i\alpha\beta\gamma}^{(3)}}{2c_l^6} g_\alpha (u_\beta u_\gamma + c_l^2 (\theta - 1) \delta_{\beta\gamma}) \right) \\
&= w_i \rho \left(\frac{\boldsymbol{\xi}_i \cdot \mathbf{g}}{c_l^2} + \frac{1}{2c_l^4} ((\boldsymbol{\xi}_i \cdot \mathbf{u})(\boldsymbol{\xi}_i \cdot \mathbf{g}) - c_l^2 (\mathbf{g} \cdot \mathbf{u})) \right. \\
&\quad \left. + \frac{1}{6c_l^6} [(\boldsymbol{\xi}_i \cdot \mathbf{g}) ((\boldsymbol{\xi}_i \cdot \mathbf{u})^2 + c_l^2 \boldsymbol{\xi}_i^2 (\theta - 1)) \right. \\
&\quad \left. - c_l^2 [(\boldsymbol{\xi}_i \cdot \mathbf{u}) (2(\mathbf{g} \cdot \mathbf{u}) + (\mathbf{u}^2 + c_l^2 (\theta - 1) D)) \right. \\
&\quad \left. + 2c_l^2 (\theta - 1) (\boldsymbol{\xi}_i \cdot \mathbf{g})] \right]. \tag{2.101}
\end{aligned}$$

Finally, we can also rewrite the off-equilibrium term of the distribution function

$$\begin{aligned}
f_i^{(1)} &= -w_i \frac{\tau \rho \theta}{c_s^2} \left[\frac{1}{2} \mathcal{H}_{i\alpha\beta}^{(2)} \Lambda_{\alpha\beta} + \frac{1}{6c_s^2} \mathcal{H}_{i\alpha\beta\gamma}^{(3)} (\Lambda_{\alpha\beta} u_\gamma + \Lambda_{\alpha\gamma} u_\beta \right. \\
&\quad \left. + \Lambda_{\beta\gamma} u_\alpha + c_s^2 (\delta_{\alpha\beta} \partial_\gamma \theta + \delta_{\alpha\gamma} \partial_\beta \theta + \delta_{\beta\gamma} \partial_\alpha \theta) \right]. \tag{2.102}
\end{aligned}$$

We now have the BGK equation, discretized in velocity space, for f^N (we omit the superscript N in the sequel) which reads

$$\partial_t f_i + \boldsymbol{\xi}_i \cdot \nabla f_i = -\frac{1}{\tau} (f_i - f_i^{(0)}) + F_i. \tag{2.103}$$

The different moments that we used to describe the hydrodynamic observables are no longer computed as integrals over an infinite velocity set, but as

weighted sums over a finite number of discrete velocities

$$\rho = \sum_i f_i, \quad (2.104)$$

$$\rho \mathbf{u} = \sum_i \boldsymbol{\xi}_i f_i, \quad (2.105)$$

$$\rho \epsilon = \frac{1}{2} \sum_i \mathbf{c}_i^2 f_i, \quad (2.106)$$

$$\mathbf{P} = \sum_i \mathbf{c}_i \mathbf{c}_i f_i, \quad (2.107)$$

$$\mathbf{q} = \frac{1}{2} \sum_i \mathbf{c}_i \mathbf{c}_i^2 f_i, \quad (2.108)$$

where $\mathbf{c}_i = \boldsymbol{\xi}_i - \mathbf{u}$ is the discrete counterpart of the continuous microscopic velocity vector in the comoving frame. It must be noted that because of the renormalization of the Hermite basis we have the following expressions for the viscosity, diffusivity and the perfect gas law

$$\mu_s = c_i^2 \rho \theta \tau, \quad \kappa = c_i^4 \rho \theta \tau, \quad p = c_i^2 \rho \theta. \quad (2.109)$$

The degree of the quadrature determines the number of abscissae that have to be used in order to evaluate exactly the Hermite coefficients. As seen in the previous subsection, depending on the physics one wants to represent we need a different number of these coefficients to be computed exactly. In the incompressible case we need them up to order two, but up to order four in the thermal compressible case. Thus for incompressible flows one will use the degree five quadratures (the standard lattices) $E_{2,5}^9$ or $E_{3,5}^{19}$ depending on the physical dimension, for isothermal flows the order seven quadrature $E_{2,7}^{17}$ or $E_{3,7}^{39}$ and finally for the thermal case the order nine $E_{2,9}^{37}$ or $E_{3,9}^{121}$ (see Appendix B for the definitions of the lattices $E_{D,m}^Q$). Thus, increasing the quadrature degree, implies huge differences in terms of the computational power needed to perform the simulations. In two dimensions the memory needs per node can be increased by a factor four whereas in three dimensions it can reach a factor six. The price to pay for the simulation of more and more complex cases is therefore quite high.

2.5 Time and space discretization

After discretizing the velocity space of the BGK equation, we are left with the discretization of the convective term of (2.103). The idea for the time dis-

cretization was first used in He et al. [49] and can also be found in Dellar [25].

We rewrite the discretized BGK Boltzmann equation (Eq. (2.103))

$$\frac{df_i}{dt} = -\frac{1}{\tau} (f_i - f_i^{\text{eq}}),$$

where $f_i^{\text{eq}} \equiv f_i^{(0)} + \tau F_i$. By direct integration of this equation one finds

$$\begin{aligned} f_i(\mathbf{x} + \boldsymbol{\xi}\Delta t, t + \Delta t) - f_i(\mathbf{x}, t) = \\ -\frac{1}{\tau} \int_0^{\Delta t} ds (f_i(\mathbf{x} + \boldsymbol{\xi}s, t + s) - f_i^{\text{eq}}(\mathbf{x} + \boldsymbol{\xi}s, t + s)), \end{aligned} \quad (2.110)$$

where Δt is the time increment to go from one timelevel to the next one. The r.h.s. of this last equation can be approximated using the trapezoidal rule

$$\begin{aligned} f_i(\mathbf{x} + \boldsymbol{\xi}_i\Delta t, t + \Delta t) - f_i(\mathbf{x}, t) = \\ -\frac{\Delta t}{2\tau} (f_i(\mathbf{x} + \boldsymbol{\xi}_i\Delta t, t + \Delta t) - f_i^{\text{eq}}(\mathbf{x} + \boldsymbol{\xi}_i\Delta t, t + \Delta t) \\ + f_i(\mathbf{x}, t) - f_i^{\text{eq}}(\mathbf{x}, t)) + \mathcal{O}(\Delta t^2). \end{aligned} \quad (2.111)$$

This second order approximation is implicit, but can be rendered explicit thanks to an appropriate change of variables ($\Delta t = 1$ from now on)

$$\bar{f}_i(\mathbf{x}, t) = f_i(\mathbf{x}, t) + \frac{1}{2\tau} (f_i(\mathbf{x}, t) - f_i^{\text{eq}}(\mathbf{x}, t)). \quad (2.112)$$

Using this relation in Eq. (2.111) one gets

$$\bar{f}_i(\mathbf{x} + \boldsymbol{\xi}_i, t + 1) - \bar{f}_i(\mathbf{x}, t) = -\frac{1}{\bar{\tau}} (\bar{f}_i(\mathbf{x}, t) - f_i^{\text{eq}}(\mathbf{x}, t)). \quad (2.113)$$

which is now an explicit equation and where we defined

$$\bar{\tau} = \tau + \frac{1}{2}. \quad (2.114)$$

We will therefore simulate these \bar{f}_i using the modified relaxation time $\bar{\tau}$. The reader should remark that although this time-stepping looks like a first-order upwind finite difference scheme for the time discretization, it is in reality a second order approximation since the BGK equation has been approximated using the trapezoidal rule.

We still have to see what is the effect of the change of variables on the moments of the distribution function. The first straightforward effect that can be noticed is that we have to substitute τ by $\bar{\tau} - 1/2$ in the computation of the viscosity and of the diffusivity. We therefore have

$$\mu_s = c_i^2 \rho \theta \left(\bar{\tau} - \frac{1}{2} \right), \quad \kappa = c_i^4 \rho \theta \left(\bar{\tau} - \frac{1}{2} \right). \quad (2.115)$$

We see appearing a so-called ‘‘lattice viscosity’’ (respectively diffusivity) which is due to this change of variables that allows us to have a second order explicit time-stepping. It can be seen as the price to pay to have a better time discretization.

Since f_i is now expressed in terms of \bar{f}_i we should see what changes are implied for the different macroscopic variables. Let us compute $\bar{\rho}$

$$\bar{\rho} \equiv \sum_{i=0}^{q-1} \bar{f}_i = \rho, \quad (2.116)$$

where the last equality is the consequence of the fact that $\sum_i f_i = \sum_i f_i^{(0)}$ and that $\sum_i F_i = 0$. For the momentum we have

$$\rho \bar{\mathbf{u}} = \sum_i \boldsymbol{\xi}_i \bar{f}_i = \rho \mathbf{u} - \rho \mathbf{g} / 2. \quad (2.117)$$

We see therefore that the velocity in terms of \bar{f}_i , that we will now note $\bar{\mathbf{u}}$ is not anymore the ‘‘physical velocity’’ which is the one given by the f_i but is computed as

$$\mathbf{u} = \bar{\mathbf{u}} + \frac{\mathbf{g}}{2}. \quad (2.118)$$

The reader should therefore be aware that the physical velocity should always be computed this way. Furthermore when doing the Chapman–Enskog expansion of the discrete scheme (see the books by Wolf-Gladrow [100] or Succi [94]) the equilibrium distribution function should be taken with respect to this velocity in order to get coherent results. Furthermore in order to be able to compare the scheme with the *a posteriori* scheme proposed in Guo et al. [45] we have to remember that f_i^{eq} is given in terms of the physical velocity and the physical relaxation time τ , which implies that the r.h.s. of the numerical scheme (2.113) is given by

$$\begin{aligned} & -\frac{1}{\tau} \left(\bar{f}_i(\mathbf{x}, t) - f_i^{(0)}(\rho, \mathbf{u}) - \tau F_i(\rho, \mathbf{u}) \right), \\ & = -\frac{1}{\tau} \left(\bar{f}_i(\mathbf{x}, t) - f_i^{(0)}(\rho, \mathbf{u}) \right) + \left(1 - \frac{1}{2\bar{\tau}} \right) F_i(\rho, \mathbf{u}). \end{aligned} \quad (2.119)$$

We therefore see that at order two in Hermite polynomials the scheme is the same to the one proposed by Guo, but in our case was derived completely *a priori*. In the sequel we will only use the overlined distribution functions and relaxation times, and therefore we will omit this notation for simplicity.

2.6 Summary

In this section, we have shown how to discretize the BGK Boltzmann equation, in order to simulate the Navier–Stokes equations (also shown in the diagram of Fig. 2.1). We have started by averaging the continuous BGK equation and found the conservation equations (mass, momentum, and energy) but with generic pressure tensor and energy flux. In order to close the system (find constitutive equations for the pressure tensor and the energy flux), following the idea of Grad, we chose to project the BGK equation on the Hermite basis and truncate the expansion. Then we have carried out Chapman–Enskog expansion of this truncated Hermite basis projection. Using this expansion, the Newton’s law for the pressure tensor and the Fourier law for the energy flux have been obtained. Furthermore it has allowed us to relate the relaxation time, with the viscosity and thermal diffusivity. We have also been able to express the distribution function in terms of macroscopic quantities. We have then discretized the velocity space using the Gauss–Hermite quadrature rule to evaluate the integrals as finite weighted sums. The order of quadrature needed is obviously related to the order of quadrature used to express the distribution function. Finally we have discretized the equations in time and space by a second order trapezoidal integration. This implicit scheme was cast into an explicit one by a change of variables, which had as an effect to add a lattice viscosity and thermal diffusivity to the physical ones. In this thesis we restrict ourselves to incompressible fluids, and therefore use only the second order truncated Hermite representation. Below are given the formulae that we have used for implementation purpose. The different Hermite coefficients are obtained from Eqs. (2.94)-(2.96) by setting $\theta \equiv 1$ and (2.87)

$$a_0^{(0)} = \rho, \quad a_{0\alpha}^{(1)} = \rho u_\alpha, \quad a_{0\alpha\beta}^{(2)} = \rho u_\alpha u_\beta, \quad (2.120)$$

$$a_1^{(0)} = 0, \quad a_{1\alpha}^{(1)} = 0, \quad a_{1\alpha\beta}^{(2)} = -2\tau\rho S_{\alpha\beta}. \quad (2.121)$$

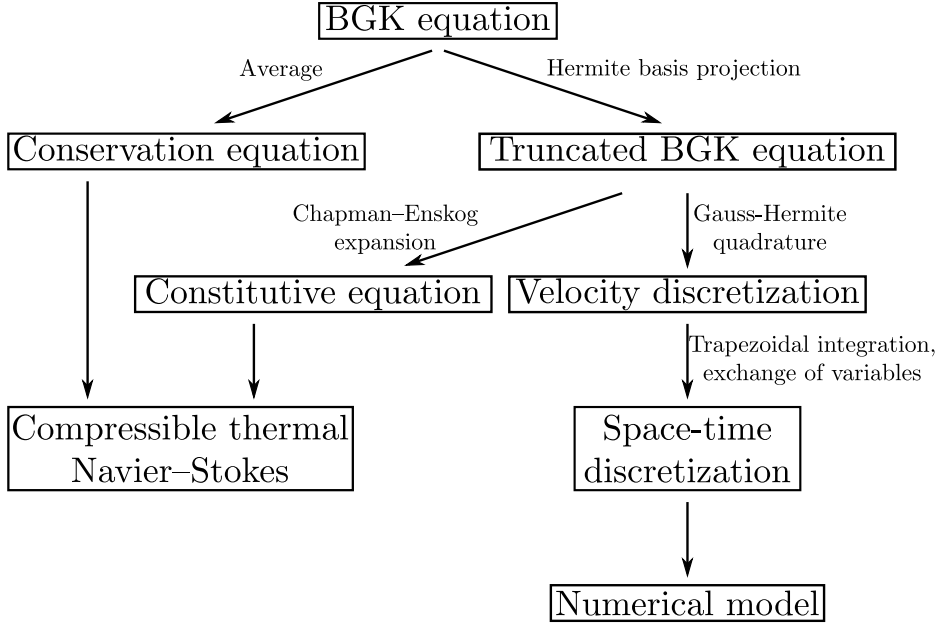


Fig. 2.1: A summary of the path followed to build the numerical scheme from the continuous Boltzmann equation to simulate the compressible thermal Navier–Stokes equations.

And therefore the $f_i^{(0)}$, $f_i^{(1)}$ and F_i are given by

$$f_i^{(0)} = w_i \rho \left(1 + \frac{\boldsymbol{\xi}_i \cdot \mathbf{u}}{c_l^2} + \frac{1}{2c_l^4} \mathcal{H}_{i\alpha\beta}^{(2)} u_\alpha u_\beta \right), \quad (2.122)$$

$$f_i^{(1)} = -\frac{w_i \rho}{c_l^2} \mathcal{H}_{i\alpha\beta}^{(2)} S_{\alpha\beta}, \quad (2.123)$$

$$F_i = w_i \rho \left(\frac{\boldsymbol{\xi}_i - \mathbf{u}}{c_l^2} - \frac{(\boldsymbol{\xi}_i \cdot \mathbf{u}) \boldsymbol{\xi}_i}{c_l^4} \right) \cdot \mathbf{g}, \quad (2.124)$$

where \mathbf{u} is the physical velocity of Eq. (2.118). The pressure tensor is given by

$$\mathbf{P}^{(0)} = c_l^2 \rho \mathbf{I}, \quad \mathbf{P}^{(1)} = -\frac{\tau}{2c_l^4} \mathbf{S}. \quad (2.125)$$

Finally the numerical scheme is given by

$$f_i(\mathbf{x} + \boldsymbol{\xi}_i, t+1) - f_i(\mathbf{x}, t) = -\frac{1}{\tau} \left(f_i(\mathbf{x}, t) - f_i^{(0)}(\rho, \mathbf{u}) \right) + \left(1 - \frac{1}{2\tau} \right) F_i. \quad (2.126)$$

If we now turn our attention to the implementation of this method, we see that it can be separated in two distinct steps : the collision and the

propagation. Define f_i^{in} and f_i^{out} as the f_i 's before and after the collision we have the two following steps

1. The collision :

$$f_i^{\text{out}}(\mathbf{x}, t) = f_i^{\text{in}}(\mathbf{x}, t) - \frac{1}{\tau} \left(f_i^{\text{in}}(\mathbf{x}, t) - f_i^{(0)}(\rho, \mathbf{u}) \right) + \left(1 - \frac{1}{2\tau} \right) F_i, \quad (2.127)$$

2. The propagation :

$$f_i^{\text{in}}(\mathbf{x} + \boldsymbol{\xi}_i, t + 1) = f_i^{\text{out}}. \quad (2.128)$$

Each one of these steps must be applied to the whole system before the next one can take place. The collision is a completely local operation while the propagation only communicates with a few neighbors. This “locality” of the scheme is particularly efficient for parallelization.

Chapter 3

Non-Newtonian fluid dynamics

In the last chapter we presented a general derivation of the equations governing the fluid dynamics and faced a closure problem for the stress tensor and the heat flux. We now consider only isothermal flows and therefore focus our attention only on the stress tensor. The single relaxation time model, although being able to reproduce the complex physics behind compressible and thermal flows can only describe Newtonian fluids (i.e. $\mathbf{P} = p\mathbf{I} - \mu_s\mathbf{\Lambda}$, with $\mu_s = \text{cte}$ and $\mathbf{\Lambda}$ the extra-stress tensor). The sign convention used here is the same as in Bird et al. [8], namely that negative (respectively positive) normal stress corresponds to extension (respectively compression). In this chapter, we will present more complicated closure (constitutive) relations which will describe the mechanical behavior of more complex (viscoelastic) fluids. We will also assume that the fluids are incompressible.

3.1 Generalized Newtonian fluids

The viscosity will be no longer considered as a constant viscosity, as in the Newtonian model, but rather strain rate dependent. This leads to a stress tensor

$$\mathbf{P} = p\mathbf{I} - 2\mu_s(\dot{\gamma})\mathbf{S}, \quad (3.1)$$

where the shear rate $\dot{\gamma}$ is defined as

$$\dot{\gamma} = \sqrt{2\mathbf{S} : \mathbf{S}}. \quad (3.2)$$

These models describe successfully fluids like molten polymers, uncooked paste or toothpaste. There exist a lot of different models for the dependence of μ_s on $\dot{\gamma}$ (see Agassant et al. [1] for a more complete list). One of the more popular is the “power-law” model

$$\mu_s(\dot{\gamma}) = k\dot{\gamma}^{n-1}, \quad (3.3)$$

where n is the non-dimensional power-law index, and k is a constant¹. In the limit $n = 1$ we recover the Newtonian case with $\mu_s = k$. The cases $n < 1$

¹The dimensions of k are dependent on n and are $[\text{kg} \cdot \text{m}^{-1} \cdot \text{s}^{n-2}]$.

and $n > 1$ represent respectively pseudoplastic (shear-thinning) and dilatant (shear-thickening) fluids. The “power-law” fluids fail to represent real fluids in the limits of small and high shear rate. Especially in the case where $n < 1$, we have that $\mu_s \rightarrow \infty$ when $\dot{\gamma} \rightarrow 0$, which of course is unphysical. To correct these flaws other models have been proposed where we have two Newtonian plateaus at low and high shear rate. In the Carreau–Yasuda which model exhibits such a behavior we define

$$\frac{\mu_s - \mu_\infty}{\mu_0 - \mu_\infty} = (1 + (\lambda\dot{\gamma})^a)^{(n-1)/a}, \quad (3.4)$$

where λ is a time constant, a a dimensionless number, μ_0 and μ_∞ are the viscosities at zero and infinite shear rate.

3.2 Viscoelastic fluids

The second class of non-Newtonian fluids that are presented here are those that exhibit memory effects. This section follows the monograph of Lin [70]. For more informations the reader could also refer to Bird et al. [8], Larson [63] or Fiétier [33]. The models presented here are not representing correctly real viscoelastic fluid for all type of flows. In particular the Oldroyd-B model is limited to small deformation rates whereas the FENE-P model is of low accuracy for highly elongational flows. Although they are not able to represent correctly liquids that are subject to high deformations, their simplicity make them good candidates for implementation and testing purposes. These models are based on the representation of the polymers as dumbbells connected by a spring that interact with a Newtonian solvent (see Fig. 3.1). We will first present the effect of the solvent on the dumbbells and therefore their evolution equation, and then we will discuss the action of the dumbbells on the flow.

3.2.1 Action of the solvent on the polymers

The beads have mass m and radius a . Each bead experiences a force due to the velocity difference between itself and the fluid surrounding it. This force (Stokes formula) is opposite to the velocity difference and has a proportionality constant, the mobility, $\zeta = 6\pi\mu_s a$, where μ_s is the solvent dynamic viscosity. It is furthermore assumed that there are ρ_p dumbbells per unit volume. We also introduce \mathbf{r} (see Fig. 3.1) the configuration vector connecting two beads of position \mathbf{r}_1 and \mathbf{r}_2

$$\mathbf{r} = \mathbf{r}_2 - \mathbf{r}_1. \quad (3.5)$$

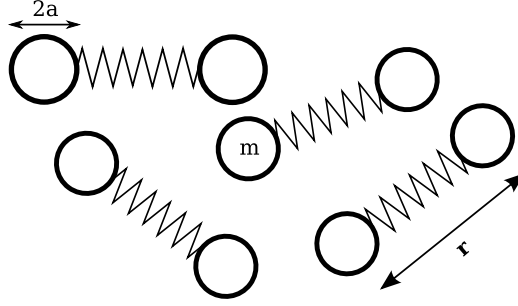


Fig. 3.1: The polymers are modelled by dumbbells connected by a spring.

The position of the center of mass of the dumbbell, \mathbf{r}_c , is defined as

$$\mathbf{r}_c = \frac{\mathbf{r}_1 + \mathbf{r}_2}{2}. \quad (3.6)$$

It is assumed that the fluid has a homogeneous velocity field in the vicinity of the beads, which is a reasonable assumption if the dumbbell characteristic size is low compared to the length scale of the velocity variations. We will also define the probability distribution $\psi(\mathbf{r}, t)$, such that $\psi(\mathbf{r}, t)d\mathbf{r}$ is the probability that a randomly chosen dumbbell has a configuration vector $\mathbf{r} \in [\mathbf{r}, \mathbf{r} + d\mathbf{r}]$. We also normalize ψ in order to have

$$\int d\mathbf{r} \psi(\mathbf{r}, t) = 1. \quad (3.7)$$

With these assumptions one can show (see Lin [70]) that the time evolution

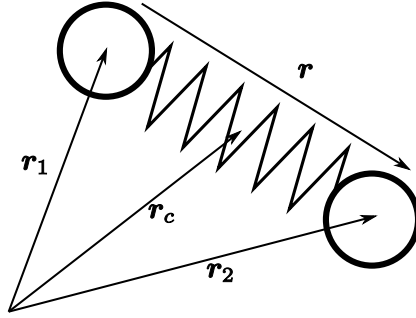


Fig. 3.2: The beads are at positions \mathbf{r}_1 and \mathbf{r}_2 respectively, while \mathbf{r}_c is the position of the center of mass and \mathbf{r} is the configuration vector.

of ψ is given by a Smoluchowski equation

$$\partial_t \psi = -\nabla_{\mathbf{r}} \cdot \left[\left((\nabla_{\mathbf{x}} \mathbf{u})^T \cdot \mathbf{r} \right) \psi - \frac{2\theta}{\zeta} \nabla_{\mathbf{r}} \psi - \frac{2\mathbf{s}\psi}{\zeta} \right], \quad (3.8)$$

where \mathbf{s} is the connector (spring) force, which has the same direction as \mathbf{r} . We are now interested in the averaged quantities. As in the preceding chapter we will define the average of the observable \mathcal{A} is defined as

$$\langle \mathcal{A}(t) \rangle = \int d\mathbf{r} \mathcal{A}(\mathbf{r}) \psi(\mathbf{r}, t). \quad (3.9)$$

As a consequence we have that the time derivative of this averaged observable is

$$\frac{d\langle \mathcal{A} \rangle}{dt} = \int d\mathbf{r} \mathcal{A} \partial_t \psi. \quad (3.10)$$

Substituting Eq. (3.8) in this last equation and integrating by parts, we obtain

$$\begin{aligned} \frac{d\langle \mathcal{A} \rangle}{dt} = & - \int d\mathbf{r} \nabla_{\mathbf{r}} \cdot \left[\mathcal{A} \left(\left((\nabla_{\mathbf{x}} \mathbf{u})^{\text{T}} \cdot \mathbf{r} \right) \psi - \frac{2\theta}{\zeta} \nabla_{\mathbf{r}} \psi - \frac{2\mathbf{s}\psi}{\zeta} \right) \right] \\ & + \int d\mathbf{r} \left(\left((\nabla_{\mathbf{x}} \mathbf{u})^{\text{T}} \cdot \mathbf{r} \right) \psi - \frac{2\theta}{\zeta} \nabla_{\mathbf{r}} \psi - \frac{2\mathbf{s}\psi}{\zeta} \right) \cdot \nabla_{\mathbf{r}} \mathcal{A}. \end{aligned} \quad (3.11)$$

Using the divergence theorem and the fact that ψ vanishes for $\mathbf{r} \rightarrow \infty$ we find that the first integral above vanishes, and one is left with

$$\frac{d\langle \mathcal{A} \rangle}{dt} = \int d\mathbf{r} \left(\left((\nabla_{\mathbf{x}} \mathbf{u})^{\text{T}} \cdot \mathbf{r} \right) \psi - \frac{2\theta}{\zeta} \nabla_{\mathbf{r}} \psi - \frac{2\mathbf{s}\psi}{\zeta} \right) \cdot \nabla_{\mathbf{r}} \mathcal{A}. \quad (3.12)$$

Integrating by parts the second term of this integral, and remembering that ψ vanishes at infinity, we get

$$\frac{d\langle \mathcal{A} \rangle}{dt} = \langle (\nabla_{\mathbf{x}} \mathbf{u})^{\text{T}} \cdot \mathbf{r} \cdot \nabla_{\mathbf{r}} \mathcal{A} \rangle + \frac{2\theta}{\zeta} \langle \nabla_{\mathbf{r}}^2 \mathcal{A} \rangle - \frac{2}{\zeta} \langle \mathbf{s} \cdot \nabla_{\mathbf{r}} \mathcal{A} \rangle. \quad (3.13)$$

As it will become clearer later, the quantity of interest is the average of the tensorial product $\langle \mathbf{r}\mathbf{r} \rangle$. It can be interpreted as the average size and orientation of the dumbbells. Substituting $\mathcal{A} = \mathbf{r}\mathbf{r}$ in the preceding equation one gets

$$\begin{aligned} \frac{d\langle \mathbf{r}\mathbf{r} \rangle}{dt} = & \langle (\nabla_{\mathbf{x}} \mathbf{u})^{\text{T}} \cdot \mathbf{r} \cdot \nabla_{\mathbf{r}} (\mathbf{r}\mathbf{r}) \rangle + \frac{2\theta}{\zeta} \langle \nabla_{\mathbf{r}}^2 (\mathbf{r}\mathbf{r}) \rangle - \frac{2}{\zeta} \langle \mathbf{s} \cdot \nabla_{\mathbf{r}} (\mathbf{r}\mathbf{r}) \rangle, \\ = & \langle \mathbf{r}\mathbf{r} \rangle \cdot \nabla \mathbf{u} + (\nabla \mathbf{u})^{\text{T}} \cdot \langle \mathbf{r}\mathbf{r} \rangle + \frac{4\theta}{\zeta} \mathbf{I} - \frac{4}{\zeta} \langle \mathbf{r}\mathbf{s} \rangle. \end{aligned} \quad (3.14)$$

In order to clarify the notations, we remind the reader that the velocity gradient $\nabla \mathbf{u}$ is given in components by $(\nabla \mathbf{u})_{\alpha\beta} = \partial_{\alpha} u_{\beta}$.

3.2.2 Action of the polymers on the solvent

The dumbbells act on the fluid momentum through the stress tensor \mathbf{P} . Following Lin [70], we will decompose it in two parts, the first, \mathbf{P}_s , corresponding to the Newtonian solvent and the second, \mathbf{P}_p , to the polymer

$$\begin{aligned}\mathbf{P} &= \mathbf{P}_s + \mathbf{P}_p, \\ &= p\mathbf{I} + \mathbf{\Lambda}_s + \mathbf{\Pi}, \\ &= p\mathbf{I} - 2\mu_s\mathbf{S} + \mathbf{\Pi}.\end{aligned}\tag{3.15}$$

where $p = p_s + p_p$, p_p being an isotropic pressure, $\mathbf{\Pi}$ is the contribution to the stress which vanishes at equilibrium (when the system is time independent, and the velocity gradients go to zero). The stress is decomposed in this fashion in order to obtain a non-zero contribution only when the system is off-equilibrium. The viscoelastic stress tensor $\mathbf{\Pi}$ is determined as follows. When considering a surface of fluid, there are two distinct possible contributions from the dumbbells on the stress acting on this surface. The first one is the action of the spring force when each bead of the dumbbell is on one side of the surface, \mathbf{P}_{sp} (see Fig. 3.3). The second one is the motion of the beads through the surface that carry some momentum, \mathbf{P}_{me} (see Fig. 3.4). The spring force

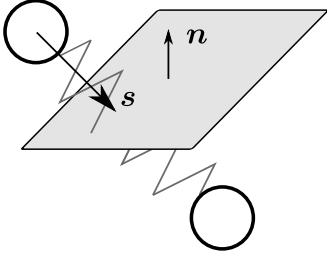


Fig. 3.3: The force \mathbf{s} applied on the surface of normal vector of length unity.

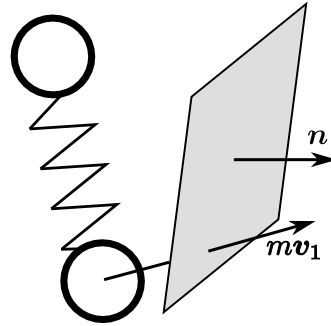


Fig. 3.4: The momentum $m\mathbf{v}_1$ carried by bead 1 through the surface with normal vector \mathbf{n} .

contribution, \mathbf{g}_{sp} , to the stress is computed as follows. Let us define a cubic volume which contains one dumbbell. If the density of dumbbells is defined as ρ_p then this volume is

$$V = \rho_p.\tag{3.16}$$

Let us also define a surface in this volume which will be identified by a normal vector of unit length \mathbf{n} . We now assume that the center of mass, \mathbf{r}_c , of a dumbbell is at a random position inside this volume. Then the probability

that a dumbbell of length \mathbf{r} will cross the plane \mathbf{n} is given by the ratio of the projection of \mathbf{r} on \mathbf{n} over the side length of the cube $V^{1/3}$. This probability must be multiplied by the probability distribution of having a dumbbell of size \mathbf{r} to get the specific statistical contribution due to dumbbells of size \mathbf{r} . Finally the force exerted by the spring on the surface defined by \mathbf{n} is obtained by multiplying by the spring force (see Fig. 3.3) and integrating over all the possible configurations of springs which leads to

$$\mathbf{g}_{\text{sp}} = -\rho_p \mathbf{n} \cdot \int d\mathbf{r} \mathbf{r} \mathbf{s} \psi. \quad (3.17)$$

Using the fact that the force in the \mathbf{n} direction is the stress projected on \mathbf{n} ($\mathbf{g}_{\text{sp}} = \mathbf{P}_{\text{sp}} \cdot \mathbf{n}$), the spring contribution to the stress tensor is

$$\mathbf{P}_{\text{sp}} = -\rho_p \langle \mathbf{r} \mathbf{s} \rangle. \quad (3.18)$$

We are now left with the evaluation of the contribution of the momentum carried by the beads of the dumbbells to the polymer stress. This contribution is equal to the momentum flux, in the comoving frame, of the beads through the surface defined by \mathbf{n} . Thus, defining $\mathbf{v}_i \equiv \dot{\mathbf{r}}_i - \mathbf{u}_i$, where $i = 1, 2$ is the index of each of the beads, and \mathbf{u}_i is the velocity of the fluid at the position of bead i , the contribution of each bead is

$$m \int d\mathbf{r}_1 d\mathbf{r}_2 \mathbf{v}_i \mathbf{v}_i \prod_{j=1}^2 f^{(0)}(\mathbf{v}_j), \quad (3.19)$$

where we assume that the dumbbells are at thermal equilibrium, and thus $f^{(0)}(\mathbf{v}_j)$ is the Maxwell-Boltzmann distribution function

$$f^{(0)}(\mathbf{v}_j) = \rho_p \left(\frac{m}{2\pi\theta} \right)^{D/2} \exp\left(-\frac{m\mathbf{v}_j^2}{2\theta}\right). \quad (3.20)$$

Carrying out this integral and summing over the two beads one gets, that the contribution of the dumbbells, through momentum exchange, to the stress tensor \mathbf{P}_{me} is given by

$$\mathbf{P}_{\text{me}} = 2\rho_p \theta \mathbf{I}. \quad (3.21)$$

Finally this gives us the total polymer contribution to the stress tensor

$$\mathbf{P}_p = \mathbf{P}_{\text{sp}} + \mathbf{P}_{\text{me}} = -\rho_p (\langle \mathbf{r} \mathbf{s} \rangle - 2\theta \mathbf{I}). \quad (3.22)$$

Using this relation in Eq. (3.15), the stress tensor is given by

$$\mathbf{P} = p_s \mathbf{I} - 2\mu_s \mathbf{S} - \rho_p (\langle \mathbf{r} \mathbf{s} \rangle - 2\theta \mathbf{I}). \quad (3.23)$$

When the system is time independent and the velocity gradients are vanishing, we have, using Eq. (3.14), that

$$\langle \mathbf{r}\mathbf{s} \rangle|_{\text{eq}} = \theta \mathbf{I}, \quad (3.24)$$

Therefore we get that the stress tensor is given by

$$p\mathbf{I} = (p_s + p_p)\mathbf{I} = \mathbf{P}|_{\text{eq}} = \mathbf{P}_s|_{\text{eq}} + \mathbf{P}_p|_{\text{eq}} = p_s\mathbf{I} + \rho_p\theta\mathbf{I} \Rightarrow p_p = \rho_p\theta. \quad (3.25)$$

Using this relation in Eq. (3.22), we can identify the viscoelastic stress tensor $\mathbf{\Pi}$ as

$$\mathbf{\Pi} = \mathbf{P}_p - p_p\mathbf{I} = -\rho_p(\langle \mathbf{r}\mathbf{s} \rangle - \theta\mathbf{I}). \quad (3.26)$$

We have made no hypothesis so far on the form of the force exerted by the spring on the beads. Assuming it to be Hookean

$$\mathbf{s} = H\mathbf{r}, \quad H = \frac{D\theta}{r_0^2} \quad (3.27)$$

where H is the Hookean constant², D is the physical dimension, and r_0 the length of the spring when the system is at equilibrium. We can rewrite $\mathbf{\Pi}$ as

$$\mathbf{\Pi} = -\frac{D\rho_p\theta}{r_0^2} \langle \mathbf{r}\mathbf{r} \rangle + \rho_p\theta\mathbf{I}. \quad (3.28)$$

We now would like to rewrite the constitutive equation in a more familiar way. To do so we define the polymer viscosity and relaxation time, respectively, μ_p and λ , as

$$\mu_p \equiv \rho_p\theta\lambda, \quad (3.29)$$

$$\lambda^{-1} \equiv \frac{4D\theta}{\zeta r_0^2}. \quad (3.30)$$

Inserting these relations in Eq. (3.28) we get

$$\mathbf{\Pi} = -\frac{\mu_p}{\lambda} (\mathbf{A} - \mathbf{I}), \quad (3.31)$$

where we defined, \mathbf{A} , the conformation tensor as

$$\mathbf{A} \equiv \frac{D}{r_0^2} \langle \mathbf{r}\mathbf{r} \rangle. \quad (3.32)$$

²At equilibrium, the energy contained in the spring must be equal to the thermal energy $\mathcal{E} = D\theta/2 = Hr_0^2/2$ and therefore $H = D\theta/r_0^2$

Expressing the constitutive equation (3.14) in terms of the conformation tensor, we get

$$\frac{d\mathbf{A}}{dt} \equiv \partial_t \mathbf{A} + (\mathbf{u} \cdot \nabla) \mathbf{A} = -\frac{1}{\lambda} (\mathbf{A} - \mathbf{I}) + \mathbf{A} \cdot \nabla \mathbf{u} + (\nabla \mathbf{u})^T \cdot \mathbf{A}. \quad (3.33)$$

This equation is also often written in terms of the viscoelastic stress tensor $\mathbf{\Pi}$, and reads

$$\frac{d\mathbf{\Pi}}{dt} = -\frac{1}{\lambda} \mathbf{\Pi} + \frac{2\mu_p}{\lambda} \mathbf{S} + \mathbf{\Pi} \cdot \nabla \mathbf{u} + (\nabla \mathbf{u})^T \cdot \mathbf{\Pi}. \quad (3.34)$$

This equation is known under the name of the Oldroyd-B constitutive equation and was derived by Oldroyd [77]. Although it exhibits the memory effects that can be observed in polymers, this model contains one defect, which is that the viscoelastic stress goes to infinity for a fixed elongation deformation. In order to correct this feature the FENE models (Finitely Extensible Nonlinear Elastic) have been proposed. Among them the most popular was devised by Peterlin [83] and is known under the name of FENE-P model. The FENE-P model imposes a finite maximum elongation, noted r_e , to the dumbbells. This additional constraint modifies Eqs. (3.33) and (3.34) in the following way

$$\frac{d\mathbf{A}}{dt} = -\frac{1}{\lambda} (h\mathbf{A} - b\mathbf{I}) + \mathbf{A} \cdot \nabla \mathbf{u} + (\nabla \mathbf{u})^T \cdot \mathbf{A}, \quad (3.35)$$

$$\frac{d\mathbf{\Pi}}{dt} = -\frac{\mu_p}{\lambda} \left(\frac{h}{\mu_p} \mathbf{\Pi} - 2b\mathbf{S} \right) + \mathbf{\Pi} \cdot \nabla \mathbf{u} + (\nabla \mathbf{u})^T \cdot \mathbf{\Pi}, \quad (3.36)$$

where the Peterlin function h , and the parameter b are given by

$$h = \frac{1}{1 - \text{tr} \mathbf{A} / r_e^2}, \quad b = \frac{1}{1 - 3/r_e^2}. \quad (3.37)$$

The relation between the viscoelastic stress tensor, $\mathbf{\Pi}$ and the conformation tensor \mathbf{A} becomes

$$\mathbf{\Pi} = -\frac{\mu_p}{\lambda} (h\mathbf{A} - b\mathbf{I}). \quad (3.38)$$

This dumbbell model although being able to reproduce some of the important viscoelastic properties is still not really quantitatively representing a real viscoelastic fluid (normal stresses effects). One of its major defects is that it only possesses one relaxation time. In reality, the behavior of polymeric fluids is governed by many different relaxation rates. This problem can be corrected by considering long chains of dumbbells connected by spring with different Hooke constants, as in the Rouse model (see Bird et al. [8]). An alternative is to consider a set of equations, each one with a specific relaxation time λ_i and viscoelastic stress $\mathbf{\Pi}_i$. The total viscoelastic stress being the sum of all the viscoelastic stresses related to each equations, $\mathbf{\Pi} = \sum_i \mathbf{\Pi}_i$, (see Bird et al. [9] for more informations).

3.2.3 Jeffreys model

The Jeffreys model is one of the very few “viscoelastic” models that has been implemented with the lattice Boltzmann method (see Giraud et al. [38], Lallemand et al. [62]). We will therefore shortly discuss it here and compare it with the dumbbell model described in the previous section. For more details one should refer to Bird et al. [8].

The Jeffreys model is a linear model (derived from the Maxwell model) that relates the viscoelastic stress tensor to the history of the strain rate tensor

$$\mathbf{\Pi} + \lambda_1 \partial_t \mathbf{\Pi} = -2\mu_s (\mathbf{S} + \lambda_2 \partial_t \mathbf{S}), \quad (3.39)$$

where λ_1 and λ_2 are the relaxation and the retardation time respectively. One may note that the upper-convected derivative³ that accounts for the memory effects is not present in this equation. This equation can be integrated, and assuming that $\mathbf{\Pi}$, \mathbf{S} and $\partial_t \mathbf{S}$ remain finite at $t = -\infty$ we have

$$\begin{aligned} \mathbf{\Pi} &= - \int_{-\infty}^t dt' \left(\frac{2\mu_s}{\lambda_1} \left(1 - \frac{\lambda_2}{\lambda_1} \right) e^{-(t-t')/\lambda_1} \mathbf{S} \right) - \frac{2\mu_s \lambda_2}{\lambda_1} \mathbf{S}, \\ &= - \int_{-\infty}^t dt' \left(\frac{2\mu_s}{\lambda_1} \left(1 - \frac{\lambda_2}{\lambda_1} \right) e^{-(t-t')/\lambda_1} + \frac{4\mu_s \lambda_2}{\lambda_1} \delta(t-t') \right) \mathbf{S}, \end{aligned} \quad (3.40)$$

where δ is the Dirac delta function. We see more clearly with this formulation that the viscoelastic stress is relaxing towards the strain rate. This model is valid only if the deformations applied on the fluid are small (because of its linearity). One of its major defects is its inability to describe the normal stresses difference phenomena, such as the climbing rod, or the tubeless syphon (see Bird et al. [8] for more details) effects.

3.3 Summary

In this chapter we have presented a viscoelastic model of a solution of polymers, that are described by dumbbells. This description couples the classical fluid conservation equations for mass and momentum with the conformation tensor constitutive equation that dictates time evolution of the polymer

³The upper convected derivative of a tensor \mathbf{A} , $\overset{\nabla}{\mathbf{A}}$, is defined as $\overset{\nabla}{\mathbf{A}} \equiv \partial_t \mathbf{A} + (\mathbf{u} \cdot \nabla) \mathbf{A} - \mathbf{A} \cdot \nabla \mathbf{u} - (\nabla \mathbf{u})^T \cdot \mathbf{A}$.

stress. Therefore the equations to be solved are in the Oldroyd-B case

$$\nabla \cdot \mathbf{u} = 0, \quad (3.41)$$

$$\partial_t \mathbf{u} + (\mathbf{u} \cdot \nabla) \mathbf{u} = \nabla \cdot (-p\mathbf{I} + 2\mu_s \mathbf{S} - \mathbf{\Pi}), \quad (3.42)$$

$$\partial_t \mathbf{A} + (\mathbf{u} \cdot \nabla) \mathbf{A} = -\frac{1}{\lambda}(\mathbf{A} - \mathbf{I}) + \mathbf{A} \cdot \nabla \mathbf{u} + (\nabla \mathbf{u})^T \cdot \mathbf{A}, \quad (3.43)$$

where the relation between the conformation tensor \mathbf{A} and the viscoelastic stress tensor is characterized by Eq. (3.31)

$$\mathbf{\Pi} = -\frac{\mu_p}{\lambda}(\mathbf{A} - \mathbf{I}).$$

In the case of the FENE-P model the constitutive equation is modified, and reads (see Eq. (3.35))

$$\frac{d\mathbf{A}}{dt} = -\frac{1}{\lambda}(h\mathbf{A} - b\mathbf{I}) + \mathbf{A} \cdot \nabla \mathbf{u} + (\nabla \mathbf{u})^T \cdot \mathbf{A},$$

and the relation between $\mathbf{\Pi}$ and \mathbf{A} is

$$\mathbf{\Pi} = -\frac{\mu_p}{\lambda}(h\mathbf{A} - b\mathbf{I}),$$

with $h = 1/(1 - \text{tr}\mathbf{A}/r_e^2)$ and $b = 1/(1 - 3/r_e^2)$. These two constitutive equations will be considered in the following chapters. Before proceeding further with the method used to solve them numerically, we will define the non-dimensional parameters that we will use later. The viscoelasticity of a fluid is usually given in terms of the Weissenberg or Deborah numbers defined in Chapter 1 as

$$\text{Wi} = \lambda \dot{\gamma}, \quad \text{De} = \lambda \frac{U}{L}, \quad (3.44)$$

where $\dot{\gamma}$ is the rate of deformation of the fluid, and U and L the characteristic velocity and length of the physical system of interest. The higher these numbers are, the higher the elastic effects. Finally we also define the kinematic viscosity ratio R_ν by

$$R_\nu = \frac{\nu_s}{\nu_p + \nu_s}, \quad (3.45)$$

where ν_p and ν_s are kinematic viscosities of, respectively, the polymer and the solvent.

Chapter 4

Velocity boundary conditions in the lattice Boltzmann method

In Chapter 2, the lattice Boltzmann scheme has been discussed, and we have shown how to recover the Navier–Stokes equations from the simplified BGK approximation. All this development did not take into account the boundary conditions (BC). In this chapter we will discuss the difficulties that arise when one tackles problems with boundaries while using the lattice Boltzmann method. More details may be found in Latt et al. [69].

4.1 Description of the problem

A time step in the LBM is constituted of two principal parts : the collision step and the propagation step. A cell next to a boundary will contain information that is incoming from the wall, and therefore will be non-physical. As can be seen on Fig. 4.1 the distributions number 3, 4, and 5 are pointing in the domain. They can therefore be interpreted as incoming from a node located inside the wall. Since the wall itself is not simulated these *missing* distributions must be recomputed in an appropriate way in order to guarantee that the physics of the whole system remains correct. Fortunately,

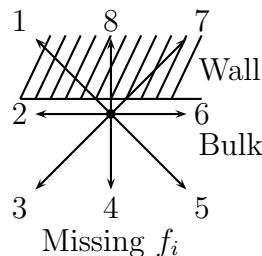


Fig. 4.1: A boundary node with the known (0, 1, 2, 6, 7, 8) and the missing (3, 4, 5) population, f_i .

we do not only know the distribution functions that are incoming from the bulk but we also have information about macroscopic observables. Usually, in fluid flows, one knows the velocity (Dirichlet boundary conditions) or its derivatives (Neumann boundary conditions). As we will see later this information is enough to completely specify the system. We will discuss here only the case of the homogeneous Dirichlet BC also known as the no-slip BC when the boundary is a solid wall.

4.2 Proposed solutions

There has been a large number of boundary conditions that were developed through the years. In this section we will only discuss the flat wall BC in order to keep the discussion simple. But the reader should be aware that BC for curved boundaries have also been developed (see Bouzidi et al. [14], Ginzburg and d’Humières [36], Junk and Yang [57] for example).

Let us now study in more details four different BCs chosen among the most popular ones :

- the Inamuro method (*BC1*),
- the Zou/He method (*BC2*),
- the regularized method (*BC3*).
- the finite difference method (*BC4*),

Of course this list does not intend to be exhaustive. Numerous other BC have been developed (among others the articles by Ansumali and Karlin [4] or Halliday et al. [46]). *BC1* was developed in Inamuro et al. [54]. The *BC2* is a result of the work of Zou and He [104]. Finally *BC3* is described in Latt and Chopard [68]. Method *BC4* is based on the work by Skordos [93] with a slight modification.

4.2.1 General properties of the studied BC

The precited BC have several common points that will be described now. First of all they are all “wet” BC, which means that the lattice nodes on which the boundary treatment will be applied are located in the fluid infinitesimally close to the wall¹. This has as a consequence that the BGK collision must be applied on the boundary cell after the recovery of the missing information.

¹In opposition to the “dry” BC like the bounceback where the boundary node lies into the wall

Since the collision depends on the equilibrium distribution, and therefore on the density and the velocity, we have the first constraint that in a boundary node the density and the velocity must be exactly recovered according to Eqs. (2.104) and (2.105). Since we assume Dirichlet boundary conditions, the velocity is imposed and the density must be computed. Let us now describe the procedure to determine it.

4.2.2 Computation of the density

The procedure describe below can also be found in Zou and He [104]. If \mathbf{n} is the unit vector normal to the wall pointing outwards of the computational domain, we define three quantities, ρ_0 , ρ_- and ρ_+ as

$$\rho_0 = \sum_{i \in I_0} f_i, \quad \text{where } I_0 = \{i \mid \boldsymbol{\xi}_i \cdot \mathbf{n} = 0\} \quad (4.1)$$

$$\rho_- = \sum_{i \in I_-} f_i, \quad \text{where } I_- = \{i \mid \boldsymbol{\xi}_i \cdot \mathbf{n} < 0\} \quad (4.2)$$

$$\rho_+ = \sum_{i \in I_+} f_i, \quad \text{where } I_+ = \{i \mid \boldsymbol{\xi}_i \cdot \mathbf{n} > 0\} \quad (4.3)$$

The density is therefore given by

$$\rho = \rho_0 + \rho_+ + \rho_-. \quad (4.4)$$

Then defining u_\perp as the projection of the velocity on \mathbf{n}

$$u_\perp = \mathbf{u} \cdot \mathbf{n}. \quad (4.5)$$

and the preceding definitions one has for u_\perp (see also Eq. (2.105))

$$u_\perp = (\rho_+ - \rho_-)/\rho. \quad (4.6)$$

Combining this result with Eq. (4.4) we find

$$\rho = \frac{1}{1 + u_\perp} (2\rho_+ + \rho_0). \quad (4.7)$$

Since ρ_+ and ρ_0 are always known on a boundary cell, it can be seen that imposing u_\perp , one can compute ρ or vice versa. The relation between u_\perp and ρ allows us to impose either velocity or pressure BC. Of course these results are only true for flat walls. In the case of corners, for example, the missing information is too important to be recovered from the few available f_i 's. Therefore one needs to interpolate the missing quantities from the bulk, using finite difference schemes for example.

4.2.3 Replacing only the missing populations or all of them

Having determined the density and the velocity we are able to compute the equilibrium distribution function. Only the unknown particle distribution functions are lacking. There are two approaches to solve this problem. The first one consists in keeping the known distributions and therefore in using the information incoming from the bulk, by building only the missing ones. This is the path followed by *BC1* and *BC2*. The other approach is to replace all the distribution functions using only the macroscopic information available, such as the density, the velocity and higher order moments. This is the approach used for *BC3* and *BC4*. A naive approach when using the technique of replacing all distribution functions could be to give all the f_i 's their equilibrium value. Although the density and the velocity will be correctly recovered, the result is of rather low quality, since the approximation of the f_i will be of only $\mathcal{O}(\text{Kn}^0)$ whereas the approximation made for the distribution functions in the Chapman–Enskog expansion is of $\mathcal{O}(\text{Kn}^1)$. One example of this BC can be found in Fig. 4.2. One immediately sees that the hydrodynamics are only qualitatively recovered. In fact in order to get the expected accuracy, we do not only need to get the correct density and velocity but also the strain rate tensor.

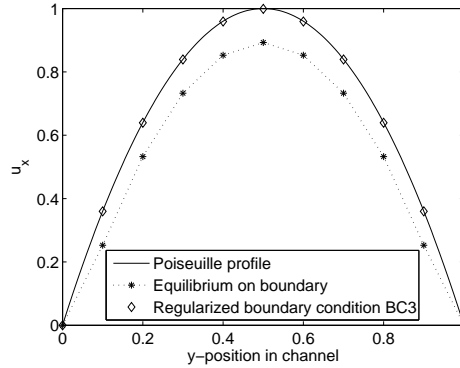


Fig. 4.2: 2D channel flow, simulated once with an equilibrium distribution on boundaries, and once with the Regularized boundary condition *BC3*. The solid line plots the analytic Poiseuille profile. Although only 11 data points are used to resolve the channel width, the result obtained with *BC3* is visually indistinguishable from the Poiseuille profile. With equilibrium boundaries on the other hand, the velocity gradients are ill represented on boundary nodes, which impairs the accuracy of the whole simulation.

Method	Replaces all f_i 's	Local	Explicit in 3D
$BC1$ (Inamuro)	No	Yes	No
$BC2$ (Zou/He)	No	Yes	Yes
$BC3$ (Regularized)	Yes	Yes	Yes
$BC4$ (FD)	Yes	No	Yes

Table 4.1: The boundary conditions analyzed

4.3 Description of the boundary conditions

The four BC listed in Table 4.1 are analyzed in this chapter. The table specifies whether a BC replaces all the populations by a new value, or only the unknown ones. It also states whether they are local to a given boundary node, or if they need to access neighboring nodes in order to execute their algorithm. The original papers on boundary conditions most often concentrate on 2D implementations. Table 4.1 specifies whether there exists a straightforward way to extend them to 3D, or if this requires the solution of an implicit equation. In the remaining part of this section, the four analyzed BC are introduced within a common framework. A formalism is used in which all particle populations f_i on a boundary node are replaced by alternative populations h_i resulting from the boundary condition algorithm. It is stated explicitly when some of the original f_i 's are kept. It is then assumed that a BGK collision is applied to the populations h_i , followed by a streaming step. Explicit formulas are given for the D2Q9 lattice of Fig. 4.3 and for a top grid node as in Fig. 4.1. Furthermore, it is explained how the algorithms can be extended to 3D lattices.

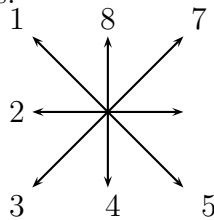


Fig. 4.3: The D2Q9 lattice.

All presented BC are shown to recover the desired hydrodynamic behavior. According to the discussion in Chapter 2, this amounts to showing that the expected value for the boundary velocity is obtained from Eq. (2.105), and that the deviatoric stress tensor $\mathbf{P}^{(1)}$ calculated from Eq. (2.71) is consistent with its hydrodynamic value predicted by Eq. (2.90). These requirements

are enforced by construction in the $BC3$, and $BC4$, and are therefore valid for obvious reasons. However, for the BC $BC1$ and $BC2$ only the value of the velocity is explicitly enforced. In this thesis, the value of the deviatoric stress tensor is therefore evaluated separately for the D2Q9 lattice, as a proof of validity. For this, the six known particle populations are assumed to obey the value predicted by the perturbative analysis in Eq. (2.88), and thus to depend on ρ , \mathbf{u} and the gradients of \mathbf{u} . Based on this assumption, the missing particle populations are evaluated from the algorithm of the boundary condition, and the deviatoric stress tensor $\mathbf{P}^{(1)}$ is computed from Eq. (2.71).

4.3.1 The Inamuro method ($BC1$)

The boundary condition $BC1$ by Inamuro et al. originates from the kinetic theory of gases to find an appropriate value for the missing particle populations. It assumes that the missing populations are at a local thermodynamic equilibrium and thus yield the discrete analog of a Maxwell-Boltzmann distribution. This distribution is however centered around a fictitious density ρ' and velocity \mathbf{u}' , both of which are different from the macroscopic fluid variables ρ and \mathbf{u} . While ρ' is a free variable, \mathbf{u}' is assumed to differ from \mathbf{u} only by a “slip velocity” $\delta\mathbf{u}$, which is tangential to the wall: $\mathbf{u}' = \mathbf{u} + \delta\mathbf{u}$. These assumptions introduce a set of 2 unknowns in 2D, and 3 unknowns in 3D. For an upper boundary on a D2Q9 lattice, they read

$$h_i = f_i^{(0)}(\rho', u_x + \delta u_x, u_y) \quad \text{for } i = 3, 4, 5, \quad (4.8)$$

where $f_i^{(0)}$ is the equilibrium distribution described by Eq. (2.83). As has been emphasized in Section 4.2.1, the value of the velocity \mathbf{u} needs to be enforced on the boundary *via* Eq. (2.105). This system of two equations can be solved for the two unknowns ρ' and δu_x , with the following result:

$$\rho' = 6 \frac{-\rho u_y + f_1 + f_7 + f_8}{1 + 3u_y^2 - 3u_y}, \quad (4.9)$$

$$\delta u_x = -u_x + 6 \frac{\rho u_x + f_1 + f_2 - f_6 - f_7}{\rho'(1 - 3u_y)}. \quad (4.10)$$

In these relations, the lattice constants have been replaced by their numerical value for the D2Q9 lattice. The missing populations are now evaluated with the help of Eq. (4.8), and the other populations are left unchanged:

$$h_i = f_i \quad \text{for } i = 0, 1, 2, 6, 7, 8. \quad (4.11)$$

In 3D, the slip velocity $\delta\mathbf{u}$ has two independent components. The unknowns thus add up to 3, which matches the number of closure relations introduced

by Eq. (2.105). This system of equations is more difficult to solve than in the 2D case, and to our knowledge, no analytic solution is presently known. For the 3D benchmarks in Section 4.4, the lattice Boltzmann code solves these equations numerically on each boundary cell and at every time step with the help of a multivariate Newton–Raphson method [85].

The velocity \mathbf{u} is explicitly enforced by this boundary condition, but nothing is known *a priori* about the value of the deviatoric stress tensor. The method described in the introduction of Section 4.3 is now applied to evaluate $\mathbf{P}^{(1)}$ for the D2Q9 lattice, by making use of a computer algebra system. The following values are found:

$$P_{xx}^{(1)} = \frac{\tau\rho}{9(1-3u_y)^2\left(\frac{1}{\tau} - 2S_{yy} - \frac{3}{\tau}u_y + \frac{3}{\tau}u_y^2\right)} \cdot \left(10S_{xx}S_{yy}(1-3u_y)^2 + 6S_{xy}^2(1-3u_y+3u_y^2) - \frac{1}{\tau}(3u_y-1)(5S_{xx}(-1+6u_y-12u_y^2+9u_y^3) + S_{yy}u_x^2(3-9u_y) + 18u_xS_{xy}\left(\frac{1}{3}-u_y+u_y^2\right))\right), \quad (4.12)$$

$$P_{yy}^{(1)} = -\frac{2\tau}{3}\rho S_{yy}, \quad \text{and} \quad (4.13)$$

$$P_{xy}^{(1)} = -\frac{2\tau}{3}\rho S_{xy}. \quad (4.14)$$

The tensor components $P_{yy}^{(1)}$ and $P_{xy}^{(1)}$ yield an exact match with the value expected from Eq. (2.90). As has been pointed out in Section 4.2.1, this is a direct corollary of the fact that the equation for the velocity, Eq. (2.105), is satisfied. The component P_{xx} is the only free parameter on a D2Q9 lattice, and can be viewed as a signature of how the boundary condition influences the dynamics of the fluid. In its present form, Eq. (4.12) is not very enlightening, and we present therefore two limit values. The first represents the case of a no-slip wall, with vanishing values of \mathbf{u} :

$$P_{xx}^{(1)} = \rho \frac{-5S_{xx} + 2\tau(3S_{xy}^2 + 5S_{xx}S_{yy})}{\frac{9}{\tau} - 18S_{yy}} \quad \text{when } \mathbf{u} = 0. \quad (4.15)$$

For the second limit case, $P_{xx}^{(1)}$ is expanded into a finite Taylor series for small values of the components S_{xx} , S_{yy} and S_{xy} . It is assumed that, in the limit of fluid incompressibility, the Mach number Ma , and thus the velocity \mathbf{u} in lattice units, scales like the Knudsen number (see the book of Hänel [47]). The components of the strain rate tensor \mathbf{S} exhibit therefore order

Kn^2 scaling. From the above Taylor series, only terms of order Kn^2 are retained, and terms of the form $S_{\alpha\beta}S_{\gamma\delta}$ or $S_{\alpha\beta}u_\gamma$ are neglected. This leads to the following approximation of Eq. (4.12):

$$P_{xx}^{(1)} = -\frac{5\tau}{9}\rho S_{xy} + \mathcal{O}(\text{Kn}^3). \quad (4.16)$$

This expression is identical to the hydrodynamic tensor value in Eq. (2.90), except for the factor $5/9$, which differs from the expected value $2/3$ by 20%. This error has however no important effect in practice, as the Inamuro method is found to be equally or even more accurate than the other approaches in low-Reynolds applications (see Section 4.4). One may however point out that as a consequence of Eq. (4.16), the trace of $\mathbf{P}^{(1)}$ is non-zero with *BC1*. This is a huge defect since it is a violation of the continuity equation, because Eqs. (2.90) and (2.86) show that the trace of $\mathbf{P}^{(1)}$ is proportional to $\nabla \cdot \mathbf{u}$ in the hydrodynamic limit (a detailed discussion about mass conservation of boundary conditions can be found in Chopard and Dupuis [19]).

4.3.2 The Zou/He method (*BC2*)

The boundary condition *BC2* by Zou and He is based on an idea referred to by its authors as “applying the bounce-back rule to off-equilibrium parts”. This expression can be understood as a reference to a symmetry property of the variables $f_i^{(1)}$ that follows from Eq. (2.88). Let the opposite $\text{opp}(i)$ of a lattice direction i be defined by the relation $\boldsymbol{\xi}_{\text{opp}(i)} = -\boldsymbol{\xi}_i$. Then, Eq. (2.88) is symmetric under the operation of taking the opposite direction :

$$f_i^{(1)} = f_{\text{opp}(i)}^{(1)}. \quad (4.17)$$

This relation can be used to copy data from known particle populations to the opposite, unknown ones. It may however not be applied blindly to all unknown particle populations, because this would lead to a conflict with Eq. (2.105) for the velocity. On a D2Q9 lattice for example, there is really only one degree of freedom left on which the boundary condition may have an influence. The choice made in the case of *BC2* was to apply Eq. (4.17) to the unknown particle population whose lattice vector is parallel to the boundary normal. Thus, for an upper boundary on a D2Q9 lattice, the following relation is applied:

$$h_4^{(1)} = f_8^{(1)}. \quad (4.18)$$

The remaining two unknown populations $h_3^{(1)}$ and $h_5^{(1)}$ are computed by enforcing the values of the velocity through Eq. (2.105). On a D2Q9 lattice, the obtained values are

$$h_3^{(1)} = f_7^{(1)} + \frac{1}{2}(f_6^{(1)} - f_2^{(1)}) \quad \text{and} \quad (4.19)$$

$$h_5^{(1)} = f_1^{(1)} + \frac{1}{2}(f_2^{(1)} - f_6^{(1)}). \quad (4.20)$$

The value of the unknown particle populations is constructed by adding equilibrium and off-equilibrium parts:

$$h_i = f_i^{(0)}(\rho, \mathbf{u}) + h_i^{(1)} \quad \text{for } i = 3, 4, 5. \quad (4.21)$$

All other populations are kept as they are:

$$h_i = f_i \quad \text{for } i = 0, 1, 2, 6, 7, 8. \quad (4.22)$$

Like the Inamuro method *BC1*, the Zou/He method *BC2* enforces the value of the velocity explicitly, but does not specify anything about the deviatoric stress tensor. The values of $\mathbf{P}^{(1)}$ can again be computed for the D2Q9 lattice, using the procedure described in the introduction of Section 4.3. The resulting expressions match exactly the hydrodynamic values of $\mathbf{P}^{(1)}$ claimed in Eq. (2.90). This confirms analytically the validity of the Zou/He boundary condition for 2D simulations.

This algorithm cannot be used as it is in 3D. In that case, Eq. (2.105) yields three independent equations, but the number of unknowns is larger (5 or 9). In that case, Zou and He [104] suggest to proceed as follows. To begin with, the “off-equilibrium bounce-back” rule, Eq. (4.17), is applied to assign a value to all missing particle populations. By doing this, the exact value of the component u_α is recovered for the index α in direction of the boundary normal. To keep this relation valid, the sum over the value of the unknown particle populations is kept invariant during subsequent operations. The excess of momentum is then evaluated in the remaining directions:

$$\delta_\beta \equiv \sum_i f_i^{\text{neq}} \xi_{i\beta} \quad \text{for } \beta \neq \alpha. \quad (4.23)$$

Finally, the values δ_β are redistributed over the unknown particle populations, in order to find an exact match for Eq. (2.105):

$$h_i^{\text{neq}} = f_i^{\text{neq}} - \sum_{\beta \neq \alpha} \frac{1}{n_\beta} \xi_{i\beta} \delta_\beta \quad \text{for all unknown } f_i \text{'s}, \quad (4.24)$$

where n_β is the number of unknown particle populations for which $\xi_{i\beta}$ is non-zero.

4.3.3 The Regularized method (*BC3*)

The method by Latt and Chopard (see Latt and Chopard [68]) replaces all particle populations on a boundary node. It first evaluates the value of $\mathbf{P}^{(1)}$, based on the knowledge of the known particle populations. Then, the equations of the perturbative analysis in Chapter 2 are used to reconstruct all particle populations from ρ , \mathbf{u} and $\mathbf{P}^{(1)}$ in a way consistent with the hydrodynamic limit of the model. With the use of Eq. (2.90), and by exploiting the symmetry of tensors $\mathcal{H}_{i\alpha\beta}^{(2)}$ and of tensor $\mathbf{P}^{(1)}$, Eq. (2.88) is rewritten as follows:

$$f_i^{(1)} \approx -\frac{\rho w_i \tau}{c_l^2} \mathcal{H}_{i\alpha\beta}^{(2)} S_{\alpha\beta} = \frac{w_i}{2c_l^4} \mathcal{H}_{i\alpha\beta}^{(2)} P_{\alpha\beta}^{(1)}. \quad (4.25)$$

With this relation, the off-equilibrium part of the particle populations can be reconstructed from the tensor $\mathbf{P}^{(1)}$. This procedure is called a regularization of the particle populations, because it compels the $f_i^{(1)}$ to respect exactly (without numerical error) the symmetry properties of Eq. (2.90), such as the symmetry explicit in Eq. (4.17).

The tensor $\mathbf{P}^{(1)}$ is evaluated as follows. First, all unknown particle populations are assumed to take the value obtained by a “bounce-back of off-equilibrium parts”, as in Eq. (4.17). Thus, their value is described as $f_i = f_i^{(0)}(\rho, \mathbf{u}) + f_{\text{opp}(i)} - f_{\text{opp}(i)}^{(0)}$. It was emphasized in Section 4.3.2 that this relation may not be used to define a boundary condition, because it prohibits an exact implementation of Eq. (2.105). It is therefore used temporarily only, to evaluate the value of $\mathbf{P}^{(1)}$ by means of Eq. (2.71).

After this, Eq. (4.25) is used to construct all particle populations:

$$h_i = f_i^{(0)}(\rho, \mathbf{u}) + \frac{w_i}{2c_l^4} \mathcal{H}_{i\alpha\beta}^{(2)} P_{\alpha\beta}^{(1)} \quad \text{for } i = 0, \dots, q-1. \quad (4.26)$$

It is obvious that not only ρ and \mathbf{u} , but also the tensor $\mathbf{P}^{(1)}$ are recovered appropriately by this boundary condition.

4.3.4 The Finite-difference velocity gradient method (*BC4*)

In the regularized approach of *BC3*, the deviatoric stress tensor $\mathbf{P}^{(1)}$ is evaluated on the boundary from information locally available on the cell. The boundary condition *BC4* computes $\mathbf{P}^{(1)}$ from Eq. (2.90) instead, by relating it to the strain rate tensor \mathbf{S} . The components of \mathbf{S} are evaluated by a second-order accurate finite-difference scheme, which accesses the value of

the velocity on neighboring grid cells. The algorithm of *BC4* is summarized by the following equation:

$$h_i = f_i^{(0)}(\rho, \mathbf{u}) - \frac{\rho\tau w_i}{c_l^2} \mathcal{H}_{i\alpha\beta}^{(2)} S_{\alpha\beta} \quad \text{for } i = 0, \dots, q-1, \quad (4.27)$$

where the strain rate tensor \mathbf{S} is defined as in Eq. (2.86). The velocity gradients of \mathbf{S} that run along directions parallel to the boundary are evaluated by a symmetric finite difference scheme. This presumes the knowledge of the velocity on nearest-neighbor cells. Velocity gradients along the boundary normal use non-symmetric finite differences, which access the velocity on nearest and next-to-nearest neighbors. Please note that, instead of evaluating the symmetric tensor \mathbf{S} , one may alternatively compute all components of the non-symmetric tensor $\nabla\mathbf{u}$ and calculate the population functions from Eq. (2.88). These two ways of constructing the boundary condition are algebraically equivalent.

This boundary condition was published in 1993 in a pioneering article by Skordos [93]. It should be mentioned that Skordos makes a slightly different assumption on the asymptotic value of $\mathbf{P}^{(1)}$, as it uses the following expression instead of Eq. (2.90), in which the density ρ resides inside the spatial derivative:

$$\mathbf{P}^{(1)} = -c_l^2\tau \left(\nabla(\rho\mathbf{u}) + (\nabla(\rho\mathbf{u}))^T \right). \quad (4.28)$$

It has however been shown that Eq. (2.90), which predicts the proper deviatoric strain rate for a compressible fluid, can be recovered in a perturbative analysis by taking into account derivatives of non-linear velocity terms (cf. Dellar [25], Guo et al. [45], Latt [66]). The difference between Eqs. (2.90) and (4.28), however subtle, may have a noticeable effect in compressible fluids. We decided therefore to depart from the original article on this point, but emphasize that the credit for this boundary condition should be attributed to Skordos [93]. It should also be mentioned that Skordos discusses both first- and second-order accurate finite difference approximations to velocity gradients, but only second-order schemes are presented here and for a wall like described in Fig. 4.1 are given by

$$\partial_x \mathbf{u}(x, y) = \frac{\mathbf{u}(x+1, y) - \mathbf{u}(x-1, y)}{2}, \quad (4.29)$$

$$\partial_y \mathbf{u}(x, y) = \frac{-3\mathbf{u}(x, y) + 4\mathbf{u}(x, y-1) - \mathbf{u}(x, y-2)}{2}. \quad (4.30)$$

An interesting discussion of BC with finite-difference approximation to velocity gradients is presented in Halliday et al. [46]. This paper points out

that other choices than Eq. (4.27) also lead to the expected value of the deviatoric stress tensor, and it proposes different closure schemes that are compatible with the hydrodynamic limit of the model.

4.4 Benchmarks

This section produces the results of lattice Boltzmann simulations, based on the four reviewed boundary conditions. For some of the simulated problems, analytical solutions are known which describe the steady state of the flow. In that case, the simulation is iterated until a steady state is reached. Then, the numerical result is validated against the analytical solution on every grid point, and the mean discrepancy between these two values is evaluated as a measure of quality for the implemented boundary condition. Other benchmarks focus on the time evolution of the flow. They keep track of the time-evolution of a scalar quantity (such as the average energy in Section 4.4.3), or the occurrence of a particular event (as the maximum value of the enstrophy in Section 4.4.4). Several 2D benchmarks are presented, implemented on a D2Q9 lattice, and one 3D benchmark on a D3Q19 lattice.

The simulated domains of all presented problems have a rectangular shape. This raises the issue of how to implement corner nodes and, in 3D, edge nodes that lie on the connection between two plane walls. Although some authors of boundary conditions make suggestions on how to treat these cases, some authors do not. To guarantee an equal treatment, the present work treats corners and edges always in the same way, independently of the boundary condition that is being tested. The finite-difference algorithm of BC_4 was selected to implement these special boundary nodes. This approach is straightforward, because the velocity gradients referred to in Eq. (4.26) can be evaluated on corners and edges just like everywhere else. It must furthermore be pointed out that Eq. (4.7) for the particle density cannot be evaluated on corners and edges. In those cases, the density is extrapolated with second-order accuracy from neighboring cells.

In all benchmark problems, a velocity U and a length L are selected so that they are characteristic of the flow. The dynamics of the flow is then described in a system of dimensionless variables, independent of the numerical grid, in which $U = 1$ and $L = 1$. The equations of motion depend only on the dimensionless Reynolds number, defined as a function of the kinematic fluid

viscosity ν :

$$\text{Re} = \frac{UL}{\nu}. \quad (4.31)$$

The parameters used for the numerical implementation are described by the number N , which is the number of grid nodes used to resolve the length L , and the velocity U_{lb} , representing the velocity U in a system of lattice units. This is a common choice in LB simulations, because U_{lb} is proportional to the Mach number of the fluid. It can therefore be fine-tuned to make sure the flow is close enough to the limit of incompressibility. For the numerical implementation of the benchmarks, the prescribed velocity field for the initial and/or boundary condition needs to be converted from dimensionless variables to lattice units. This is simply done by multiplying the dimensionless velocity by U_{lb} . From these definitions, the discrete parameters of the simulation are characterized as follows. The grid interval Δx is given by

$$\Delta x = \frac{1}{N-1}. \quad (4.32)$$

The time lapse of an iteration Δt is defined through the relation $U = \Delta x / \Delta t U_{\text{lb}}$, which recovers the lattice-independent form of the velocity. Given our choice of $U = 1$, this leads to

$$\Delta t = \Delta x U_{\text{lb}}. \quad (4.33)$$

At a given time step, the numerical error is evaluated by computing a L^2 -norm of the difference between the simulated velocities $\mathbf{u}_{\text{lb}}(\mathbf{x}_k)$ on grid nodes located at position \mathbf{x}_k , and the dimensionless analytical solution $\mathbf{u}_{\text{analytic}}(\mathbf{x}_k)$:

$$E = \sqrt{\frac{1}{M} \sum_{k=0}^{M-1} \left\| \frac{\mathbf{u}_{\text{lb}}(\mathbf{x}_k)}{U_{\text{lb}}} - \mathbf{u}_{\text{analytic}}(\mathbf{x}_k) \right\|^2}, \quad (4.34)$$

where the sum runs over all M nodes of the numerical grid.

The parameters of the simulation for a varying resolution N are adapted in order to keep the value of the Reynolds number constant, and to prevent effects of fluid compressibility from interfering with the accuracy of the result. Compressibility errors E_{compr} are known to scale like the square Mach-number and are therefore estimated by $E_{\text{compr}} = \mathcal{O}(U_{\text{lb}}^2)$. This error is required to decrease at least as fast as the discretization error $E_{\Delta x} = \mathcal{O}(1/L_{\text{lb}}^2) = \mathcal{O}(1/N^2)$. It follows from Eq. (4.31) that this is achieved by keeping the viscosity ν_{lb} constant when the grid resolution is modified.

The benchmark codes can be retrieved from the Internet address [13]. The programs are based on the open source, C++, parallel, lattice Boltzmann library OpenLB [86], which is publicly available.

4.4.1 Steady plane channel flow (2D)

This 2D stationary flow evolves in a straight channel, which extends in x -direction between $x = 0$ and $x = l_x$. The walls of the channel are parallel to the x -axis, and defined by the equations $y = 0$ for the lower wall, and $y = l_y$ for the upper wall. A no-slip condition for the velocity is enforced on these walls. The flow is characterized by a constant pressure drop along the channel. This pressure drop can be obtained by using pressure boundary conditions on the inlet and the outlet. An alternative approach, which is used in this benchmark, is to enforce the velocity profile from the analytical solution of the flow on the inlet and on the outlet. The velocity is parallel to the walls, and the only non-zero component u_x is independent of x : $u_x = u_x(y)$. The height l_y of the channel is taken to represent the characteristic length L . The maximum value of the velocity, which is measured in the middle of the channel, is selected for the characteristic velocity U . The analytic solution to this flow is given by the parabolic Poiseuille profile, which, in dimensionless variables, reads

$$u_x(y) = 4(y - y^2). \quad (4.35)$$

The pressure drop amounts to $\partial_x p = -8/\text{Re}$.

The two-dimensional channel flow is for many reasons an inappropriate benchmark, and it is presented here in a historical spirit, to conform with some authors of boundary conditions that use it as a basic test case for their algorithm. One shortcoming of this flow as a benchmark case is that the components S_{xx} and S_{yy} of the strain rate tensor \mathbf{S} vanish in the analytical solution. Boundary conditions that replace only unknown particle populations (*BC1* and *BC2* in this work) are therefore automatically exempt from numerical errors. This follows from the discussion in Section 4.2.1, where the components $P_{xx}^{(1)}$ (on horizontal boundaries) and $P_{yy}^{(1)}$ (on vertical boundaries) of the stress tensor are shown to be the only degrees of freedom for which the boundary condition could possibly be wrong. If the corner nodes are implemented in an appropriate way, these two methods are shown in Refs. Inamuro et al. [54], Zou and He [104] to solve the 2D channel flow with a precision close to machine accuracy (the best accuracy one can expect to obtain, given the limited precision of floating point arithmetic on a computer), independent of the grid resolution. Machine accuracy is however not exhibited in the present benchmark, because we chose to implement corner nodes in a generic way that works for all boundary conditions and for all types of flows. In this benchmark, the channel has square shape ($l_x = l_y = 1$), the grid resolution is varied from $N = 25$ to $N = 400$, and the Reynolds number is $\text{Re} = 10$. Compressibility effects are controlled by setting $U_{\text{lb}} = 0.01$ at

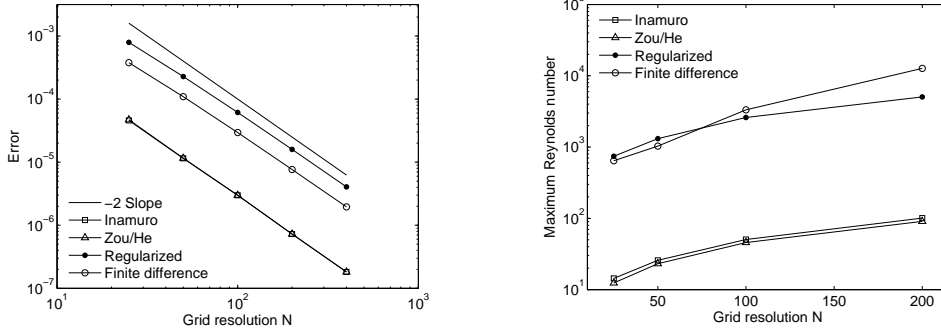


Fig. 4.4: (a) Numerical accuracy in a 2D channel flow. (b) Numerical stability in a 2D channel flow. The maximum Reynolds number which can be reached before numerical instabilities appear is plotted.

a grid resolution $N = 50$. As discussed in the previous paragraph, the velocity U_{lb} is recomputed for different values of N in such a way as to keep the viscosity ν_{lb} constant from one grid to another. Fig. 4.4 (a) shows that all reviewed boundary conditions lead to the expected second-order accuracy with respect to grid resolution. Boundary conditions $BC1$ and $BC2$ are distinctly more accurate, because their overall accuracy is only affected by the error in corner nodes. The numerical stability of boundary conditions is explored in Fig. 4.4 (b) by means of the maximum Reynolds number which can be reached before numerical instabilities occur. The most accurate boundary conditions, $BC1$ and $BC2$ are also the distinctly least stable. The most stable boundary condition is based on a non-local algorithm, i.e. $BC4$.

4.4.2 Oscillating plane channel flow (2D)

This laminar channel flow is also known as “Womersley flow” on ground of the analytical solution proposed by Womersley [101]. It is defined by the same geometry as the steady flow in Section 4.4.1. The pressure gradient is however not constant, but oscillates in time. Before the reference velocity U can be defined, the equations of the flow are written in an arbitrary system of units, other than the dimensionless one. In this system, the amplitude of the oscillations is denoted by A , and the frequency by ω :

$$\partial_x p = -A \cos(\omega t). \quad (4.36)$$

In the low-frequency limit $\omega \rightarrow 0$, the solution to this flow is defined by a succession of Poiseuille profiles, with oscillating amplitude:

$$u_x(y, t) = \frac{A}{2\nu}(l_y y - y^2) \cos(\omega t). \quad (4.37)$$

The maximum velocity of the flow, reached in the middle of the channel at time $t = n2\pi/\omega$ for any integer value n , is found to be

$$u_{\max} = \frac{A}{8\nu}L^2. \quad (4.38)$$

This value u_{\max} is used to define the reference velocity U , as it characterizes the flow reasonably well when oscillations are slow. The reference length L is described by the channel height l_y . As the flow depends on time, it is characterized by two dimensionless parameters, which are the Reynolds number Re defined in Eq. (4.31), and the Womersley number α , defined as

$$\alpha = \frac{L}{2} \sqrt{\frac{\omega}{\nu}}. \quad (4.39)$$

In a system of dimensionless variables described by U and L , the time-dependent solution to this flow is given by the Womersley profile (see the works of Womersley [101] or Cosgrove et al. [23] :

$$u_x(y, t) = \Re \left(e^{i \frac{\alpha^2}{\text{Re}} t} \frac{8}{i\alpha^2} \left(1 - \frac{\cosh(\sqrt{2}(\alpha + i\alpha)(y - \frac{1}{2}))}{\cosh(\frac{\sqrt{2}}{2}(\alpha + i\alpha))} \right) \right), \quad (4.40)$$

where i is the imaginary unit, and \Re means that the real part of the formula needs to be evaluated.

The error E of the numerical result with respect to the analytical solution of Eq. (4.40) is computed at each time step by evaluation of Eq. (4.34) over the whole computational domain. The overall error \overline{E} of the simulation is defined as an average of E over one time period. Only the asymptotic value of \overline{E} , reached after a large number of iterations, is accounted for. One should not pay attention to the initial transient regime of the fluid, and consequently the choice of the initial condition.

Although this benchmark is time dependent we will not discuss the time accuracy of the model. As discussed in the beginning of this section, we have to decrease the time-stepping Δt like Δx^2 in order to get rid of the compressibility errors and therefore the time discretization is completely determined by the space discretization.

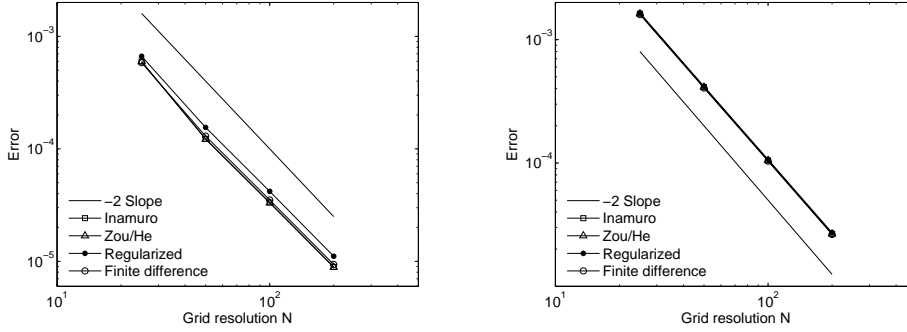


Fig. 4.5: Numerical accuracy in an oscillating channel. (a) $\alpha = 2$ (slow oscillations), (b) $\alpha = 5$ (fast oscillations).

In this simulation, the geometry is quadratic ($l_x = l_y = 1$). The Reynolds number is $\text{Re} = 1$, and the lattice velocity is set to $U_{\text{lb}} = 0.01$ at a reference resolution of $N = 10$. The numerical results are plotted in Fig. 4.5 (a) for a value of $\alpha = 2$ and on Fig. 4.5 (b) for a value of $\alpha = 5$. All other parameters are chosen to be the same as in the steady channel flow. Fig. 4.4 (a) in Section 4.4.1 can also be included in this discussion, as it corresponds to the limit $\alpha = 0$ in which the flow does not oscillate. It should be pointed out that, as α increases, inertial effects on the flow dominate over viscous effects. The velocity profile in the bulk becomes progressively independent of the boundary condition. As it can be seen from the simulations (see Fig. 4.5 (b)), the result is practically independent of the chosen boundary condition at a value of $\alpha = 5$.

4.4.3 Periodic array of vortices (2D)

The following benchmark measures the energy dissipation during time evolution of a laminar flow without external energy input. The initial velocity field consists of an array of counter-rotating vortices (see Walsh [99]). The flow is described by its value inside a periodic box of size $L \times L$. The flow velocity \mathbf{u}_0 at any time t_0 is derived from its stream function Ψ_0 , which is an eigenfunction of the Laplacian operator with eigenvalue λ :

$$\nabla^2 \Psi_0 = \lambda \Psi_0. \quad (4.41)$$

The time evolution of this flow is characterized by an exponential decrease of the velocity amplitude:

$$\mathbf{u}(t) = \mathbf{u}_0 e^{\lambda \nu (t-t_0)} \quad \text{for } t \geq t_0. \quad (4.42)$$

Among all eigenfunctions of the Laplacian, the following initial stream function Ψ_0 was selected for the benchmark (expressed in dimensionless variables):

$$\Psi_0(x, y) = \frac{1}{2\pi\sqrt{m^2 + n^2}} \cos(2\pi mx) \cos(2\pi ny), \quad (4.43)$$

where x and y are contained in the interval $[0, 1]$ and label the two space directions. The velocity field is given by

$$u_{x,0}(x, y) = \frac{\partial \Psi_0}{\partial y} = \frac{-\cos(2\pi mx) \sin(2\pi ny)}{\sqrt{\frac{m^2}{n^2} + 1}} \quad (4.44a)$$

and

$$u_{y,0}(x, y) = -\frac{\partial \Psi_0}{\partial x} = \frac{\sin(2\pi mx) \cos(2\pi ny)}{\sqrt{\frac{n^2}{m^2} + 1}}. \quad (4.44b)$$

It is clear that $\|\mathbf{u}_0(x, y)\| \leq 1$ for all values of x and y , which justifies the

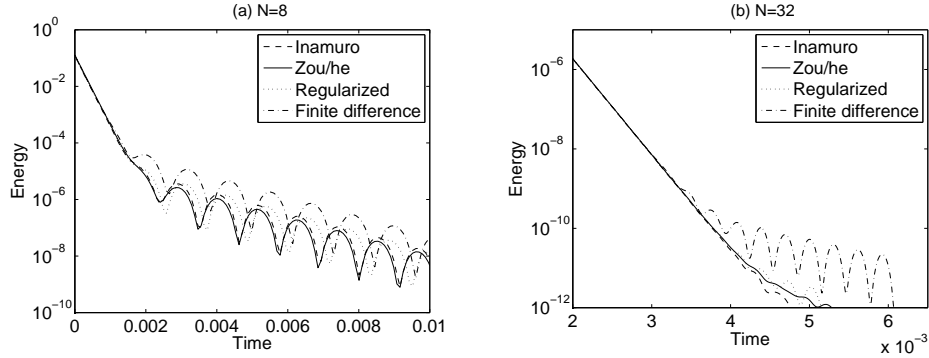


Fig. 4.6: Time-evolution of the average energy in the 2D periodic array of vortices. The curves for the linear and the non-linear finite difference methods overlap as far as one can distinguish visually. Therefore, only the linear case is presented. A boundary condition is considered to be of good quality if the point at which the curve oscillates and departs from an exponential decay occurs late.

choice of the reference velocity U . The decay rate λ is found by solving the eigenvalue problem (4.41):

$$\lambda = -(m^2 + n^2). \quad (4.45)$$

The initial pressure distribution of the flow is evaluated by solving the Poisson equation $\nabla^2 p = -(\mathbf{u} \cdot \nabla)\mathbf{u}$:

$$p = -(n^2 \cos(4\pi mx) + m^2 \cos(4\pi ny)). \quad (4.46)$$

The dimensionless pressure p is converted as follows to the density ρ_{lb} , expressed in lattice units: $\rho_{\text{lb}} = 1 + p\Delta x^2/(c_i^2\Delta t^2)$, where the discrete steps Δx and Δt are defined as in Eq. (4.32). This benchmark is formulated as an initial value problem. The initial condition is therefore set up with special care, using an approach described in Latt and Chopard [67] and based on an original suggestion of Skordos [93]. The particle populations are first initialized at an equilibrium distribution, based on the desired value of \mathbf{u} and p . The off-equilibrium parts of the particle populations are then instantiated with the help of Eq. (2.123), with numerically computed velocity gradients. Although the flow is periodic in both space directions, non-periodic boundary conditions are used, and the boundary velocity is explicitly imposed by use of the reviewed boundary conditions. The velocity along the boundaries of this sub-domain is set to the analytical value of Eq. (4.42) at every time step with the help of the five tested boundary conditions.

Fig. 4.6 displays the time-evolution of the average energy with different boundary conditions at a resolution $N = 8$ and a resolution $N = 32$. The Reynolds number is $\text{Re} = 1$, and the reference velocity in lattice units is set to $U_{\text{lb}} = 0.01$ at a grid resolution $N = 8$. In an initial regime, the energy decay is in all cases exponential, as predicted by Eq. (4.42). After reaching a critical value of the time t , the curve starts oscillating and deviates from the theoretical prediction. For the non-local boundary condition $BC4$, this deviation occurs earlier than for the local ones. It is therefore concluded that the non-local boundary condition are less accurate for this problem.

4.4.4 Dipole-wall collision (2D)

This benchmark, based on Refs. Clercx and Bruneau [22] and Latt and Chopard [67], analyzes the time evolution of a self-propelled dipole confined within a square box. The box is located in the geometrical domain $[-L, L] \times [-L, L]$ and implements no-slip walls. The initial condition describes two counter-rotating monopoles, one with positive core vorticity at the position (x_1, y_1) and one with negative core vorticity at (x_2, y_2) . This is obtained with an initial velocity field $\mathbf{u}_0 = (u_0, v_0)$ which reads as follows in dimensionless variables:

$$u_0 = -\frac{1}{2} \|\eta_e\| (y - y_1) e^{-(r_1/r_0)^2} + \frac{1}{2} \|\eta_e\| (y - y_2) e^{-(r_2/r_0)^2} \quad \text{and} \quad (4.47a)$$

$$v_0 = +\frac{1}{2} \|\eta_e\| (x - x_1) e^{-(r_1/r_0)^2} - \frac{1}{2} \|\eta_e\| (x - x_2) e^{-(r_2/r_0)^2}, \quad \text{where} \quad (4.47b)$$

$$r_i = \sqrt{(x - x_i)^2 + (y - y_i)^2}$$

Here, the distance to the monopole centers is defined as r_i , the parameter r_0 labels the diameter of a monopole and η_e its core vorticity.

The average kinetic energy of this system at a given time is defined by the expression

$$\mathcal{E}(t) = \frac{1}{2} \int_{-1}^1 \int_{-1}^1 \|\mathbf{u}\|^2(\mathbf{x}, t) d^2x, \quad (4.48)$$

and the average enstrophy by

$$\bar{\Omega}(t) = \frac{1}{2} \int_{-1}^1 \int_{-1}^1 \eta^2(\mathbf{x}, t) d^2x, \quad (4.49)$$

where $\eta = \partial_x v - \partial_y u$ is the flow vorticity.

The dipole described by Eqs. (4.47a) and (4.47b) develops a net momentum directed in the positive x -direction and is self-propelled toward the right wall. The collision between the dipole and the no-slip wall is characterized by a turbulent dynamics where the wall acts as a source of small-scale vortices that originate from detached boundary layers. This problem is therefore particularly interesting as a test for the ability of boundary conditions to reproduce the dynamics of boundary layers during collision. For this purpose, the maximum value of the flow enstrophy, which is reached during the dipole-wall collision, is evaluated and retained for a comparison among boundary conditions. As no analytical result for this flow is known, the measured values are compared against benchmark results obtained with a spectral method in Clercx and Bruneau [22].

In the benchmark, the initial core vorticity of the monopoles is fixed to $\eta_e = 299.5286$, which leads to an initial average kinetic energy of $\mathcal{E}(0) = 2$. Furthermore, the Reynolds number and the monopole radius are set to $\text{Re} = 625$ and $r_0 = 0.1$. The lattice velocity is set to $U_b = 0.01$ at a reference resolution of $N = 300$. The monopoles are aligned symmetrically with the box, in such a way that the dipole-wall collision is frontal and takes place in the middle of the wall. The position of the monopole centers is $(x_1, y_1) = (0, 0.1)$ and $(x_2, y_2) = (0, -0.1)$. As in Section 4.4.3, we use the approach described in Latt and Chopard [67] to set up the initial condition. The initial pressure is evaluated numerically, by solving the Poisson equation with a successive overrelaxation (SOR) scheme, using an algorithm described e.g. Griebel et al. [42]. The off-equilibrium parts of the particle populations are then instantiated with the help of Eq. (2.123), with numerically computed velocity gradients.

The boundary conditions *BC1* and *BC2* could not be used for this problem, because they experienced numerical instabilities at the required Re number (the benchmark values in the literature start at $\text{Re} = 625$). These instabilities are due to an inherent limitation of the boundary conditions, and do

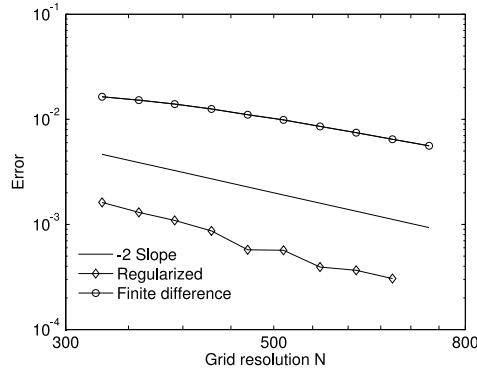


Fig. 4.7: Accuracy on the value of the enstrophy peak in a 2D dipole-wall collision at $Re = 625$. Boundary conditions $BC1$ and $BC2$ could not be tested, because they are numerically unstable at this Reynolds number.

not originate from the way the initial condition or the boundary condition in corner nodes are treated. This possibility was ruled out by initializing the population functions with an equilibrium distribution, and keeping the corner nodes at an equilibrium distribution with constant pressure throughout the simulation. Although this approach to setting up initial states and boundaries tends to be exceptionally stable, the simulations using $BC1$ and $BC2$ were still subject to numerical instabilities.

The numerical results are presented in Fig. 4.7. Please be aware that a resolution of $N = 300$, for example, stands for a total grid size of 600×600 , as the size of the system is 2×2 in dimensionless variables. For a Reynolds number $Re = 625$, both boundary conditions $BC3$, $BC4$, the error decreases roughly at a second-order rate. The local boundary condition $BC3$ is however an order of magnitude more accurate than the other candidates. This is striking, because $BC3$ is less accurate in all benchmarks of laminar flows. It is concluded that the local, cell-based approach of $BC3$ is particularly well adapted to reproduce the dynamics of a boundary layer. A possible interpretation is that the finite difference approximation of velocity gradients in $BC4$ becomes inaccurate in presence of a turbulent boundary layer due to small-scale velocity variations.

4.4.5 Rectangular steady channel flow (3D)

This problem is a 3D generalization of the channel flow presented in Section 4.4.1. This time, the channel extends in z -direction. At every fixed

value of z , the cross section is rectangular, extending from $x = 0$ to $x = L$, and from $y = 0$ to $y = \zeta L$, where ζ is the aspect ratio of the cross section. The pressure is independent of the x - and the y -coordinates, and decreases linearly as a function of z . The velocity has a non-zero component in z -direction only, and depends on x and y : $u_z = u_z(x, y)$. As in the 2D case, nonlinear contributions to the Navier–Stokes equation cancel out, and the velocity profile u_z is described as a solution of the following Poisson equation:

$$\nabla^2 u_z \equiv \frac{\partial^2 u_z}{\partial x^2} + \frac{\partial^2 u_z}{\partial y^2} = \frac{\partial p}{\partial z}. \quad (4.50)$$

As this equation is non-homogeneous, the general solution is described as the sum of a particular solution u_p and a general solution u_h to the homogeneous problem $\nabla^2 u_z = 0$. The particular solution can be taken from the 2D flow in Section 4.4.1. In dimensionless variables, it reads

$$u_p(x, y) = 4(x - x^2). \quad (4.51)$$

This settles the choice for the reference velocity U as being the maximum velocity reached in a 2D channel. The pressure drop is the same in the 3D as in the 2D case: $\partial p / \partial z = -8 / \text{Re}$. The complete analytic solution to this 3D channel flow can be found for example in Yih [102]

$$u_z(x, y) = 4(x - x^2) + \frac{32}{\pi^3} \sum_{n=0}^{+\infty} \left(\frac{(-1)^n}{(2n+1)^3} \cos((2n+1)\pi x) \frac{\cosh\left((2n+1)\frac{\pi y}{\zeta}\right)}{\cosh\left((2n+1)\frac{\pi}{2}\zeta\right)} \right). \quad (4.52)$$

Fig. 4.8 (a) shows results of numerical accuracy at $\text{Re} = 10$. The reference velocity is $U_{\text{lb}} = 0.01$ at $N = 50$, and the aspect ratio is $\zeta = 1$. Numerical stability is explored in Fig. 4.8 (b) for a varying grid resolution. The numerical results in this 3D benchmark are similar to those of the 2D application in Section 4.4.1. The boundary condition *BC1* is again much more accurate than boundary conditions that replace all particle populations. As in the 2D case, all but one component of the strain rate tensor vanish in this flow, and some potential deficiencies of *BC1*, as the one shown in Eq. (4.16), might not be visible in the benchmark. An unexpected result is that boundary condition *BC2*, which in 2D applications achieves results very similar to those of *BC3*, falls in the same category as the non-local boundary condition *BC4* in this 3D case. Another remarkable difference with respect to 2D results is

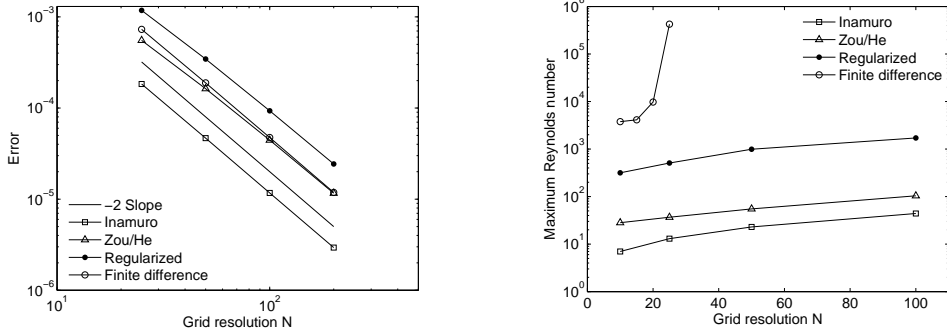


Fig. 4.8: (a) Numerical accuracy in a 3D channel flow. (b) Numerical stability in a 3D channel flow. The maximum Reynolds number which can be reached before numerical instabilities appear is plotted. Boundary condition based of finite difference approximation (BC_4) are unconditionally stable at high grid resolution. Numerical values are therefore only reported in the unstable regime, up to $N = 25$ for BC_4 .

that the non-local boundary conditions achieve unconditional numerical stability in 3D, as soon as the grid resolution exceeds a certain threshold value. At this point, further investigations would be needed to decide if this difference is due to different flow geometries (the analytical solutions in Eqs. (4.35) and (4.52) are qualitatively different) or if 3D simulations are inherently more stable than 2D ones for this type of boundary conditions.

4.5 Discussion

As it is concluded from applying the results of a perturbative analysis to boundary nodes, all methods are second-order accurate with respect to the grid resolution. This implies that their error varies asymptotically at the same rate. Beyond this asymptotic estimate, the accuracy experienced in numerical simulation differs however from one boundary condition to another, depending on the flow geometry and the grid resolution. A way to understand these differences might be to take the perturbative analysis to a higher order $\mathcal{O}(\text{Kn}^2)$ and discuss the couplings between higher order terms and the hydrodynamic scales. This technique, employed for example in Dellar [26], is used understand the asymptotic low Mach-number behavior of a family of lattice Boltzmann models. Instead of doing this, this thesis produces benchmark results for the boundary conditions in many different geometries. Those results are intended to serve as a reference to help select the most

appropriate boundary condition for a given lattice Boltzmann simulation. It is emphasized once more that the results are valid only for straight boundaries which are aligned with the numerical grid. Different conclusions can be expected when off-lattice boundaries are implemented.

In 2D simulations and at low Reynolds numbers, the Inamuro and Zou/He boundary conditions $BC1$ and $BC2$ are found to produce the most accurate results. Both of them preserve the known particle populations on a boundary node, a fact which probably explains their excellent benchmark performance. They retrieve a large amount of information from the bulk of the fluid, and manipulate only few particle populations (3 out of 9). It seems plausible that by doing this, they retain information on higher order terms which are not visible in the hydrodynamic terms of the perturbative analysis. The Inamuro method is also very accurate in the 3D benchmark, but this performance is not reproduced by the Zou/He condition, which in this case compares to non-local boundary conditions. The main deficiency of $BC1$ and $BC2$ is that they are numerically unstable at (even moderately) high Reynolds numbers. They could for example not be used to simulate the turbulent dipole-wall collision presented in Section 4.4.4.

In conclusion, $BC1$ and $BC2$ are the boundary conditions of choice for the simulation of laminar 2D flows, when high accuracy is important. The Inamuro boundary condition $BC1$ is also a good candidate for laminar 3D flows, although it is not an explicit method, and thus tends to be complicated to implement. The extension of $BC2$ to 3D flow does not seem to make much sense, as it does not exhibit exceptional accuracy, and is less stable than other approaches.

The Regularized boundary condition $BC3$ uses a hybrid approach, as it is local like $BC1$ and $BC2$, but it replaces all particle populations like $BC4$. As such, it is less accurate than $BC1$ and $BC2$ in laminar flows, but it has the ability to reach much higher Reynolds numbers. Furthermore, simulations of a turbulent dipole-wall collision show that it reproduces the dynamics of boundary layers more accurately than all other boundary conditions. Because of its numerical stability, and because it is easy to implement in 2D and 3D applications, $BC3$ is a good general-purpose method, and seems to be the best candidate for high Reynolds number flows.

The non-local boundary condition $BC4$, which use a finite difference scheme to approximate the off-equilibrium part of particle populations, exhibit the best numerical stability. In the laminar 3D benchmark, it is even uncon-

ditionally stable on a sufficiently large grid. Its main disadvantage is the non-locality of its algorithm. This is at odds with the basic principles of the LBM, and may substantially increase the complexity of a code, and may even be an obstacle to parallelizing the program in a straightforward way. On the other hand, the boundary condition BC_4 are very general and may adapt well to a larger software project. They are for example not bound to the lattice structure, and can be extended to the case of off-lattice boundaries, like the one described in Guo and Zheng [43].

Chapter 5

Simulation of non-Newtonian fluids

In this chapter we will discuss the numerical schemes used to simulate two classes of non-Newtonian fluids, namely the generalized Newtonian and viscoelastic fluids.

5.1 Generalized Newtonian fluids

The method presented in this section can be found in Malaspinas et al. [71]. As seen in Chapter 2 the relaxation time τ and the kinematic viscosity ν are related through

$$\nu = c_l^2 \left(\tau - \frac{1}{2} \right). \quad (5.1)$$

The generalized Newtonian fluids have a constitutive equation that expresses the kinematic viscosity as a function of the shear rate $\dot{\gamma}$ (see Eq. (3.2) for the definition of $\dot{\gamma}$)

$$\nu = \mathcal{F}(\dot{\gamma}), \quad (5.2)$$

Eliminating ν from these equations, one has that the relaxation time is related to the shear rate through the function \mathcal{F}

$$\tau(\mathbf{x}, t) = \frac{\mathcal{F}(\dot{\gamma}(\mathbf{x}, t))}{c_l^2} + \frac{1}{2}. \quad (5.3)$$

Furthermore we know that the shear rate can be computed from the shear rate tensor \mathbf{S} , through (see Agassant et al. [1] for example)

$$\dot{\gamma} = \sqrt{2S_{\alpha\beta}S_{\alpha\beta}}. \quad (5.4)$$

Then using Eq. (2.125), we find

$$\dot{\gamma} = \frac{1}{2c_l^2\rho\tau} \sqrt{P_{\alpha\beta}^{(1)}P_{\alpha\beta}^{(1)}}. \quad (5.5)$$

As a consequence, we can compute the relaxation time with the second order moment of the off-equilibrium part of the distribution functions

$$\mathbf{P}^{(1)} = \sum_i \boldsymbol{\xi}_i \boldsymbol{\xi}_i f_i^{(1)}, \quad f_i^{(1)} = f_i - f_i^{(0)}. \quad (5.6)$$

Since the space and time dependence of τ does not change the Chapman–Enskog expansion of Section 2.3, we only need to specify the function \mathcal{F} in order to be able to recompute τ at each lattice point and timestep. Unfortunately, since $\dot{\gamma}$ also depends on τ (see Eq. (5.5)) the equation to solve in order to recompute the relaxation time is in general implicit.

In this thesis we used the Carreau constitutive equation, which has already been presented in Section 3.1 and for which the \mathcal{F} function is given by

$$\mathcal{F}(\dot{\gamma}) = (\nu_0 - \nu_\infty) \left(1 + (\lambda\dot{\gamma})^2\right)^{\frac{n-1}{2}} + \nu_\infty. \quad (5.7)$$

Using Eqs. (5.5) and (5.3), we can write the relation between the deviatoric stress tensor, $\mathbf{P}^{(1)}$, and the relaxation time, τ , for the Carreau model

$$\tau = \left(\frac{\nu_0 - \nu_\infty}{c_l^2}\right) \left(1 + \frac{\lambda^2}{4c_l^4 \rho^2 \tau^2} P_{\alpha\beta}^{(1)} P_{\alpha\beta}^{(1)}\right)^{\frac{n-1}{2}} + \frac{\nu_\infty}{c_l^2} + \frac{1}{2} \quad (5.8)$$

As pointed out previously, these equations are implicit in τ and therefore in order to be completely general, one must solve them using a root finding algorithm like the Newton–Raphson method (see [85] for example). An alternative faster method would be to use fixed-point iterations. Anyway, this is very expensive in terms of computing time, and is unnecessary when dealing with stationary flows, where one can just replace τ in the r.h.s. by its value at the preceding timestep

$$\tau(\mathbf{x}, t) = \left(\frac{\nu_0 - \nu_\infty}{c_l^2}\right) \left(1 + \frac{\lambda^2}{4c_l^4 \rho^2 \tau(\mathbf{x}, t-1)^2} P_{\alpha\beta}^{(1)} P_{\alpha\beta}^{(1)}\right)^{\frac{n-1}{2}} + \frac{\nu_\infty}{c_l^2} + \frac{1}{2}. \quad (5.9)$$

The method can also be applied to simulate the flow of materials with yield stress (viscoplastic) with same regularization technique for low shear rates [98].

5.2 Viscoelastic fluids

As seen in Chapter 3 the coupling between the constitutive equations and the incompressible Navier–Stokes equations is done through the velocity field and the viscoelastic stress tensor. On the other hand, the interaction of the

fluid with the viscoelastic stress tensor is only an effect of the velocity and its gradient. The idea used in order to do this coupling is similar to what is done in Denniston et al. [27]. The Navier–Stokes equations are simulated with the weakly compressible scheme (see Eq. (2.126)), and each component of the constitutive equation will be modelled with a lattice Boltzmann advection–diffusion scheme, to which a source term is added.

5.2.1 The advection–diffusion with source scheme

To simulate each component of the constitutive equation, we will use a modified version of the advection–diffusion LBM (for a reference on the advection–diffusion scheme see for example Guo et al. [44], Latt [66], Chopard et al. [21]). In order to simplify the notations, we first discuss the general algorithm and then apply it to the viscoelastic fluids. We modify the BGK equation and add an “external source” term \mathcal{G}

$$\partial_t f + \boldsymbol{\xi} \cdot \nabla f = -\frac{1}{\tau} (f - f^{(0)}) + \frac{1}{\rho} \mathcal{G} f. \quad (5.10)$$

We will also assume that the density flux, \mathbf{j} , is different if computed from f than from $f^{(0)}$ (i.e. $\mathbf{j} = \int d\boldsymbol{\xi} \boldsymbol{\xi} f \neq \int d\boldsymbol{\xi} \boldsymbol{\xi} f^{(0)}$). The generic mass conservation equation obtained from Eq. (2.11) with $\mathcal{A} = 1$ is

$$\partial_t \rho + \nabla \cdot \mathbf{j} = \mathcal{G}. \quad (5.11)$$

One needs to compute \mathbf{j} . To this aim, we will use the Chapman–Enskog expansion (as in Section (2.3)) to get it.

Before proceeding further we will expand the Maxwell–Boltzmann equilibrium distribution $f^{(0)}$ up to order one in Hermite polynomials

$$f^{(0)} = \omega(\boldsymbol{\xi}) (\rho + \boldsymbol{\xi} \cdot (\rho \mathbf{u})). \quad (5.12)$$

Replacing f by $f^{(0)}$ in Eq. (5.10) and projecting it on the Hermite polynomial, one gets

$$\partial_t a_{0\boldsymbol{\alpha}}^{(n)} + \nabla \cdot a_{0\boldsymbol{\alpha}}^{(n+1)} + (\nabla a^{(n-1)\boldsymbol{\alpha}} + \text{perm}) = \frac{\mathcal{G}}{\rho} a_{0\boldsymbol{\alpha}}^{(n)}, \quad (5.13)$$

where “perm” represents all the cyclic index permutations. For $n = 0$, one gets from Eq. (2.44)

$$\partial_t \rho + \nabla \cdot (\rho \mathbf{u}) = \mathcal{G}. \quad (5.14)$$

We now use the multiscale *Ansatz*, $f = f^{(0)} + f^{(1)}$ in Eq. (5.10), and project it on the Hermite basis, and we find in a similar way as for Eq. (2.52)

$$\partial_t a_{0\boldsymbol{\alpha}}^{(n)} + \nabla \cdot a_{0\boldsymbol{\alpha}}^{(n+1)} + (\nabla a_{0\boldsymbol{\alpha}}^{(n)} + \text{perm}) = -\frac{1}{\tau} a_{1\boldsymbol{\alpha}}^{(n)} + \frac{\mathcal{G}}{\rho} a_{0\boldsymbol{\alpha}}^{(n)}. \quad (5.15)$$

Because of the orthogonality properties of the Hermite polynomials, we have that $a_{0\alpha}^{(n)} = 0$, for $n \geq 2$. Therefore for $n = 1$, this last equation becomes

$$\begin{aligned} -\frac{1}{\tau}a_{1\alpha}^{(1)} &= \partial_t a_{0\alpha}^{(1)} + \nabla a_{0\alpha}^{(0)} - \frac{\mathcal{G}}{\rho}a_{0\alpha}^{(1)}, \\ &= \partial_t(\rho\mathbf{u}) + \nabla\rho - \mathcal{G}\mathbf{u}, \\ &= \rho\partial_t\mathbf{u} - \mathbf{u}\nabla \cdot (\rho\mathbf{u}) + \nabla\rho, \end{aligned} \quad (5.16)$$

where in the last line we used Eq. (5.14). The flux $\mathbf{j} = \mathbf{j}^{(0)} + \mathbf{j}^{(1)}$ is therefore given by

$$\mathbf{j} = \rho\mathbf{u} - \tau(\rho\partial_t\mathbf{u} - \mathbf{u}\nabla \cdot (\rho\mathbf{u})). \quad (5.17)$$

We finally we find the following macroscopic equation

$$\partial_t\rho + \nabla \cdot (\rho\mathbf{u}) = \mathcal{G} + \kappa_p\nabla^2\rho + \nabla \cdot [\tau(\rho\partial_t\mathbf{u} - \mathbf{u}\nabla \cdot (\rho\mathbf{u}))], \quad (5.18)$$

where $\kappa_p = \tau$ stands for the diffusivity coefficient. Comparing this equation with the form of the constitutive equation, we see that the two last terms of its r.h.s are in excess (we will discuss them later). These terms simply reflect the fact that the BGK Boltzmann equation is not representing correctly the macroscopic advection–diffusion, since the underlying model is designed to mimic the physics of fluids.

We now have to discretize the velocity space using the Gauss–Hermite quadrature. In this case, we need the quadrature to be exact only up to the third polynomial degree (see Appendix B). Following the same procedure as in Section 2.4 we have the equilibrium distribution function $f_i^{(0)}$ is given by

$$f_i^{(0)} = w_i\rho \left(1 + \frac{\boldsymbol{\xi}_i \cdot \mathbf{u}}{c_i^2} \right). \quad (5.19)$$

Because of the rescaling of the quadrature abscissae, also discussed in Section 2.4, we have that $a_{1\alpha}^{(1)}$ is modified in the following way

$$a_{1\alpha}^{(1)} = -\tau(\rho\partial_t\mathbf{u} - \mathbf{u}\nabla \cdot (\rho\mathbf{u}) + c_i^2\nabla\rho). \quad (5.20)$$

As a consequence the macroscopic Eq. (5.18) remains unchanged, but the diffusivity, κ_p is rescaled

$$\kappa_p = c_i^2\tau. \quad (5.21)$$

Finally we can reconstruct the off-equilibrium distribution function, $f_i^{(1)}$ as

$$f_i^{(1)} = -\frac{w_i\tau}{c_i^2}\boldsymbol{\xi}_i \cdot (\rho\partial_t\mathbf{u} - \mathbf{u}\nabla \cdot (\rho\mathbf{u}) + c_i^2\nabla\rho). \quad (5.22)$$

We are now left with the time discretization of Eq. (5.10). Following the same procedure as in Section 2.5 we define a “effective equilibrium” distribution function

$$f_i^{\text{eq}} \equiv f^{(0)} + \tau \frac{\mathcal{G}}{\rho} f_i^{(0)}. \quad (5.23)$$

Applying the change of variables of Eq. (2.112) will lead to the following numerical scheme

$$\begin{aligned} \bar{f}_i(\mathbf{x} + \boldsymbol{\xi}_i, t + 1) - \bar{f}_i(\mathbf{x}, t) = & -\frac{1}{\bar{\tau}} \left(\bar{f}_i(\mathbf{x}, t) - f_i^{(0)}(\rho, \mathbf{u}) \right) \\ & + \left(1 - \frac{1}{2\bar{\tau}} \right) \frac{\mathcal{G}}{\rho} f_i^{(0)}(\rho, \mathbf{u}), \end{aligned} \quad (5.24)$$

where $\bar{\tau} = \tau - 1/2$. The change of variables has an effect on the diffusivity κ_p

$$\kappa_p = c_l^2 \left(\bar{\tau} - \frac{1}{2} \right), \quad (5.25)$$

and the computation of the physical density ρ

$$\rho = \bar{\rho} + \frac{\mathcal{G}}{2}. \quad (5.26)$$

The physical velocity, \mathbf{u} , will not be simulated by this scheme but given as an external field determined by solving the Navier–Stokes equations. It will therefore remain unaffected by the change of variables.

5.2.2 Constitutive equation simulation

We have described in the last section the generic numerical scheme to be used. More details about how to apply it to the viscoelastic constitutive equation are given here. In fact the evolution of each component of the conformation tensor \mathbf{A} is simulated, by a separate “advection–diffusion with source” lattice Boltzmann model. Defining $f_{i\alpha\beta}$ as the distribution functions that describe the components of the conformation tensor, $A_{\alpha\beta}$ and $\mathcal{G}_{\alpha\beta}$ the components of the source term

$$A_{\alpha\beta} = \sum_i f_{i\alpha\beta} + \frac{\mathcal{G}_{\alpha\beta}}{2}, \quad (5.27)$$

we can rewrite the equilibrium distribution function as

$$f_{i\alpha\beta}^{(0)} = A_{\alpha\beta} \left(1 + \frac{\boldsymbol{\xi}_i \cdot \mathbf{u}}{c_l^2} \right), \quad (5.28)$$

This leads to the macroscopic equations by replacing ρ by $A_{\alpha\beta}$ (see (5.18))

$$\begin{aligned} \partial_t A_{\alpha\beta} + (\mathbf{u} \cdot \nabla) A_{\alpha\beta} = & \kappa_p \nabla^2 A_{\alpha\beta} + \mathcal{G}_{\alpha\beta} \\ & + \frac{\kappa_p}{c_l^2} \nabla \cdot (A_{\alpha\beta} \partial_t \mathbf{u} - \mathbf{u} \nabla \cdot (A_{\alpha\beta} \mathbf{u})), \end{aligned} \quad (5.29)$$

where $\mathcal{G}_{\alpha\beta}$ depends on the constitutive equation. For the Oldroyd-B and the FENE-P model it is given respectively by (see Eqs. (3.33) and (3.35))

$$\mathcal{G}_{\alpha\beta} = -\frac{1}{\lambda} (A_{\alpha\beta} - \delta_{\alpha\beta}) + A_{\alpha\gamma} \partial_\gamma u_\beta + \partial_\gamma u_\alpha A_{\gamma\beta}, \quad (5.30)$$

$$\mathcal{G}_{\alpha\beta} = -\frac{1}{\lambda} (hA_{\alpha\beta} - b\delta_{\alpha\beta}) + A_{\alpha\gamma} \partial_\gamma u_\beta + \partial_\gamma u_\alpha A_{\gamma\beta}. \quad (5.31)$$

As can be seen by comparing Eq. (5.29) with Eq. (3.33) (or Eq. (3.35)) in the macroscopic equation obtained from the Boltzmann equation we have a “diffusive term” and two “error terms”. As has been shown in El-Kareh and Leal [32], this diffusive term is present in real fluids, but is very small, therefore making the diffusivity very small ($\tau \rightarrow 1/2$) will also make the error terms negligible. In our simulations we will fix it by making it much smaller than the polymer viscosity. Typically we will have $\kappa_p/\mu_p \sim 10^{-6}$.

The numerical scheme will be the following. At time t the conformation tensor \mathbf{A} is computed and converted into the viscoelastic stress tensor $\mathbf{\Pi}$ according to Eqs. (3.31) and (3.38). Then, using a finite difference scheme, its divergence is evaluated and considered as a force that will be communicated to the Navier–Stokes scheme according to Eq. (3.42). In turn, the velocity gradient will be determined by a finite difference scheme once the Navier–Stokes equations are solved for the velocity field and used as given data in the constitutive equations according to Eq. (3.43) (see Fig. 5.1 for a schematic view of the communication needed at time t). This method relies on a decoupled approach in the sense that the Navier–Stokes and the constitutive equations are solved independently at each time iteration.

5.2.3 Conformation tensor boundary conditions

As seen in Chapter 4 we have missing distribution functions on the boundaries. We therefore need to impose some quantities in order to recover them. In the case of the advection–diffusion, the Gauss–Hermite quadrature is only needed to be exact up to order three, thus we only need the D2Q5 or D3Q7 lattices (see Appendix B) and hence we only have one unknown population (see Fig. 5.2). It follows that we need only one constraint in order to impose

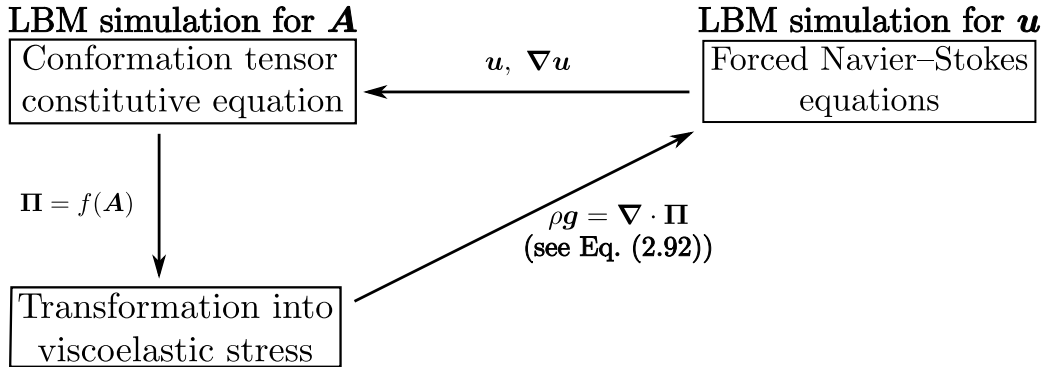


Fig. 5.1: The communication steps that need to be done between two time steps.

the missing distribution function. Unfortunately there exists no equivalent to the Newtonian fluid no-slip boundary condition for the viscoelastic tensor. In some rare cases we know analytical solutions for the value of the conformation tensor, \mathbf{A} (see Subsection 6.2.3), but in simulations with complex geometries we need of course more generic solutions. It is therefore needed to interpolate the value of the viscoelastic tensor on the boundaries of our domain. In order to explain the method we will consider the upper flat 2D wall case (see Fig. 5.2), although the method can be generalized straightforwardly to the 3D case. In the case of the scheme for the viscoelastic fluid,

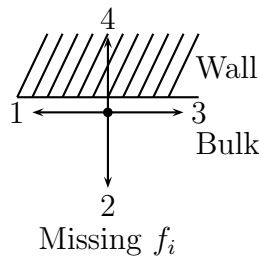


Fig. 5.2: A boundary node with the known f_i 's (0, 1, 3, 4) and the missing one (2).

we know that the distribution function has the following Chapman–Enskog

expansion (see Eqs. (5.19), and (5.22))

$$f_{i\alpha\beta} = w_i A_{\alpha\beta} \left(1 + \frac{\boldsymbol{\xi}_i \cdot \mathbf{u}}{c_l^2} \right) - \frac{w_i \tau}{c_l^2} \boldsymbol{\xi}_i \cdot (A_{\alpha\beta} \partial_t \mathbf{u} - \mathbf{u} \nabla \cdot (A_{\alpha\beta} \mathbf{u}) + c_l^2 \nabla A_{\alpha\beta}). \quad (5.32)$$

The rest population, $i = 0$, contains the information about the conformation tensor, since

$$A_{\alpha\beta} = \frac{f_{0\alpha\beta}}{w_0}. \quad (5.33)$$

Using this equation, the missing population can be computed straightforwardly as (see Eq. (2.104))

$$A_{\alpha\beta} = f_{0\alpha\beta} + f_{1\alpha\beta} + f_{2\alpha\beta} + f_{3\alpha\beta} + f_{4\alpha\beta}, \quad (5.34)$$

which leads to

$$f_{2\alpha\beta} = -f_{0\alpha\beta} \left(1 - \frac{1}{w_0} \right) - f_{1\alpha\beta} - f_{3\alpha\beta} - f_{4\alpha\beta}. \quad (5.35)$$

This boundary condition is not the only one that can be implemented. We can also interpolate the value of the conformation tensor by using the term $\nabla A_{\alpha\beta}$ contained in $f_{i\alpha\beta}$. Then using a backward finite difference scheme, we can compute the value of the conformation tensor. For the wall in Fig. 5.2 a decentered second order finite difference scheme give us for estimating the conformation tensor gradient

$$\partial_y A_{\alpha\beta}(x, y) = -\frac{1}{2} (A_{\alpha\beta}(x, y-2) - 4A_{\alpha\beta}(x, y-1) - 3A_{\alpha\beta}(x, y)). \quad (5.36)$$

From Eq. (5.32) we have that $\partial_y A_{\alpha\beta}(x, y)$ is contained in $f_{4\alpha\beta}$ and given by

$$\partial_y A_{\alpha\beta} = \frac{1}{\tau} \left(-\frac{f_{4\alpha\beta}}{w_4} + A_{\alpha\beta} \right). \quad (5.37)$$

Replacing this relation in Eq. (5.36) one can solve for $A_{\alpha\beta}$

$$A_{\alpha\beta}(x, y) = \frac{2\tau}{2-3\tau} \left(\frac{f_{4\alpha\beta}}{w_4\tau} - \frac{1}{2} (A_{\alpha\beta}(x, y-2) - 4A_{\alpha\beta}(x, y-1)) \right) \quad (5.38)$$

and then recompute the missing distribution function $f_{2\alpha\beta}$ from Eq. (5.35). We can also, for steady state simulations and for no-slip walls, simplify the Oldroyd-B constitutive equation as

$$-\frac{1}{\lambda} (A_{\alpha\beta} - \delta_{\alpha\beta}) + A_{\alpha\gamma} \partial_\gamma u_\beta + \partial_\gamma u_\alpha A_{\gamma\beta} = 0, \quad (5.39)$$

which gives us an algebraic system of equations to solve, since the velocity gradient can be computed from the Navier–Stokes solver by using a finite difference scheme (see Eqs. (4.29) and (4.30)). Using a numerical solver, one can compute the roots of this system and get the values of the components of the conformation tensor $A_{\alpha\beta}$ on the boundaries. Then from Eq. (5.35) one can recover $f_{2\alpha\beta}$. Unfortunately these two methods did not improve the results obtained with Eq. (5.35).

5.3 Description of the algorithm

We have shown how the lattice Boltzmann scheme for advection-diffusion with source can simulate the constitutive equation. We are now going to describe the algorithm used in the following section. A time-step will be composed of four distinct operations. The first and the second will be the resolution, respectively, of the incompressible Navier–Stokes and of the constitutive equations (see Eqs. (2.16), (2.91) and (3.33))

$$\begin{aligned}\nabla \cdot \mathbf{u} &= 0, & \partial_t \mathbf{u} + (\mathbf{u} \cdot \nabla) \mathbf{u} &= -\nabla \cdot (-p\mathbf{I} + 2\mu_s \mathbf{S} - \mathbf{\Pi}), \\ \partial_t \mathbf{A} + (\mathbf{u} \cdot \nabla) \mathbf{A} &= -\frac{1}{\lambda} (\mathbf{A} - \mathbf{I}) + \mathbf{A} \cdot \nabla \mathbf{u} + (\nabla \mathbf{u})^T \cdot \mathbf{A}.\end{aligned}$$

The third step will be the coupling between the bulk of the two schemes: the advection-diffusion with source will receive the velocity field and the “source” term (see Eq. (5.30))

$$\mathcal{G}_{\alpha\beta} = -\frac{1}{\lambda} (A_{\alpha\beta} - \delta_{\alpha\beta}) + A_{\alpha\gamma} \partial_\gamma u_\beta + \partial_\gamma u_\alpha A_{\gamma\beta},$$

from the Navier–Stokes scheme, and the Navier–Stokes solver will receive an external forcing term (see Eq. (3.31))

$$\rho \mathbf{g} = \nabla \cdot \mathbf{\Pi} = \nabla \cdot \left(-\frac{\mu_p}{\lambda} (\mathbf{A} - \mathbf{I}) \right), \quad (5.40)$$

computed from the advection-diffusion with source scheme. Finally the boundaries will be coupled. A more detailed explanation of the procedure can be found below (we discuss the 2D case for simplicity, but the generalization to 3D is straightforward):

1. Collision and propagation of the advection–diffusion with source scheme (see Eq. (5.24))

$$\begin{aligned}f_{i\alpha\beta}(\mathbf{x} + \boldsymbol{\xi}_i, t + 1) &= f_{i\alpha\beta}(\mathbf{x}, t) - \frac{1}{\tau} \left(f_{i\alpha\beta}(\mathbf{x}, t) - f_{i\alpha\beta}^{(0)}(A_{\alpha\beta}, \mathbf{u}) \right) \\ &\quad + \left(1 - \frac{1}{2\tau} \right) \frac{\mathcal{G}_{\alpha\beta}}{A_{\alpha\beta}} f_i^{(0)}(A_{\alpha\beta}, \mathbf{u}).\end{aligned}$$

2. Collision and propagation of the forced Navier–Stokes scheme (see Eq. (2.126))

$$f_i(\mathbf{x} + \boldsymbol{\xi}_i, t + 1) - f_i(\mathbf{x}, t) = -\frac{1}{\tau} \left(f_i(\mathbf{x}, t) - f_i^{(0)}(\rho, \mathbf{u}) \right) + \left(1 - \frac{1}{2\tau} \right) F_i,$$

where $\rho \mathbf{g} = \nabla \cdot \mathbf{\Pi}$.

3. Coupling of the bulk of the two schemes. On the one hand, with the Navier–Stokes scheme, we compute the velocity and its gradients which will be used for the computation of the equilibrium and the source of the advection–diffusion scheme. The velocity gradients are computed with a centered finite difference scheme

$$\begin{aligned} \partial_x \mathbf{u}(x, y) &= \frac{\mathbf{u}(x + 1, y) - \mathbf{u}(x - 1, y)}{2}, \\ \partial_y \mathbf{u}(x, y) &= \frac{\mathbf{u}(x, y + 1) - \mathbf{u}(x, y - 1)}{2}. \end{aligned}$$

On the other hand the advection–diffusion scheme gives us the conformation tensor, which we transform into the viscoelastic stress tensor, $\mathbf{\Pi}$, using Eq. (3.31). We then compute the divergence of $\mathbf{\Pi}$ using a centered finite difference scheme

$$\begin{aligned} \partial_x \mathbf{\Pi}(x, y) &= \frac{\mathbf{\Pi}(x + 1, y) - \mathbf{\Pi}(x - 1, y)}{2}, \\ \partial_y \mathbf{\Pi}(x, y) &= \frac{\mathbf{\Pi}(x, y + 1) - \mathbf{\Pi}(x, y - 1)}{2}. \end{aligned}$$

We then communicate $\rho \mathbf{g} = \nabla \cdot \mathbf{\Pi}$ to the Navier–Stokes solver.

4. Coupling of the boundaries of the two schemes. On the boundaries, the regularized technique of Latt and Chopard [68] is applied for the Navier–Stokes solver and the scheme discussed in the preceding subsection (see Eq. (5.33) and (5.35)) is applied for the constitutive equation. The divergence of the viscoelastic stress and the velocity gradients cannot be computed using a centered finite difference for the direction perpendicular to the wall. Therefore the finite difference scheme used is the same as the one described in Subsection 4.3.4, and for a wall as described by Fig. 5.2 is given by

$$\begin{aligned} \partial_y \mathbf{u}(x, y) &= \frac{-3\mathbf{u}(x, y) + 4\mathbf{u}(x, y - 1) - \mathbf{u}(x, y - 2)}{2}, \\ \partial_y \mathbf{\Pi}(x, y) &= \frac{-3\mathbf{\Pi}(x, y) + 4\mathbf{\Pi}(x, y - 1) - \mathbf{\Pi}(x, y - 2)}{2}. \end{aligned}$$

At this point all the schemes have the needed information to proceed to the next time-step.

Chapter 6

Numerical results

In this chapter we will validate our models by comparing the corresponding results to analytical solutions, or to those of other numerical methods. For the generalized Newtonian fluids, the benchmarks used are the 2D planar Poiseuille flow, and the 2D planar 4:1 contraction for the generalized Newtonian fluids. For the viscoelastic fluids, we will also use the 2D planar Poiseuille flow, the steady 2D simplified four-roll mill and the 3D Taylor–Green vortex decay flow. All the codes of the present chapter were produced with Palabos [87] (the successor of OpenLB and xFlows [86]), an open source library for the lattice Boltzmann method.

6.1 Generalized Newtonian fluids

In this section we will present the planar Poiseuille and the 4:1 contraction benchmarks that we carried out for the Carreau model. We will validate and also briefly discuss the accuracy of the method. The results presented in the sequel can be also found in Malaspinas et al. [71].

6.1.1 Steady planar Poiseuille flow

The 2D planar Poiseuille flow is the first benchmark that we performed using the model of the generalized Newtonian fluids presented in Section 5.1. Thanks to the simplicity of the geometry of the problem one is able to compute semi-analytical solutions and therefore make a quantitative analysis of the accuracy of our model. This is the aim of this subsection.

The planar Poiseuille flow is the steady flow between infinite parallel plates. The height of the channel is L_y and the flow is driven by a constant pressure gradient $\partial_x p$ (see Fig. 6.1). The symmetries of the numerical geometry reduce the incompressible Navier–Stokes equations to

$$-\partial_x p + \partial_y (\nu \partial_y u_x) = 0, \tag{6.1}$$

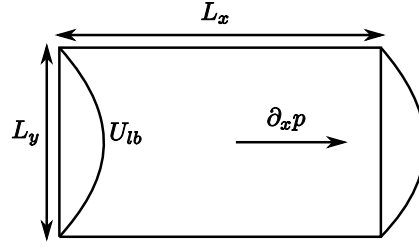


Fig. 6.1: The 2D planar Poiseuille, where U_{lb} is the maximum velocity of the flow.

where we remind the reader that ν depends on $\partial_y u_x$. Integrating this equation one gets

$$-\partial_x p y + C_1 + \nu \partial_y u_x = 0. \quad (6.2)$$

By symmetry we have $\partial_y u_x(L_y/2) = 0$ and therefore

$$C_1 = \partial_x p \frac{L_y}{2}. \quad (6.3)$$

In the case of the Carreau model described in Section 3.1 the kinematic viscosity is given by

$$\nu = (\nu_0 - \nu_\infty) \left(1 + (\lambda \partial_y u_x)^2\right)^{(n-1)/2} + \nu_\infty. \quad (6.4)$$

We are therefore left with

$$-\partial_x p \left(y - \frac{L_y}{2}\right) + \left((\nu_0 - \nu_\infty) \left(1 + (\lambda \partial_y u_x)^2\right)^{(n-1)/2} + \nu_\infty\right) \partial_y u_x = 0. \quad (6.5)$$

Although there exists no analytical solution for this equation it can be easily integrated numerically with high accuracy. We will note this solution \mathbf{u}_{num} . The velocity profile computed this way will be used as boundary conditions for our benchmark. According to this solution, the velocity field is imposed at inlet and outlet and is zero on the upper and bottom walls. We have two non-dimensional parameters for our simulation : the Reynolds number, Re , and the Carreau number, Cu , which are defined as

$$\text{Re} = \frac{U_{lb} L_y}{\nu_0}, \quad \text{Cu} = \lambda \frac{U_{lb}}{L_y}. \quad (6.6)$$

The Carreau number represents the ratio between the characteristic time of the Carreau model, λ and the characteristic time of the inertial effects of the flow. Therefore the highest the Cu , the more “non-Newtonian” the fluid

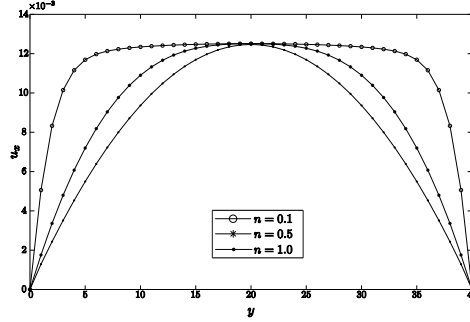


Fig. 6.2: The velocity profile of the planar Poiseuille for a fluid with a Carreau viscosity model for $n = 0.1, 0.5$ and for the Newtonian case $n = 1$.

is. We have tested the accuracy of the approach by using the same method as in Section 4.4. The size of the grid has been increased while keeping the kinematic viscosity, ν_0 constant and compared the solution with \mathbf{u}_{num} (the solution found by integrating numerically Eq. (6.5)). The comparison between the two profiles is done by computing the L^2 -error

$$E = \sqrt{\frac{1}{M} \sum_{k=0}^{M-1} \left\| \frac{\mathbf{u}_{\text{lb}}(\mathbf{x}_k)}{U_{\text{lb}}} - \mathbf{u}_{\text{num}}(\mathbf{x}_k) \right\|^2}, \quad (6.7)$$

where M is the total number nodes in our simulation. The simulations were performed for $\text{Re} = 1$, for $\text{Cu} = 1, 10, 100$ and $n = 0.1, 0.5$. The error with respect to the grid resolution is plotted in Fig. 6.3. As can be seen in Fig. 6.3

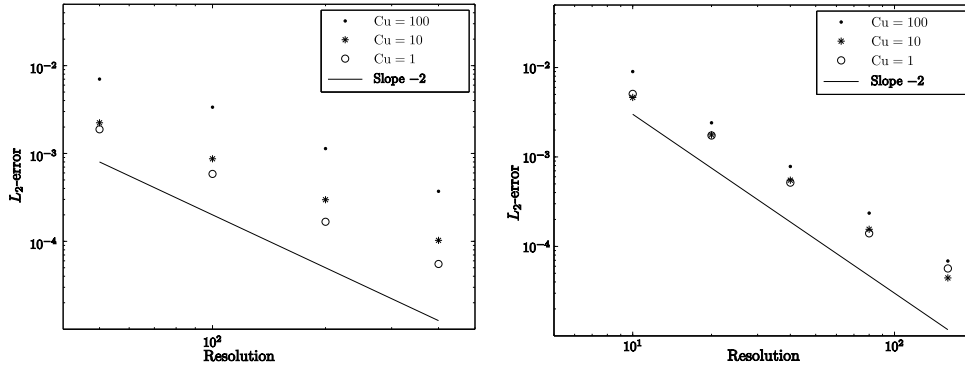


Fig. 6.3: The plot of the error with respect to the grid resolution in log-log scale, with $\text{Re} = 1$, $\text{Cu} = 1, 10, 100$, and $n = 0.1$ (left) and $n = 0.5$ (right).

for $n = 0.5$ second order accuracy for all Cu numbers is observed, whereas

for $n = 0.1$ the accuracy is second order for $\text{Cu} = 1$ but for $\text{Cu} = 10, 100$ the accuracy is respectively of order 1.8 and 1.7.

6.1.2 4:1 contraction

In this subsection, results related to the 4:1 axi-symmetric contraction as described in Fig. 6.4 are presented. The fully developed solution corresponding to the Poiseuille case is imposed at the inlet of width L_u and a zero gradient velocity is prescribed at the outlet. The sizes of the channel upstream and downstream of the contraction are noted L_u and L_d with $L_u/L_d = 4$. In our simulations we took that the length of the geometry, L_x , was given by $L_x = 2L_u$. We will study the non-dimensional reattachment length X

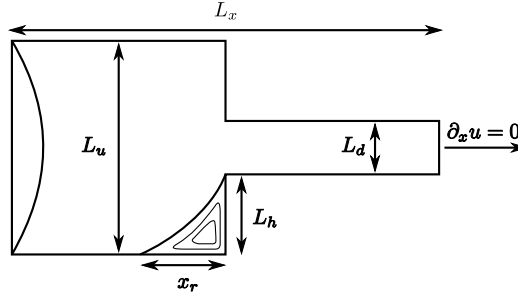


Fig. 6.4: The 2D planar 4:1 contraction.

$$X \equiv \frac{x_r}{L_h}, \quad (6.8)$$

where x_r is the length of the recirculation zone and L_h the height of the step (see Fig. 6.4). This reattachment length will be compared with results obtained with the commercial finite element method (FEM) code, COMSOL[®]. The simulations are performed at $\text{Re} = 1$, for $\text{Cu} = 1, 10$ and for n ranging from 0.1 to 0.9. As expected, the corner vortices are present and tend to disappear with a decreasing n and increasing Cu number (see Fig. 6.5). Furthermore a good agreement can be observed between the FEM solution and the lattice Boltzmann model for all values of n in the range $[0.1, 1]$.

6.2 Viscoelastic fluids benchmarks

In this section, the numerical scheme presented in Section 5.2 is tested. We will first use a periodic domain in order to test the scheme without boundary

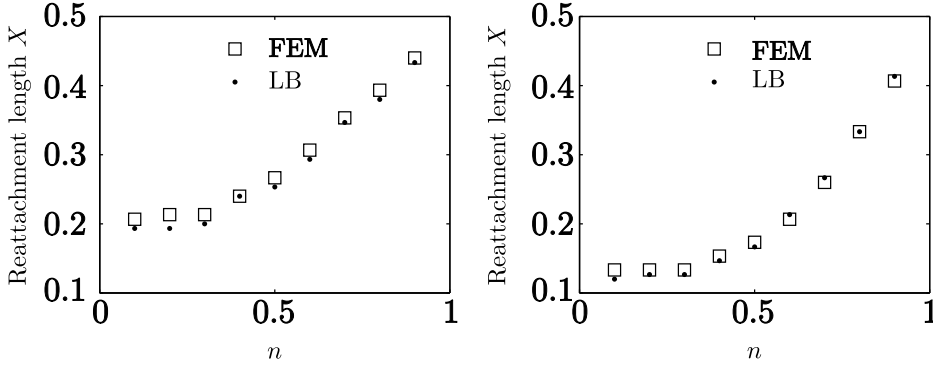


Fig. 6.5: The reattachment length, X , with respect to the power-law exponent, n , for $\text{Cu} = 1$ (left) and $\text{Cu} = 10$ (right).

conditions. There are two benchmarks (one in 2D and one in 3D) that are used for this purpose. Both are derived from the Taylor–Green vortex problem. We will then add boundaries to show that the approach for the boundary treatment discussed in Subsection 5.2.3 is correct. This test will be done on the 2D planar Poiseuille geometry. For these benchmarks we will fix the value of the diffusion constant of the advection–diffusion scheme relatively to the viscosity of the polymer such that

$$\frac{\kappa}{\mu_p} = 10^{-6}. \quad (6.9)$$

6.2.1 Taylor–Green vortex

In order to validate the numerical scheme discussed in Chapter 5, we will use the Taylor–Green vortex benchmark (see Brachet et al. [16]). The simulation setup is the following. The physical domain is a periodic box $[0, 2\pi] \times [0, 2\pi] \times [0, 2\pi]$, where we have a prescribed velocity profile at time $t = 0$ (see Fig. 6.6)

$$u_x(\mathbf{x}, t = 0) = -\frac{1}{\sqrt{3}} \sin(x) \cos(y) \cos(z), \quad (6.10)$$

$$u_y(\mathbf{x}, t = 0) = -\frac{1}{\sqrt{3}} \cos(x) \sin(y) \cos(z), \quad (6.11)$$

$$u_z(\mathbf{x}, t = 0) = \frac{2}{\sqrt{3}} \cos(x) \cos(y) \sin(z). \quad (6.12)$$

Initially, the viscoelastic stress tensor is taken equal to zero everywhere (equilibrium state). We will then let the time go forward and compare the energy

of the solvent and of the dumbbells with a high accuracy Fourier pseudospectral algorithm designed by Boeckle [10]. The kinetic and the spring energies are computed as

$$\mathcal{E}_{\text{kin}} = \frac{1}{M} \sum_k \frac{\mathbf{u}^2(\mathbf{x}_k)}{2}, \quad (6.13)$$

$$\mathcal{E}_{\text{spring}} = \frac{1}{M} \sum_k \frac{\text{tr}(\mathbf{\Pi}(\mathbf{x}_k))}{2}, \quad (6.14)$$

where k ranges over all the nodes of the simulation, and M is their total number. As in Boeckle [10], the benchmarks will be run until $t = 5$ dimensionless units for $\text{Re} = 1$, $\text{Wi} = 1, 5, 10$, and $\text{R}_\nu = 0.1, 0.7$.

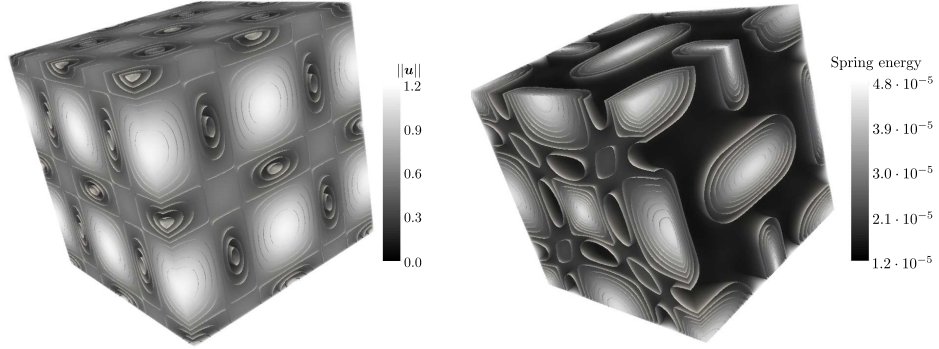


Fig. 6.6: The velocity norm at time $t = 0$ (left) and the spring energy at $t = 1.8$ and $\text{Wi} = 10$ (right) of the Taylor–Green vortex for $\theta = \pi/2$ with isocontours.

Before proceeding with the simulation, special care of the initial solution must be taken. As discussed in Mei et al. [75] or Latt [66] for a proper initialization of the problem the distribution functions, f_i must be given by their Chapman–Enskog expansion. For the incompressible Navier–Stokes model, it can be deduced from Eqs. (2.122) and (2.123) that

$$f_i = f_i^{(0)} + f_i^{(1)} = w_i \rho \left(1 + \frac{\xi_i \cdot \mathbf{u}}{c_l^2} + \frac{\mathcal{H}_i^{(2)} : \mathbf{u}\mathbf{u}}{2c_l^4} \right) - \frac{w_i \tau \rho}{c_l^2} \mathcal{H}_i^{(2)} : \mathbf{S}. \quad (6.15)$$

The velocity field and therefore the strain rate tensor are known. We are only left with the computation of the initial density field ρ . It is determined by taking the divergence of the momentum conservation equation. Using the

incompressibility condition, $\nabla \cdot \mathbf{u} = 0$, we find that the pressure field must be solution of a Poisson equation

$$\nabla^2 p = \nabla \mathbf{u} : \nabla \mathbf{u}. \quad (6.16)$$

In our case this equation has an analytical solution given by (see Appendix C)

$$\begin{aligned} p(\mathbf{x}, t = 0) = \frac{1}{6} & \left(\frac{1}{4} \cos(2y) + \frac{1}{4} \cos(-2y + 2x) + \frac{1}{4} \cos(2x + 2y) + \cos(2z) \right. \\ & + \frac{1}{24} \cos(-2z + 2y) + \frac{1}{24} \cos(2z + 2y) + \frac{1}{24} \cos(2z + 2x) \\ & \left. + \frac{1}{24} \cos(-2z + 2x) + \frac{1}{4} \cos(2x) - 1 \right). \end{aligned} \quad (6.17)$$

Then from the perfect gas law we get the initial density as

$$\rho = \frac{p}{c_l^2}.$$

For the simulation of the constitutive equation, we have that the $f_{i\alpha\beta}$'s must be equal, according to Eqs. (5.19) and (5.22), to

$$\begin{aligned} f_{i\alpha\beta} = w_i A_{\alpha\beta} & \left(1 + \frac{\boldsymbol{\xi}_i \cdot \mathbf{u}}{c_l^2} \right) \\ & - \frac{w_i \tau}{c_l^2} \boldsymbol{\xi}_i \cdot (A_{\alpha\beta} \partial_t \mathbf{u} - \mathbf{u} \nabla \cdot (A_{\alpha\beta} \mathbf{u}) + c_l^2 \nabla A_{\alpha\beta}), \end{aligned} \quad (6.18)$$

where \mathbf{u} is given by Eqs. (6.10)-(6.12). Since the viscoelastic tensor is null everywhere, we have that the components of the conformation tensor for the Oldroyd-B constitutive equation are

$$A_{\alpha\beta} = \delta_{\alpha\beta}. \quad (6.19)$$

Note also that the initial velocity field is incompressible and therefore the second term in Eq. (6.18) vanishes. We finally have

$$f_{i\alpha\beta} = w_i A_{\alpha\beta} \left(1 + \frac{\boldsymbol{\xi}_i \cdot \mathbf{u}}{c_l^2} \right). \quad (6.20)$$

As can be seen on Figs. 6.7 and 6.8 the energy curves for all Wi numbers tested coincide for $N = 100$.

Let us discuss briefly the simulated physical phenomenon. At start of the simulation the dumbbells are at equilibrium. Then due to the friction between

the solvent and the beads, the dumbbells start to stretch. When all the kinetic energy is transformed in spring potential energy, the velocity gradients are no longer strong enough to continue stretching the beads. As a consequence, the dumbbells start to give back their energy to the solvent. Then a peak of kinetic energy is reached which in turn coincides with a minimum of spring energy. This process continues until all the energy of the dumbbells and of the solvent has been dissipated through viscous processes. As can also be noticed from the plots, the higher the Wi number is the longer the period between two peaks can be observed. The lowest the value of R_ν , the highest the peaks of spring energy are. These effects can be easily understood, since a high Wi number means a high relaxation time of the dumbbells and a low value of R_ν means a high polymer viscosity, and thus a high coupling with the solvent. A simulation with the non-linear FENE-P model was performed for $Wi = 1$ and $R_\nu = 0.1$ with the maximum dumbbells length $r_e^2 = 10, 25$. As can be seen in Fig. 6.9 the energy curves have a shape similar to those obtained with the Oldroyd-B model. Furthermore, as expected, the FENE-P energy curves are approaching those of the Oldroyd-B model for $r_e \rightarrow \infty$. After testing the accuracy of the scheme by comparing it to the pseudo-spectral code, we have also performed a stability test. There, accuracy is not the main issue, but rather the numerical stability in high Wi numbers. We have fixed $R_\nu = 0.1$ for all tests. The simulations are stopped when the total energy falls below 0.1% of its initial value, which means $\mathcal{E} = \mathcal{E}_{kin} + \mathcal{E}_{spring} \sim 0.03$. All the tests were carried out with a resolution of $N = 25$ and up to $Wi = 1000$. The scheme was found numerically stable for all values of Wi .

6.2.2 The four-roll mill (simplified)

This test case consists in four cylinders that rotate to create an elongational flow in the vicinity of a stagnation point between the rollers. A sketch of the situation can be found in Fig. 6.10. For more detailed informations about this benchmark see Thomases and Shelley [95]. Here the rollers are replaced by a body force to drive the flow. This has as an effect to produce four vortices at the location of the rollers. Furthermore we take the boundary conditions to be periodic.

At the central hyperbolic point, the state of the fluid is described by a simple elongational flow, which enables us to compute analytically local solutions at steady state that will be compared with our simulations. The geometry is a 2D periodic box of size $[0, 2\pi] \times [0, 2\pi]$. The flow is supposed to be at very

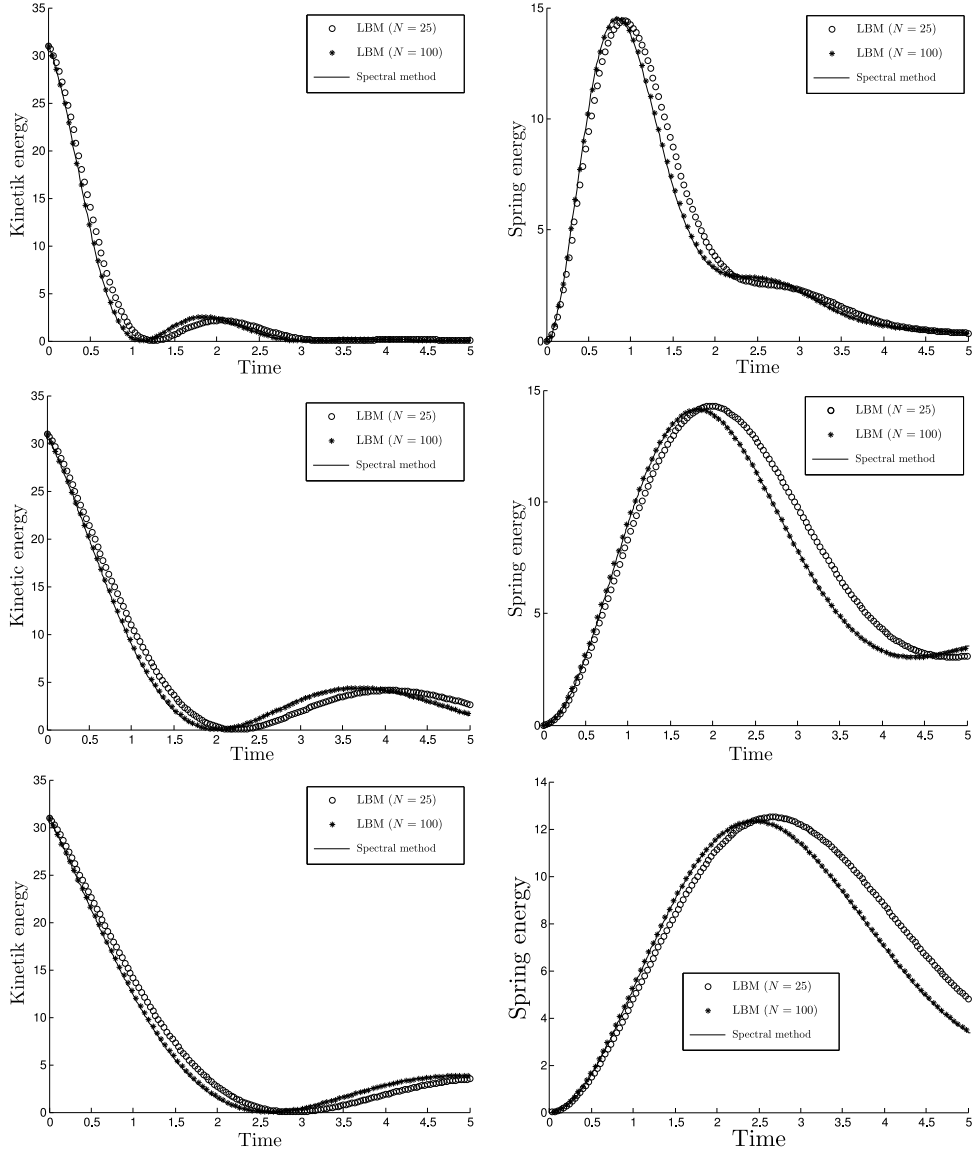


Fig. 6.7: Kinetic (left column) and spring (right column) energy evolution with time in non-dimensional units for $Re_\nu = 0.1$ and $Wi = 1$ (top row), $Wi = 5$ (middle row) and $Wi = 10$ (bottom row). The solid line is the reference solution while the circles and the dots are solutions of the LBM simulation for $N = 25$ and $N = 100$.

small Re number and is imposed by a body force

$$\mathbf{g} = (2\nu_s \sin(x) \cos(y), -2\nu_s \cos(x) \sin(y)), \quad (6.21)$$

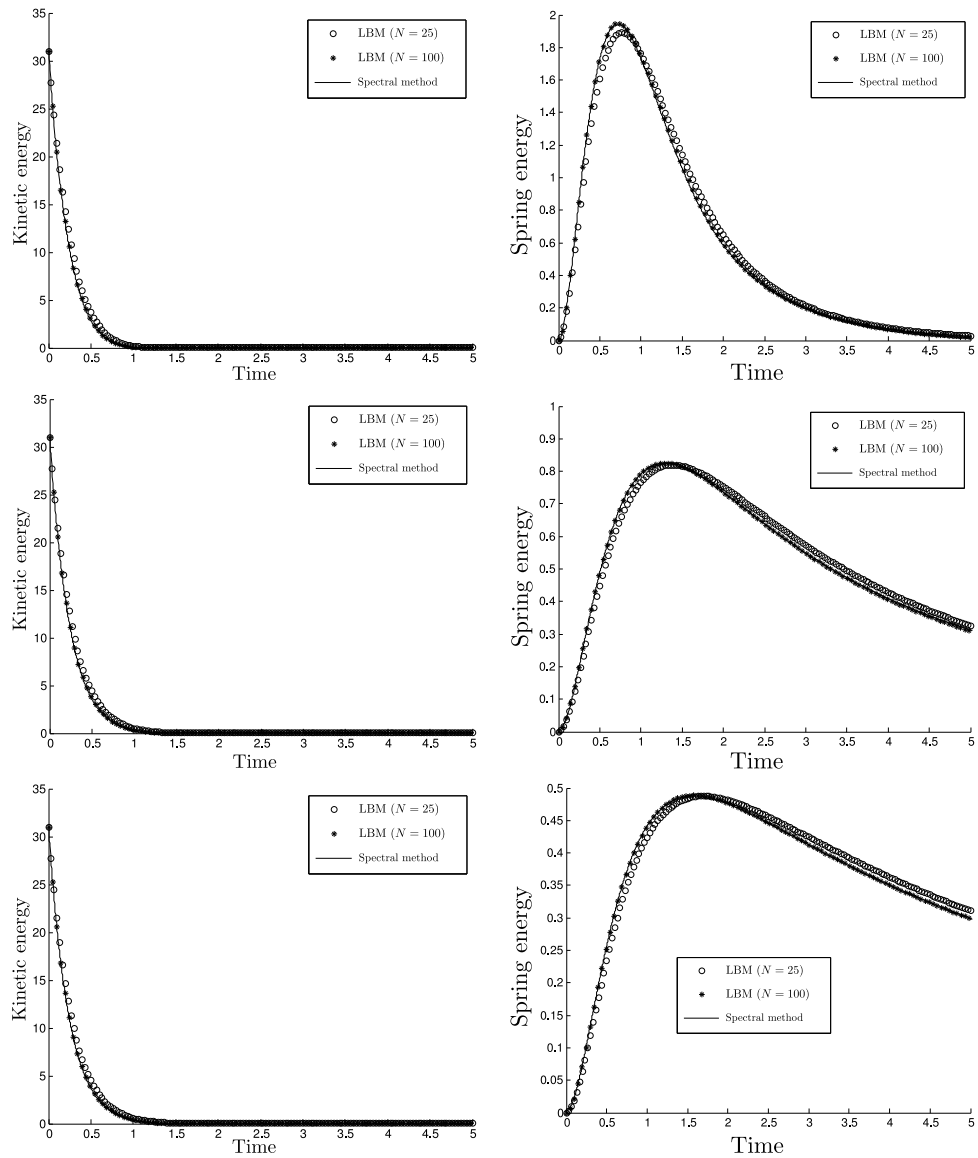


Fig. 6.8: Kinetic (left column) and spring (right column) energy evolution with time in non-dimensional units for $R_\nu = 0.7$ and $Wi = 1$ (top row), $Wi = 5$ (middle row) and $Wi = 10$ (bottom row). The solid line is the reference solutions while the circles and the dots are solution of the LBM simulation for $N = 25$ and $N = 100$.

computed in order to impose a velocity field given by

$$\mathbf{u} = (\sin(x) \cos(y), -\cos(x) \sin(y)). \quad (6.22)$$

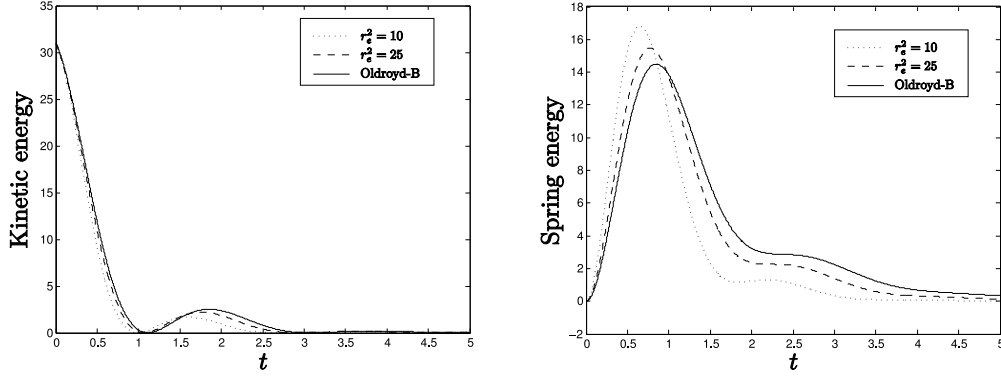


Fig. 6.9: Kinetic (left) and spring (right) energy evolution with time in non-dimensional units for $R_\nu = 0.1$ and $Wi = 1$. The solid line is the solution obtained with the Oldroyd-B model while dotted and dashed lines are the FENE-P model with $r_e^2 = 10$ and $r_e^2 = 25$.

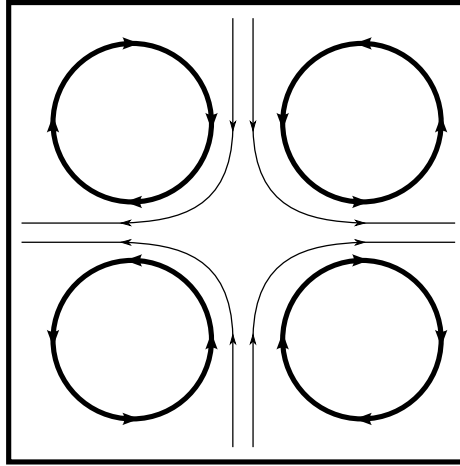


Fig. 6.10: The 2D projection of the four-roll mill.

The initial viscoelastic stress tensor is set to zero. In the central region, near the stagnation point ($x = \pi, y = \pi$), the velocity field is given by

$$\mathbf{u} = (\dot{\epsilon}x, \dot{\epsilon}y) = (\partial_x u_x(\pi, \pi), \partial_y u_y(\pi, \pi)), \quad (6.23)$$

where $\dot{\epsilon}$ is the elongational rate and the “effective” Weissenberg number, Wi^{eff} , as

$$Wi^{\text{eff}} = \dot{\epsilon}Wi, \quad (6.24)$$

with $Wi = \lambda U_{\text{max}}/L$ (with U_{max} the maximum velocity of the flow and

$L = 2\pi$). When the simulation reaches steady state ($t \rightarrow \infty$), according to Thomases and Shelley [95] the components of the conformation tensor can be expressed in a good approximation with

$$A_{xx}(\pi, y) = \frac{1}{1 - 2Wi^{\text{eff}}} + C|y|^{\frac{(1-2Wi^{\text{eff}})}{Wi^{\text{eff}}}}, \quad (6.25)$$

$$A_{yy}(\pi, y) = \frac{1}{1 + 2Wi^{\text{eff}}} + |y|^{(2+1/Wi^{\text{eff}})}, \quad (6.26)$$

$$A_{xy}(\pi, y) = 0. \quad (6.27)$$

where C is a constant. These values will be compared with the results of our simulations. As an illustration, in Fig. 6.11 the vorticity and the spring energy at time $t = 10$ are depicted. For our simulations we initialize the ve-

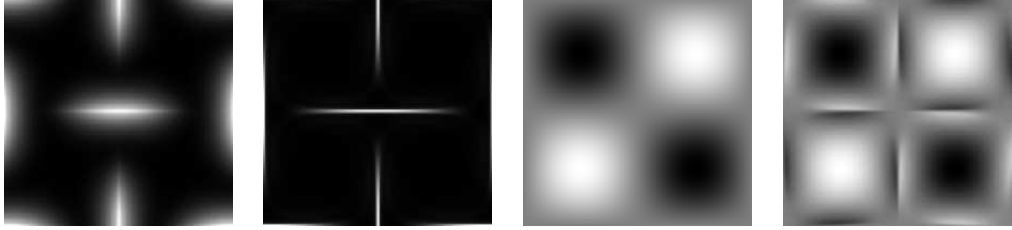


Fig. 6.11: The vorticity for $Wi = 0.6$ and $Wi = 5$ (middle and far right) and the spring energy for $Wi = 0.6$ and $Wi = 5$ (far and middle left) at time $t = 10$. The colormap is white for high and black for low values.

locity field with Eqs. (6.22) and density $\rho = 1$. The body force of Eq. (6.21) is applied on the domain. Before coupling the system with the viscoelastic fluid, we first let the Navier–Stokes scheme converge, which means that we start with a Newtonian velocity field. Once this step is completed we start the real simulation by coupling the fluid with the Oldroyd-B constitutive equation solver. The values of the conformation tensor and of the elongational rate $\dot{\epsilon}$ are then taken when the simulation has reached steady state, which happens between time $t = 10$ and $t = 15$ depending on the Wi number. From Fig. 6.12 we see that $\dot{\epsilon}$ is decreasing with increasing Wi . This result seems qualitatively very close to the results found in Thomases and Shelley [95]. The Figs. 6.13-6.14 show a very good agreement between our simulations and the analytical results of Eqs. (6.25)-(6.27) for the conformation tensor values. Nevertheless, one can notice that the component A_{xy} of Fig. 6.14, at values of Wi^{eff} close to one start to deviate from zero. This “discrepancy” should be compared with the characteristic value of A_{xy} at the given Wi^{eff} which is of several orders of magnitude higher.

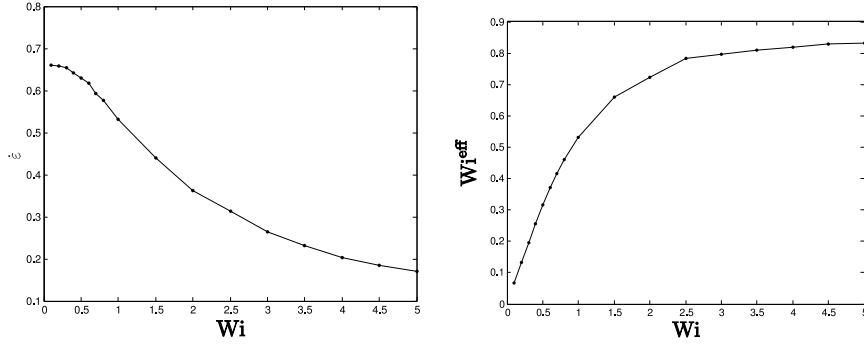


Fig. 6.12: The elongational rate $\dot{\epsilon}$ with respect to the Wi number (left) and the effective Weissenberg number Wi^{eff} with respect to the Wi number.

6.2.3 Steady planar Poiseuille flow

In order to test our numerical scheme on domains with boundaries, we have selected the Poiseuille geometry. The geometry, already discussed in Subsection 6.1.1, is depicted in Fig. 6.1. In order to avoid corners in the geometry, the flow is imposed by a body force, F_x , and the domain is made periodic in the x -direction. The top and bottom walls have no-slip boundary conditions. There exists an exact solution for the Oldroyd-B constitutive equation which can be found for example in the thesis of Fiétier [33] and which is given by

$$u_x = -\frac{F_x}{2} \left(\frac{y^2 - L_y y}{\nu_s + \nu_p} \right), \quad u_y = 0, \quad (6.28)$$

$$A_{xx} = 1 + \frac{F_x^2 \lambda^2 \nu_p (-2y + L_y)^2}{2(\nu_s + \nu_p)^2}, \quad (6.29)$$

$$A_{xy} = -\frac{F_x \lambda (2y - L_y)}{2(\nu_s + \nu_p)}, \quad (6.30)$$

$$A_{yy} = 1. \quad (6.31)$$

Defining the velocity of reference as the maximum velocity of the flow as U_{lb} , which by symmetry is located at $y = L_y/2$, we have that the force is given by

$$F_x = \frac{4}{L_y^2} (\nu_s + \nu_p) U_{\text{lb}}. \quad (6.32)$$

In Figs. 6.15-6.17, we plotted the velocity field and the components of the conformation tensor obtained from the numerical simulations for $Wi = 0.1, 1$ and from the analytical solution. As in Subsection 6.1.1 we also compute the error between the lattice Boltzmann solution and reference solution for

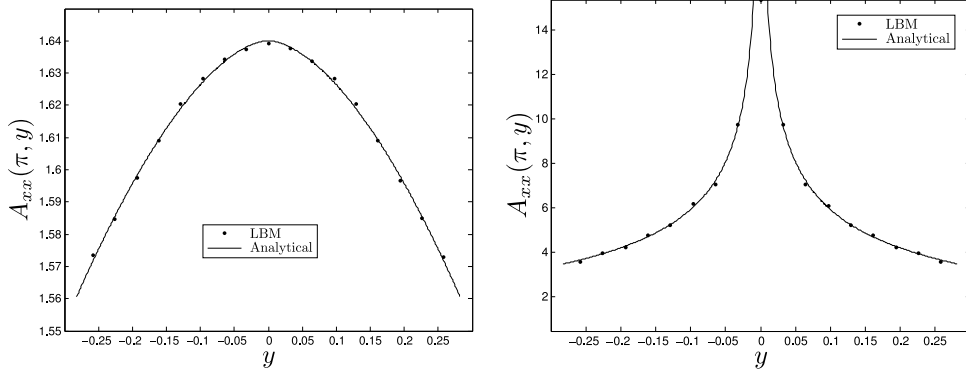


Fig. 6.13: The A_{xx} components in function of the y position for $x = \pi$ at $Wi = 0.3$ (left) and $Wi = 1.0$ (right). The dots are the results of the LBM simulation, while the plain line is the analytical result of Eq. (6.25), where we fitted the constant coefficient C .

this flow and measure its accuracy order. However this time we will not only measure the order of accuracy for the velocity field but also for the conformation tensor. The errors are defined as

$$E_u = \sqrt{\frac{1}{M} \sum_{k=0}^{M-1} \left\| \frac{\mathbf{u}_{lb}(\mathbf{x}_k)}{U_{lb}} - \mathbf{u}_a(\mathbf{x}_k) \right\|^2}, \quad (6.33)$$

$$E_{A_{\alpha\beta}} = \sqrt{\frac{1}{M} \sum_{k=0}^{M-1} \left\| A_{lb}(\mathbf{x}_k)_{\alpha\beta} - A_a(\mathbf{x}_k)_{\alpha\beta} \right\|^2}, \quad (6.34)$$

and again the sum is performed on the M nodes of the lattice, which are located on \mathbf{x}_k . The simulations were run at $Re = 1$, $R_\nu = 0.1, 0.7$ and $Wi = 0.1, 1$. We have also tested the numerical stability of the code. In this case the resolution of the lattice was fixed ($N_x = N_y = 25$), the viscosity ratio was varied in the range $R_\nu = [0.1, 0.9]$ and the maximum stable Wi was measured.

As can be seen in Figs. 6.18 and 6.19, the error of the velocity field, in the $Wi = 0.1$ and $Wi = 1$ cases, is decreasing with order two. The errors are of the same order magnitude for $R_\nu = 0.1$ or $R_\nu = 0.7$. For the conformation tensor, it can be observed that the errors on the components A_{xx} and A_{xy} are decreasing more slowly, in fact the order is of 1.5 and 1.4 for respectively $Wi = 0.1$ and $Wi = 1$. For the A_{yy} component the accuracy seems to have reached its minimum value for all the lattice resolutions since it does not really decrease (or increase) when the resolution is increased.

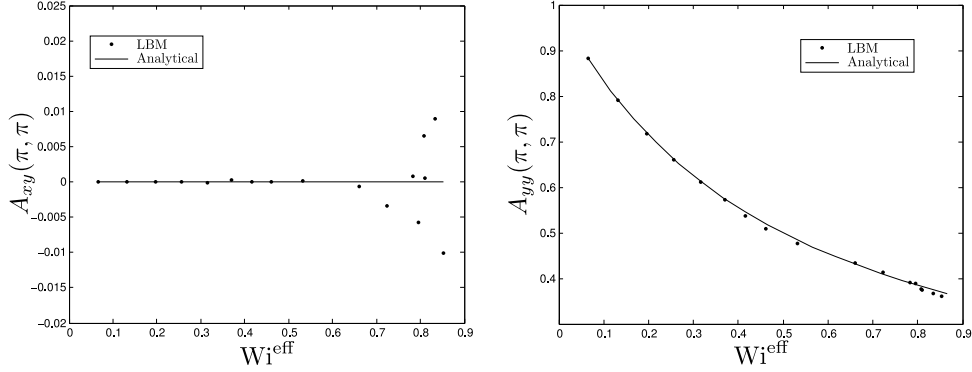


Fig. 6.14: The A_{xy} (left) and A_{yy} (right) components in function of the effective Weissenberg number Wi^{eff} at the stagnation point. The dots are the LBM measures while the plain line is the analytical solution of Eqs. (6.27) and (6.26).

For the stability test, we found that the maximum Wi number that was simulated was related to the viscosity ratio, R_ν , ranging from $Wi_{\text{max}} = 1$ for $R_\nu = 0.1$ to $Wi = 7$ for $R_\nu = 0.6$. For the chosen resolution, the Wi number could not be increased higher than seven for any R_ν . All these tests were performed for $\kappa_p/\mu_p \sim 10^{-6}$. Discussion of possible reasons for the failure of the simulations at higher values of Wi are postponed till Section 6.4.

6.3 Parallelization of the viscoelastic model

The lattice Boltzmann method is known for having a very good scalability on a parallel computer. The parallel implementation, which is the one of Palabos [87], is a so-called multiblock approach (see Heuveline and Latt [51]). In this section we will use the Taylor–Green benchmark presented in Subsection 6.2.1 to test the computational efficiency of our viscoelastic model.

A reference quantity is the Mega Sites Updates per Second (MSUPS) which is, in millions, the number of collision-propagation steps on a single cell done per second in a simulation and which is a common measure for lattice Boltzmann simulations. For comparison we will also show the results for the incompressible Newtonian case of the Taylor–Green vortex. The cluster, Pleiades 2, that has been used for this study is composed of 210 Intel Woodcrest 5150 (quadcore processors) nodes with 8Gb of memory per node. The

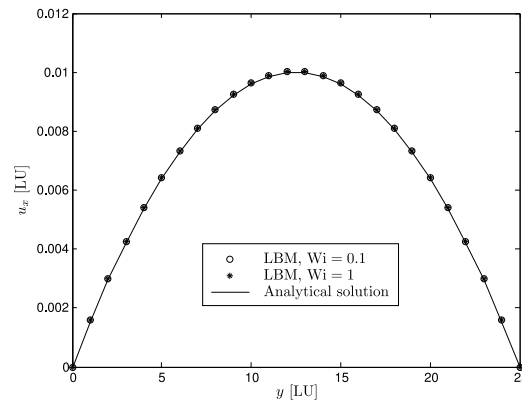


Fig. 6.15: The velocity field with respect to the y position in lattice units for $Wi = 0.1, 1$, $R_\nu = 0.1$, and $Re = 1$ for the LBM simulation and the analytical solution.

corresponding network is a SWITCH Black diamond 8810 432Gb/s.

As can be seen from Fig. 6.20, the MSUPS in the Newtonian case is about an order of magnitude higher than for the viscoelastic one. This result is not really surprising, since there are more than three times more distribution functions on each node in the viscoelastic case and additional computations must be done on each node for the viscoelastic model to incorporate the force term and the source term for the Navier–Stokes and the advection–diffusion schemes respectively (see Eqs. (2.126) and (5.24)). It can be inferred from Fig. 6.20 that the parallelization efficiency for the Newtonian case is better than in the viscoelastic one. This is due to the higher amount of information that must be transmitted across the network. In the Newtonian case only 19 distribution functions must be communicated while 61 distribution functions and twelve additional scalars (three for the force, three for the velocity field and six for the source terms) have to be communicated for the viscoelastic case, which makes a total of 73 communicated values. A typical time for the simulation of the viscoelastic case, for $N = 100$, and $\Delta t = 1.33 \cdot 10^{-4}$ on four processors for five time units (about 37'000 iterations) is of about three hours.

It must also be noted that the parallelization efficiency on the Pleiades 2 cluster is much lower than the current “state-of-the-art”, because of the relatively slow interconnexion between the nodes. A benchmark on the 3D lid-driven cavity of domain size of 4013^3 , conducted by J. Latt, on thousands

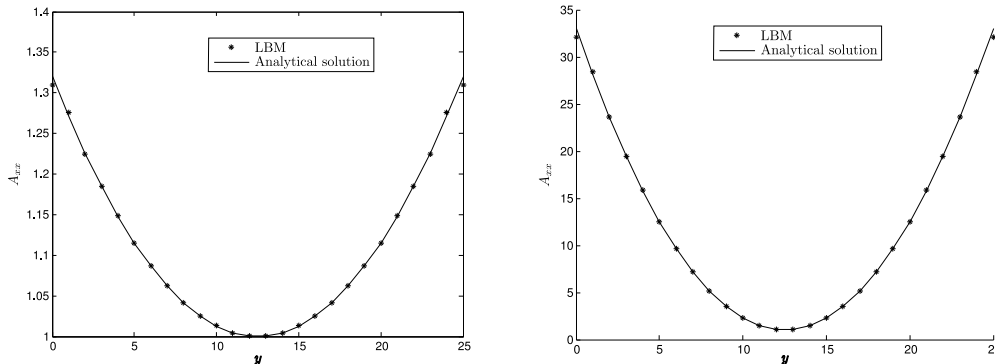


Fig. 6.16: The A_{xx} component of the conformation tensor with respect to the y position in lattice units and the analytical solution for $R_\nu = 0.1$, $Wi = 0.1$ (left) and $Wi = 1$ (right).

of cores of a Blue Gene/L machine using the Palabos library shows a quasi linear scalability for thousands of cores (see Fig. 6.21). The viscoelastic flow could also benefit from this computer architecture.

6.4 Discussion

In this chapter we validated the numerical schemes for generalized Newtonian fluids and the Oldroyd-B class of viscoelastic fluids.

The generalized Newtonian scheme was compared to a semi-analytical solution for the Poiseuille flow. We have shown that its accuracy was between order 1.7 and 2 depending on the parameters of the simulation. Especially the more non-Newtonian the less accurate the scheme proved to be. We have also computed the reattachment length in the case of the planar abrupt 4:1 contraction, and showed that we were in good agreement with a FEM commercial solver.

For the viscoelastic scheme we were able to compare the scheme with a high accuracy Fourier pseudospectral algorithm for the Taylor–Green vortex case and found identical results for the energy decay of the solvent and of the polymers, using the Oldroyd-B constitutive equation. Furthermore the code was shown to be unconditionally stable, even at very high (unphysical) Wi numbers. We also showed that in principle the FENE-P model could also be implemented with our model.

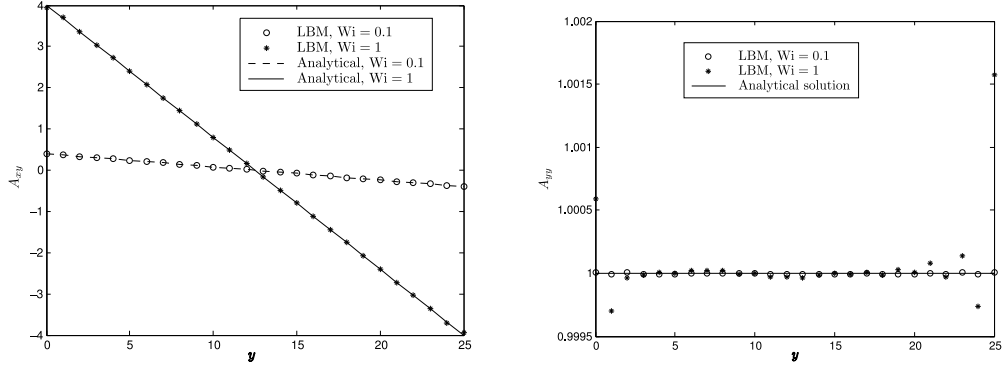


Fig. 6.17: The A_{xy} (left) and A_{yy} (right) components of the conformation tensor with respect to the y position in lattice units and the analytical solutions for $R_\nu = 0.1$ and $Wi = 0.1, 1$.

In the simplified four-roll mill we were able to reproduce accurately the analytical results for an elongational flow that were obtained by Thomases and Shelley [95] in the steady state limit.

In the Poiseuille flow case we showed that the numerical result was compatible with the analytical solution and therefore the boundary condition treatment for flat walls seems to be correct. Unfortunately we were unable to determine boundary conditions for more complicated geometries, including corners, (the simulation of the 4:1 contraction were always unstable even at low Wi number). In order to improve this defect one should make a massive effort in the modelling of the boundaries for such flows. The stability of the code did not allow us to reach a Wi number higher than one for $R_\nu = 0.1$. Although results were better with higher R_ν (up to Wi of seven were simulated), one of the major issues here is the lack of understanding of the behavior of the viscoelastic stress tensor at the boundaries. The tests performed by applying the analytical solution for the conformation tensor on the boundaries allowed to simulate flows with much higher Wi numbers. It must also be noted that increasing the “polymer diffusion” κ_p also increased the stability of the model. This is simply because the relaxation time of the advection–diffusion scheme departs from $1/2$ as κ_p is increased¹.

A possible solution for the boundary conditions problem would be to couple the LBM solver with a molecular dynamics solver (see Hernández-Ortiz

¹The limit $\tau \rightarrow 1/2$ is well known stability issue for the lattice Boltzmann.

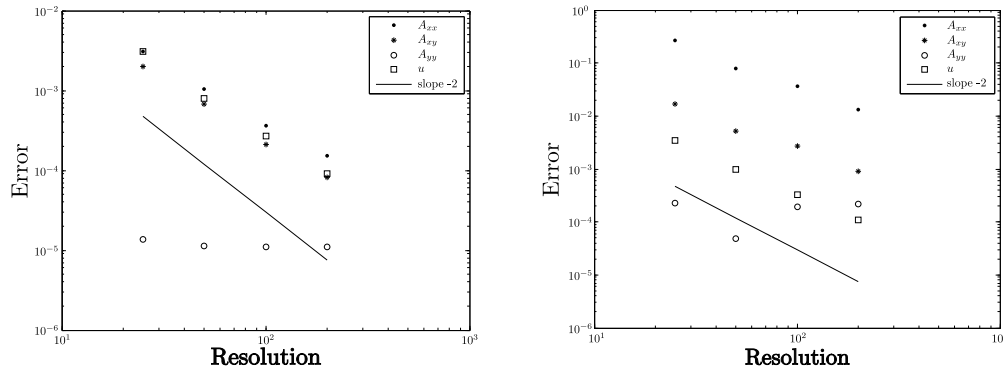


Fig. 6.18: The L_2 -error of the conformation tensor and of the velocity with respect to the resolution in lattice units for $\text{Re} = 1$, $\text{R}_\nu = 0.1$ and $\text{Wi} = 0.1$ (left) and $\text{Wi} = 1$ (right).

et al. [50] and Izmitli et al. [56]). Having a physically relevant value for the conformation tensor on the boundaries would allow us to impose a physically meaningful boundary condition for the viscoelastic constitutive scheme. Thus there would be no need for use of interpolated or extrapolated values that are mainly responsible for the lack of stability of our code.

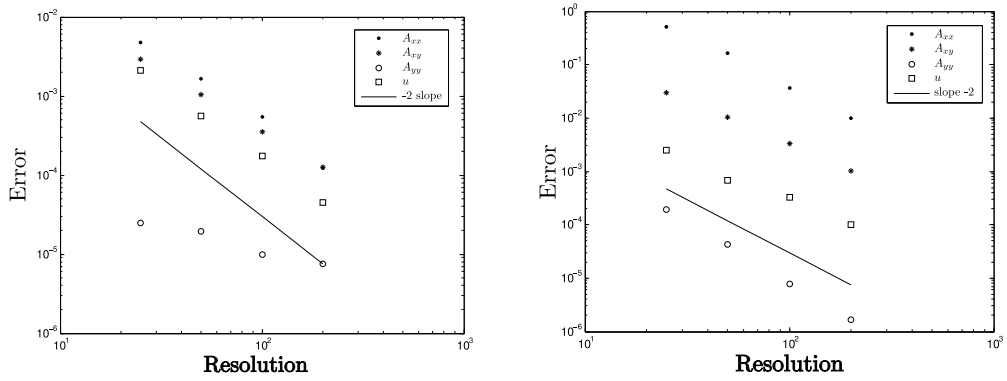


Fig. 6.19: The L_2 -error of the conformation tensor and of the velocity with respect to the resolution in lattice units for $Re = 1$, $R_\nu = 0.7$ and $Wi = 0.1$ (left) and $Wi = 1$ (right).

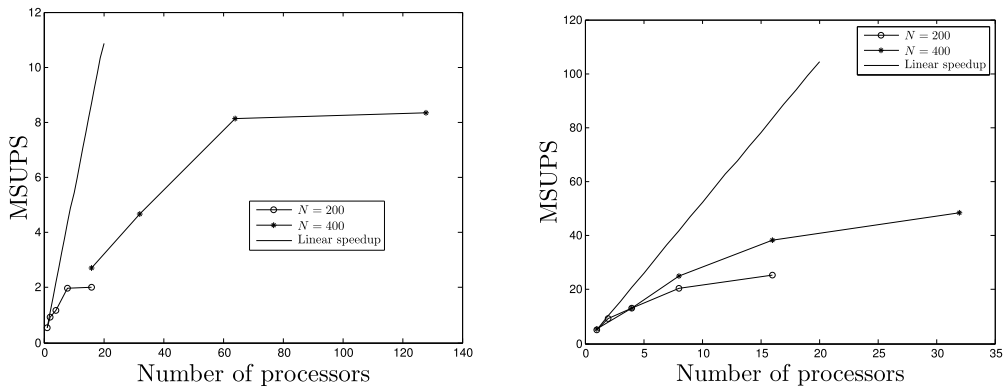


Fig. 6.20: The Mega Sites Updates per Second (MSUPS) with respect to the processor number for the viscoelastic (left) and Newtonian (right) cases, on the Taylor–Green benchmark for $Re = 1$, $R_\nu = 0.1$ and $Wi = 1$.

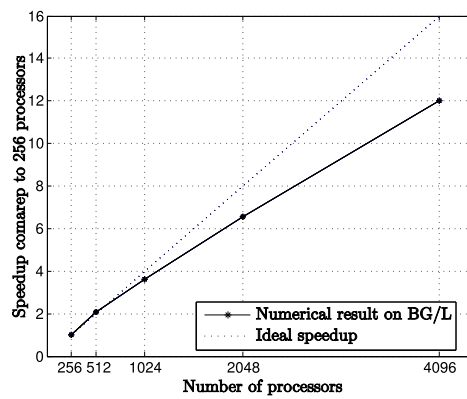


Fig. 6.21: The speedup for case of the 3D lid-driven cavity for a domain of 4013^3 nodes on a Blue Gene/L machine.

Chapter 7

Conclusions and perspectives

In this thesis we have first presented a modern approach of deriving the lattice Boltzmann method directly from the continuous BGK Boltzmann equation. In addition, we have shown how to incorporate an external force in a consistent manner. We also have presented three different discretizations that allow to simulate Newtonian fluids in different limits : the weakly compressible, the compressible isothermal, and the compressible thermal cases. This discretization which is also compatible with the 13-Grad's moments equations, offers in principle the possibility to simulate high Knudsen number flows and opens therefore new possibilities for the applications of the LBM. These more complex fluid simulations are theoretically possible, but there remains a step before being able to perform them since the boundary conditions problem is not solved yet.

The boundary conditions are a recurrent issue for the lattice Boltzmann method and thus we have presented a review and analysis of some of the more popular ones for the weakly compressible fluids case. It has been observed that in order to be satisfactory, that is, accurate and numerically stable, these boundary conditions should not only recover the correct density and velocity on the walls, but also the correct deviatoric stress tensor. The benchmarks performed on the dipole-wall collision show this fact clearly, since only the boundary conditions with a correct deviatoric stress were able to reproduce the physics of the problem.

The lattice Boltzmann scheme was then applied to tackle non-Newtonian fluid flows. We first presented an algorithm to simulate generalized Newtonian fluids and applied it to the Carreau model, and showed that in the case of the Poiseuille flow, the method was second order accurate in space. We also showed that there is good agreement between the lattice Boltzmann method and the commercial finite element solver COMSOL for the 4:1 contraction benchmark. We made full use of the properties of the lattice Boltzmann method, since the strain rate tensor was computed locally without resorting to any finite difference scheme.

One of the novelties of this thesis was the simulation of viscoelastic fluids which exhibit memory effects, using the Oldroyd-B and FENE-P models.

Our methodology proved successful in solving the Taylor–Green decay vortex case, the steady simplified 2D four-roll mill and the 2D channel flow. In the Taylor–Green vortex benchmark we have reproduced accurately the results obtained with a high accuracy spectral method for Weissenberg (Wi) numbers that were ranging from one to ten. At higher Wi numbers the simulation was shown to be also stable and the overall shape of the energy curves remains the same, but the simulation time becomes much longer since the “bounces” observed in Figs. 6.7-6.8 take longer time as Wi increases. The analytical results for the 2D four-roll mill case were reproduced accurately. We also have presented a method for taking into account flat wall boundaries in our simulations, which represents a first step towards the study of more complex geometries. We have been able to compare our simulation results with the analytical solution given for a 2D Poiseuille flow and showed that at moderate Wi numbers the accuracy was second order in space, while it was decreased to order 1.5 for $Wi = 1$. In agreement with results of other methods, we also observed that the stability of our model strongly depends on the viscosity ratio. The advection–diffusion with source model that was used for performing all these simulations with the Oldroyd-B model can in principle describe any constitutive equation which has the same form. This allowed to implement the FENE-P model and to apply it on the 3D Taylor–Green vortex decay.

Our tool could now be applied to more challenging cases like the tracking of elastic instabilities for shear flows, as the Kolmogorov flow (see Berti et al. [6]) or the elongational flows (see Poole et al. [84] and Arratia et al. [5]). Other more sophisticated constitutive equations, which would contain more physical ingredients and therefore reproduce better the behavior of the viscoelastic fluids, could also be easily implemented.

The next issue to be investigated is to obtain a better understanding of the boundaries in order to be able to use the ability of the lattice Boltzmann to easily simulate complex geometries. To this effect, it could be interesting to try to couple molecular dynamics models with the lattice Boltzmann method, since in this case it is easier to introduce more refined mesoscopic modellings (see Hernández-Ortiz et al. [50] and Izmitli et al. [56] for example).

Appendix A

Properties of the Hermite polynomials

In this appendix we will discuss different properties of the Hermite polynomials. The results that are presented here are a summary of the work by Grad [39]. We will first give relations in the continuous case and then in the discretized one.

Continuous case

The Rodrigues' formula gives the Hermite polynomial of degree n in a D dimensional space

$$\mathcal{H}^{(n)} = \frac{(-1)^n}{\omega(\boldsymbol{\xi})} \partial_{\xi_\alpha}^n \omega(\boldsymbol{\xi}), \quad (\text{A-1})$$

and the definition of the weight function associated with the Hermite polynomials $\omega(\boldsymbol{\xi})$

$$\omega(\boldsymbol{\xi}) = \frac{1}{(2\pi)^{D/2}} \exp(-\xi^2/2). \quad (\text{A-2})$$

For our purpose we will need the Hermite polynomials up to order four. With these formulas one can easily compute them

$$\mathcal{H}^{(0)} = 1, \quad (\text{A-3})$$

$$\mathcal{H}_\alpha^{(1)} = \xi_\alpha, \quad (\text{A-4})$$

$$\mathcal{H}_{\alpha\beta}^{(2)} = \xi_\alpha \xi_\beta - \delta_{\alpha\beta}, \quad (\text{A-5})$$

$$\mathcal{H}_{\alpha\beta\gamma}^{(3)} = \xi_\alpha \xi_\beta \xi_\gamma - (\delta_{\alpha\beta} \xi_\gamma + \delta_{\alpha\gamma} \xi_\beta + \delta_{\beta\gamma} \xi_\alpha), \quad (\text{A-6})$$

$$\begin{aligned} \mathcal{H}_{\alpha\beta\gamma\delta}^{(4)} = & \xi_\alpha \xi_\beta \xi_\gamma \xi_\delta - (\delta_{\alpha\beta} \xi_\gamma \xi_\delta + \delta_{\alpha\gamma} \xi_\beta \xi_\delta + \delta_{\alpha\delta} \xi_\beta \xi_\gamma \\ & + \delta_{\beta\gamma} \xi_\alpha \xi_\delta + \delta_{\beta\delta} \xi_\alpha \xi_\gamma + \delta_{\gamma\delta} \xi_\alpha \xi_\beta) \\ & + (\delta_{\alpha\beta} \delta_{\gamma\delta} + \delta_{\alpha\gamma} \delta_{\beta\delta} + \delta_{\alpha\delta} \delta_{\beta\gamma}). \end{aligned} \quad (\text{A-7})$$

There also exists a recurrence relation between the polynomials

$$\xi_{\alpha_0} \mathcal{H}_{\alpha_1 \dots \alpha_n}^{(n)} = \mathcal{H}_{\alpha_0 \dots \alpha_n}^{(n+1)} + \sum_{i=1}^n \delta_{\alpha_0 \alpha_i} \mathcal{H}_{\alpha_1 \dots \alpha_{i-1} \alpha_{i+1} \dots \alpha_n}^{(n-1)}. \quad (\text{A-8})$$

Using Eq. (A-2) one can also rewrite the expression $\nabla_{\xi} (\omega \mathcal{H}^{(n)})$

$$\nabla_{\xi} (\omega \mathcal{H}^{(n)}) = (-1)^n \nabla_{\xi}^n \omega = -\omega \mathcal{H}^{(n-1)}. \quad (\text{A-9})$$

Discrete case

When discretized using the Gauss–Hermite quadrature, the ξ_i must be rescaled in order to recover the standard formulation of the lattice Boltzmann method. This rescaling has as an effect to modify the discrete Hermite polynomials in the following way

$$\mathcal{H}_i^{(0)} = 1, \quad (\text{A-10})$$

$$\mathcal{H}_{i\alpha}^{(1)} = \xi_{i\alpha}, \quad (\text{A-11})$$

$$\mathcal{H}_{i\alpha\beta}^{(2)} = \xi_{i\alpha} \xi_{i\beta} - c_l^2 \delta_{\alpha\beta}, \quad (\text{A-12})$$

$$\mathcal{H}_{i\alpha\beta\gamma}^{(3)} = \xi_{i\alpha} \xi_{i\beta} \xi_{i\gamma} - c_l^2 (\delta_{\alpha\beta} \xi_{i\gamma} + \delta_{\alpha\gamma} \xi_{i\beta} + \delta_{\beta\gamma} \xi_{i\alpha}), \quad (\text{A-13})$$

$$\begin{aligned} \mathcal{H}_{i\alpha\beta\gamma\delta}^{(4)} &= \xi_{i\alpha} \xi_{i\beta} \xi_{i\gamma} \xi_{i\delta} - c_l^2 (\delta_{\alpha\beta} \xi_{i\gamma} \xi_{i\delta} + \delta_{\alpha\gamma} \xi_{i\beta} \xi_{i\delta} + \delta_{\alpha\delta} \xi_{i\beta} \xi_{i\gamma} \\ &\quad + \delta_{\beta\gamma} \xi_{i\alpha} \xi_{i\delta} + \delta_{\beta\delta} \xi_{i\alpha} \xi_{i\gamma} + \delta_{\gamma\delta} \xi_{i\alpha} \xi_{i\beta}) \\ &\quad + c_l^4 (\delta_{\alpha\beta} \delta_{\gamma\delta} + \delta_{\alpha\gamma} \delta_{\beta\delta} + \delta_{\alpha\delta} \delta_{\beta\gamma}). \end{aligned} \quad (\text{A-14})$$

The orthogonality of the basis vectors expresses now

$$\sum_i w_i \mathcal{H}_{\alpha_1 \dots \alpha_m}^{(m)} \mathcal{H}_{\beta_1 \dots \beta_n}^{(n)} = \delta_{mn} c_l^{2m} (\delta_{\alpha_1 \beta_1} \dots \delta_{\alpha_m \beta_m} + \text{perm}), \quad (\text{A-15})$$

where “perm” yields all the α_i permutations in the first index and β_i permutations in the second index of the $\delta_{\alpha_i \beta_i}$. In particular, we have for the

non-vanishing products

$$\sum_i w_i \mathcal{H}_i^{(0)} \mathcal{H}_i^{(0)} = 1, \quad (\text{A-16})$$

$$\sum_i w_i \mathcal{H}_{i\alpha}^{(1)} \mathcal{H}_{i\beta}^{(1)} = c_l^2 \delta_{\alpha\beta}, \quad (\text{A-17})$$

$$\sum_i w_i \mathcal{H}_{i\alpha\beta}^{(2)} \mathcal{H}_{i\gamma\delta}^{(2)} = c_l^4 (\delta_{\alpha\gamma} \delta_{\beta\delta} + \delta_{\alpha\delta} \delta_{\beta\gamma}), \quad (\text{A-18})$$

$$\begin{aligned} \sum_i w_i \mathcal{H}_{i\alpha\beta\gamma}^{(3)} \mathcal{H}_{i\delta\epsilon\zeta}^{(3)} &= c_l^6 (\delta_{\alpha\delta} (\delta_{\beta\epsilon} \delta_{\gamma\zeta} + \delta_{\beta\zeta} \delta_{\gamma\epsilon}) \\ &\quad + \delta_{\alpha\epsilon} (\delta_{\beta\delta} \delta_{\gamma\zeta} + \delta_{\beta\zeta} \delta_{\gamma\delta}) \\ &\quad + \delta_{\alpha\zeta} (\delta_{\beta\delta} \delta_{\gamma\epsilon} + \delta_{\beta\epsilon} \delta_{\gamma\delta})). \end{aligned} \quad (\text{A-19})$$

Finally the expansion in Hermite polynomials of a quantity f_i reads

$$f_i = w_i \sum_{n=0}^{\infty} \frac{1}{c_l^{2n} n!} \mathcal{H}_i^{(n)} : a^{(n)}, \quad (\text{A-20})$$

where $a^{(n)} = \sum_i w_i \mathcal{H}_i^{(n)} f_i$ is the expansion coefficient of order n .

Appendix B

Gauss–Hermite quadrature formulae

This Appendix is a short introduction on Gaussian quadrature and summarizes the important results that are used in the lattice Boltzmann method.

For a given function $p(\xi)$, Gaussian quadrature seeks to obtain the best estimate of the integral $\int_a^b \omega(\xi)p(\xi) d\xi$ by choosing the optimal set of abscissae $\xi_i, i = 1, \dots, n$ such that

$$\int_a^b \omega(\xi)p(\xi)d\xi \simeq \sum_{i=1}^n w_i p(\xi_i),$$

where $w_i, i = 1, \dots, n$ is a set of constant weights.

A quadrature rule is said to have an algebraic degree of precision m if for any p that is a polynomial of a degree up to m , exact equality holds in the above equation.

We copy here the Gauss–Hermite quadrature weights and abscissae for algebraic degree up to degree 9 for regular lattices (for details on the derivation see the papers by Shan et al. [92] and by Nie et al. [76]). The subscript FS represents a fully symmetric set of velocities. The $E_{D,m}^Q$ quadrature rule is of algebraic degree m , in D dimensions with Q abscissae. We find here the standard lattices for weakly compressible fluids $D2Q9$, $D3Q15$, $D319$ and $D3Q27$, which are nearest neighbors lattices, as well as extended neighborhood lattices which can be used for isothermal and thermal compressible fluids. Also listed here the $D2Q5$ and $D3Q7$ lattices that do not have enough isotropy to represent correctly fluids, but are adequate for the advection–diffusion schemes

Rule	Lattice	ξ_i	w_i
$E_{2,3}^5$	D2Q5	$(0, 0)$	$1 - 2c_l^2$
		$(c_l, 0)_{FS}$	$c_l^2/2$
$E_{2,5}^7$	D2Q7	$(0, 0)$	$c_l < \sqrt{1/2}$ $1/2$
		$2(\cos \frac{n\pi}{3}, \sin \frac{n\pi}{3})$	$1/12$
			$n = 1, \dots, 6$
$E_{2,5}^9$	D2Q9	$(0, 0)$	$4/9$
		$(\sqrt{3}, 0)_{FS}$	$1/9$
		$(\pm\sqrt{3}, \pm\sqrt{3})$	$1/36$
$E_{2,7}^{17}$	D2Q17	$(0, 0)$	$(575 + 193\sqrt{193})/8100$
		$(c_l, 0)_{FS}$	$(5555 - 91\sqrt{193})/18000$
		$(\pm c_l, \pm c_l)$	$(655 + 17\sqrt{193})/27000$
		$(\pm 2c_l, \pm 2c_l)$	$(685 - 49\sqrt{193})/54000$
		$(3c_l, 0)_{FS}$	$(1445 - 101\sqrt{193})/162000$
		$c_l = \sqrt{(125 + 5\sqrt{193})/72}$	
$E_{2,9}^{37}$	D2Q37	$(0, 0)$	0.23315066913235250228650
		$(c_l, 0)_{FS}$	0.10730609154221900241246
		$(\pm c_l, \pm c_l)$	0.05766785988879488203006
		$(2c_l, 0)_{FS}$	0.01420821615845075026469
		$(\pm 2c_l, \pm 2c_l)$	0.00101193759267357547541
		$(3c_l, 0)_{FS}$	0.00024530102775771734547
		$(c_l, 2c_l)_{FS}$	0.00535304900051377523273
$(c_l, 3c_l)_{FS}$	0.00028341425299419821740		
		$c_l = 1.19697977039307435897239$	

Table B.1: Gauss-Hermite quadrature formulae in two dimensions.

Rule	Lattice	ξ_i	w_i
$E_{3,3}^7$	D3Q7	$(0, 0, 0)$	$1 - 3c_l^2$
		$(c_l, 0, 0)_{FS}$	$c_l^2/2$
$E_{3,5}^{15}$	D3Q15	$(0, 0, 0)$	$c_l < \sqrt{1/3}$
		$(\sqrt{3}, 0, 0)_{FS}$	$2/9$
		$(\pm\sqrt{3}, \pm\sqrt{3}, \pm\sqrt{3})$	$1/9$
			$1/72$
$E_{3,5}^{19}$	D3Q19	$(0, 0, 0)$	$1/3$
		$(\sqrt{3}, 0, 0)_{FS}$	$1/18$
		$(\sqrt{3}, \sqrt{3}, 0)_{FS}$	$1/36$
$E_{3,5}^{27}$	D3Q27	$(0, 0, 0)$	$8/27$
		$(\sqrt{3}, 0, 0)_{FS}$	$2/27$
		$(\sqrt{3}, \sqrt{3}, 0)_{FS}$	$1/54$
		$(\pm\sqrt{3}, \pm\sqrt{3}, \pm\sqrt{3})$	$1/216$
$E_{3,7}^{39}$	D3Q39	$(0, 0, 0)$	$1/12$
		$(c_l, 0, 0)_{FS}$	$1/12$
		$(\pm c_l, \pm c_l, \pm c_l)$	$1/27$
		$(2c_l, 0, 0)_{FS}$	$2/135$
		$(2c_l, 2c_l, 0)_{FS}$	$1/432$
		$(3c_l, 0, 0)_{FS}$	$1/1620$
$E_{3,9}^{121}$	D3Q121		$c_l = \sqrt{3/2}$
		$(0, 0, 0)$	0.03059162202948600642469
		$(c_l, 0, 0)_{FS}$	0.09851595103726339186467
		$\pm c_l, \pm c_l, \pm c_l$	0.02752500532563812386479
		$(c_l, 2c_l, 0)_{FS}$	0.00611102336683342432241
		$(2c_l, 2c_l, 0)_{FS}$	0.00042818359368108406618
		$(3c_l, 0, 0)_{FS}$	0.00032474752708807381296
		$(2c_l, 3c_l, 0)_{FS}$	0.00001431862411548029405
		$(\pm 2c_l, \pm 2c_l, \pm 2c_l)$	0.00018102175157637423759
		$(c_l, 3c_l, 0)_{FS}$	0.00010683400245939109491
$(\pm 3c_l, \pm 3c_l, \pm 3c_l)$	0.00000069287508963860285		
		$c_l = 1.19697977039307435897239$	

Table B.2: Gauss-Hermite quadrature formulae in three dimensions.

Appendix C

Initial pressure field for the Taylor–Green vortex

The Taylor–Green benchmark is the time evolution of the initial condition

$$u_x(\mathbf{x}, t = 0) = \frac{2}{\sqrt{3}} \sin\left(\theta + \frac{2\pi}{3}\right) \sin(x) \cos(y) \cos(z), \quad (\text{C-1})$$

$$u_y(\mathbf{x}, t = 0) = \frac{2}{\sqrt{3}} \sin\left(\theta - \frac{2\pi}{3}\right) \cos(x) \sin(y) \cos(z), \quad (\text{C-2})$$

$$u_z(\mathbf{x}, t = 0) = \frac{2}{\sqrt{3}} \sin(\theta) \cos(x) \cos(y) \sin(z), \quad (\text{C-3})$$

where $\theta = \pi/2$. We are looking for the initial pressure. Which should be given by the solution of the incompressible Navier–Stokes equations,

$$\nabla \cdot \mathbf{u}(\mathbf{x}) = 0, \quad (\text{C-4})$$

$$\partial_t \mathbf{u} + (\mathbf{u}(\mathbf{x}) \cdot \nabla) \mathbf{u}(\mathbf{x}) = -\nabla p(\mathbf{x}) + \nu \nabla^2 \mathbf{u}(\mathbf{x}). \quad (\text{C-5})$$

One can easily show that the divergence of the velocity field is null for this initial condition. Taking divergence of Eq. (C-5) and use Eq. (C-4), and one gets

$$(\nabla \mathbf{u}) : (\nabla \mathbf{u}) = \nabla^2 p. \quad (\text{C-6})$$

Therefore one must solve

$$\begin{aligned} \frac{2}{3} (\cos^2 z + \cos^2 y \cos^2 z + \cos^2 z \cos^2 x - 2 \cos^2 y \\ + 4 \cos^2 x \cos^2 y - 2 \cos^2 x) = -\nabla^2 p. \end{aligned} \quad (\text{C-7})$$

Let us now for simplicity just rename the l.h.s. of this last equation as $f(\mathbf{x})$

$$\begin{aligned} f(\mathbf{x}) \equiv \frac{2}{3} \left(\cos^2 z + \cos^2 y \cos^2 z + \cos^2 z \cos^2 x \right. \\ \left. - 2 \cos^2 y + 4 \cos^2 x \cos^2 y - 2 \cos^2 x \right). \end{aligned}$$

We now expand $f(\mathbf{x})$ and $p(\mathbf{x})$ in Fourier series

$$f(\mathbf{x}) = \sum_{\mathbf{k} \in \mathbb{Z}} \tilde{f}_{\mathbf{k}} \exp(i\mathbf{k} \cdot \mathbf{x}), \quad (\text{C-8})$$

$$p(\mathbf{x}) = \sum_{\mathbf{k} \in \mathbb{Z}} \tilde{p}_{\mathbf{k}} \exp(i\mathbf{k} \cdot \mathbf{x}), \quad (\text{C-9})$$

where

$$\tilde{f}_{\mathbf{k}} = \frac{1}{8\pi^3} \iiint_{-\pi}^{\pi} d^3\mathbf{x} \exp(-i\mathbf{k} \cdot \mathbf{x}) f(\mathbf{x}), \quad (\text{C-10})$$

$$\tilde{p}_{\mathbf{k}} = \frac{1}{8\pi^3} \iiint_{-\pi}^{\pi} d^3\mathbf{x} \exp(-i\mathbf{k} \cdot \mathbf{x}) p(\mathbf{x}). \quad (\text{C-11})$$

Substituting this in Eq. (C-7) one gets

$$\tilde{p}_{\mathbf{k}} = \frac{\tilde{f}_{\mathbf{k}}}{\|\mathbf{k}\|^2}. \quad (\text{C-12})$$

And therefore

$$p(\mathbf{x}) = \sum_{\mathbf{k} \in \mathbb{Z}} \frac{\tilde{f}_{\mathbf{k}}}{\|\mathbf{k}\|^2} \exp(-i\mathbf{k} \cdot \mathbf{x}) \quad (\text{C-13})$$

The only relevant terms in this last sum are for $k_x, k_y, k_z \in \{-2, 0, 2\}$. And therefore one gets

$$\begin{aligned} p(\mathbf{x}) = & \frac{1}{6} \left(\frac{1}{4} \cos(2y) + \frac{1}{4} \cos(-2y + 2x) + \frac{1}{4} \cos(2x + 2y) + \cos(2z) \right. \\ & + \frac{1}{24} \cos(-2z + 2y) + \frac{1}{24} \cos(2z + 2y) + \frac{1}{24} \cos(2z + 2x) \\ & \left. + \frac{1}{24} \cos(-2z + 2x) + \frac{1}{4} \cos(2x) - 1 \right). \quad (\text{C-14}) \end{aligned}$$

Bibliography

- [1] J.-F. Agassant, P. Avenas, J.-P. Sergent, B. Vergnes, and M. Vincent. *La mise en forme des matières plastiques*. Tec & Doc, Lavoisier, Paris, 1996.
- [2] E. Aharonov and D. H. Rothman. Non-Newtonian flow (through porous media) : A lattice-Boltzmann method. *Geophys. Res. Lett.*, 679, 1993.
- [3] S. Ansumali and I. V. Karlin. Single relaxation time model for entropic lattice Boltzmann methods. *Phys. Rev. E*, 65(5):056312, May 2002. doi: 10.1103/PhysRevE.65.056312.
- [4] S. Ansumali and I. V. Karlin. Kinetic boundary conditions in the lattice Boltzmann method. *Phys. Rev. E*, 66(2):026311, Aug 2002. doi: 10.1103/PhysRevE.66.026311.
- [5] P. E. Arratia, C. C. Thomas, J. Diorio, and J. P. Gollub. Elastic instabilities of polymer solutions in cross-channel flow. *Phys. Rev. Lett.*, 96(14):144502, 2006. doi: 10.1103/PhysRevLett.96.144502.
- [6] S. Berti, A. Bistagnino, G. Boffetta, A. Celani, and S. Musacchio. Two-dimensional elastic turbulence. *Phys. Rev. E*, 77(5):055306, 2008. doi: 10.1103/PhysRevE.77.055306.
- [7] 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(3):511–525, May 1954. doi: 10.1103/PhysRev.94.511.
- [8] R. B. Bird, C. F. Curtiss, R. C. Armstrong, and O. Hassager. *Dynamics of Polymeric Liquids, Kinetic Theory (Dynamics of Polymer Liquids Vol. 2)*. Wiley-Interscience, New York, 1987. ISBN 0471802441.
- [9] R. B. Bird, C. F. Curtiss, R. C. Armstrong, and O. Hassager. *Dynamics of Polymeric Liquids, Fluid Mechanics (Dynamics of Polymer Liquids Vol. 1)*. Wiley-Interscience, New York, 1987. ISBN 0471802441.
- [10] C. Boeckle. On the origin of numerical instabilities in the simulations of viscoelastic fluid flows. Master’s thesis, EPFL, Switzerland, 2009.

- [11] B. M. Boghosian, P. J. Love, P. V. Coveney, I. V. Karlin, S. Succi, and J. Yepez. Galilean-invariant lattice-Boltzmann models with H theorem. *Phys. Rev. E*, 68(2):025103, Aug 2003. doi: 10.1103/PhysRevE.68.025103.
- [12] A. Bonito, M. Picasso, and M. Laso. Numerical simulation of 3D viscoelastic flows with free surfaces. *J. Comput. Phys.*, 215(2):691–716, JUL 1 2006. doi: {10.1016/j.jcp.2005.11.013}.
- [13] LB boundary condition benchmarks. http://spc.unige.ch/lb_boundaries.
- [14] M. Bouzidi, M. Firdaouss, and P. Lallemand. Momentum transfer of a Boltzmann-lattice fluid with boundaries. *Phys. Fluids*, 13(11):3452–3459, 2001. doi: 10.1063/1.1399290.
- [15] J. Boyd, J. Buick, and S. Green. A second-order accurate lattice Boltzmann non-Newtonian flow model. *J. Phys. A: Math. Gen.*, 39(46):14241–14247, 2006.
- [16] M. E. Brachet, D. I. Meiron, S. A. Orszag, B. G. Nickel, R. H. Morf, and U. Frisch. Small-scale structure of the Taylor–Green vortex. *J. Fluid Mech.*, 120:411–452, 1983. doi: 10.1017/S0022112083001159.
- [17] S. Chapman and T. G. Cowling. *The mathematical theory of nonuniform gases*. Cambridge University Press, Cambridge, 1960.
- [18] S. Chen and G. D. Doolen. Lattice Boltzmann method for fluid flows. *Ann. Rev. Fluid Mech.*, 30:329–364, 1998. doi: 10.1146/annurev.fluid.30.1.329.
- [19] B. Chopard and A. Dupuis. A mass conserving boundary condition for lattice boltzmann models. *Int. J. Mod. Phys. B*, 17(1/2):103–108, 2003.
- [20] B. Chopard, A. Dupuis, A. Masselot, and P. Luthi. Cellular automata and lattice Boltzmann techniques: an approach to model and simulate complex systems. *Adv. Compl. Sys.*, 5:103–246, 2002. doi: 10.1142/S0219525902000602.
- [21] B. Chopard, J.-L. Falcone, and J. Latt. The lattice Boltzmann advection-diffusion model revisited. *Eur. Phys. J.*, 171:245–249, 2009.
- [22] H. J. H. Clercx and C.-H. Bruneau. The normal and oblique collision of a dipole with a no-slip boundary. *Comput. Fluids*, 35:245–279, 2006.

- [23] J. A. Cosgrove, J. M. Buick, S. J. Tonge, C. G. Munro, C. A. Greated, and D. M. Campbell. Application of the lattice Boltzmann method to transition in oscillatory channel flow. *J. Phys. A : Math. Gen.*, 36: 2609–2620, 2003.
- [24] P. G. de Gennes. Reptation of a polymer chain in the presence of fixed obstacles. *J. Chem. Phys.*, 55(2):572–579, 1971. doi: 10.1063/1.1675789. URL <http://link.aip.org/link/?JCP/55/572/1>.
- [25] P. J. Dellar. Bulk and shear viscosities in lattice Boltzmann equations. *Phys. Rev. E*, 64(3):031203, Aug 2001. doi: 10.1103/PhysRevE.64.031203.
- [26] P. J. Dellar. Incompressible limits of lattice Boltzmann equations using multiple relaxation times. *J. Comp. Phys.*, 190:351–370, 2003.
- [27] C. Denniston, E. Orlandini, and J. M. Yeomans. Lattice Boltzmann simulations of liquid crystal hydrodynamics. *Phys. Rev. E*, 63(5): 056702, Apr 2001. doi: 10.1103/PhysRevE.63.056702.
- [28] M. Deville, P. F. Fischer, and E. Mund. *High-order methods for incompressible fluid flow*. Cambridge University Press, Cambridge, 2002. ISBN 0-521-45309-7.
- [29] D. d’Humières. Generalized lattice-Boltzmann equations. *Progress in Astronautics and Aeronautics*, 159:450–458, 1992.
- [30] D. d’Humières, I. Ginzburg, M. Krafczyk, P. Lallemand, and L.-S. Luo. Multiple-relaxation-time lattice Boltzmann models in three dimensions. *Phil. Trans. R. Soc. A*, 360:437–451, 2002.
- [31] M. Doi and S. F. Edwards. *The Theory of Polymer Dynamics*. Oxford University Press, Oxford, 1986. ISBN 0198520336.
- [32] A. W. El-Kareh and L. G. Leal. Existence of solutions for all Deborah numbers for a non-Newtonian model modified to include diffusion. *J. Non-Newtonian Fluid Mech.*, 33(3):257 – 287, 1989. ISSN 0377-0257. doi: DOI:10.1016/0377-0257(89)80002-3.
- [33] N. Fiétier. *Numerical simulation of viscoelastic fluid flows by spectral element methods and time-dependent algorithms*. PhD thesis, Ecole Polytechnique Fédérale de Lausanne, Lausanne, Switzerland, 2002. URL <http://library.epfl.ch/theses/?nr=2631>.

-
- [34] U. Frisch, B. Hasslacher, and Y. Pomeau. Lattice-gas automata for the Navier-Stokes equation. *Phys. Rev. Lett.*, 56(14):1505–1508, Apr 1986. doi: 10.1103/PhysRevLett.56.1505.
- [35] S. Gabbanelli, G. Drazer, and J. Koplik. Lattice Boltzmann method for non-Newtonian (power-law) fluids. *Phys. Rev. E*, 72(4):046312, Oct 2005. doi: 10.1103/PhysRevE.72.046312.
- [36] I. Ginzburg and D. d’Humières. Multireflection boundary conditions for lattice Boltzmann models. *Phys. Rev. E*, 68(6):066614, Dec 2003. doi: 10.1103/PhysRevE.68.066614.
- [37] L. Giraud, D. d’Humières, and P. Lallemand. A lattice-Boltzmann model for visco-elasticity. *Int. J. of Mod. Phys. C*, 8:805–815, 1997.
- [38] L. Giraud, D. d’Humières, and P. Lallemand. A lattice Boltzmann model for Jeffreys viscoelastic fluid. *Europhys. Lett.*, 42(6):625–630, 1998.
- [39] H. Grad. Note on the N-dimensional Hermite polynomials. *Commun. Pure Appl. Maths*, 9:325, 1949.
- [40] H. Grad. On the kinetic theory of rarefied gases. *Commun. Pure Appl. Maths*, 9:331, 1949.
- [41] E. Grande, M. Laso, and M. Picasso. Calculation of variable-topology free surface flows using CONNFFESSIT. *J. Non-Newtonian Fluid Mech.*, 113(2-3):127–145, 2003. ISSN 0377-0257. doi: {10.1016/S0377-0257(03)00103-4}.
- [42] M. Griebel, T. Dornseifer, and T. Neunhoeffler. *Numerical simulation in fluid dynamics: a practical introduction*. Society for Industrial and Applied Mathematics, Philadelphia, PA, USA, 1998. ISBN 0-89871-398-6.
- [43] Z. Guo and C. Zheng. An extrapolation method for boundary conditions in lattice Boltzmann method. *Phys. Fluids*, 14:2007–2010, 2002.
- [44] Z. Guo, B. Shi, and C. Zheng. A coupled lattice BGK model for the Boussinesq equations. *Int. J. Num. Meth. Fluids*, 39:325–342, 2002. doi: 10.1002/flid.337.
- [45] 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.

-
- [46] 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.
- [47] D. Hänel. *Einführung in die kinetische Theorie der Gase und Lattice-Boltzmann-Methoden*. Springer Verlag, Berlin, 2004.
- [48] X. He and L.-S. Luo. Theory of the lattice Boltzmann method : From the Boltzmann equation to the lattice Boltzmann equation. *Phys. Rev. E*, 56:6811–6818, 1997. doi: 10.1103/PhysRevE.56.6811.
- [49] X. He, X. Shan, and G. D. Doolen. Discrete Boltzmann equation model for non-ideal gases. *Phys. Rev. E*, 57(1):R13–R16, Jan 1998. doi: 10.1103/PhysRevE.57.R13.
- [50] J. P. Hernández-Ortiz, J. J. de Pablo, and M. D. Graham. Fast computation of many-particle hydrodynamic and electrostatic interactions in a confined geometry. *Phys. Rev. Lett.*, 98(14):140602, 2007. doi: 10.1103/PhysRevLett.98.140602.
- [51] V. Heuveline and J. Latt. The OpenLB project: an open source and object oriented implementation of lattice Boltzmann methods. *Int. J. Mod. Phys. C*, 18:627–634, 2007.
- [52] F. J. Higuera and J. Jimenez. Boltzmann approach to lattice gas simulations. *Europhys. Lett.*, 9(7):663–668, 1989.
- [53] K. Huang. *Statistical Mechanics*. J. Wiley, New York, 1987.
- [54] T. Inamuro, M. Yoshina, and F. Ogino. A non-slip boundary condition for lattice Boltzmann simulations. *Phys. Fluids*, 7:2928–2930, 1995.
- [55] I. Ispolatov and M. Grant. Lattice Boltzmann method for viscoelastic fluids. *Phys. Rev. E*, 65(5):056704, May 2002. doi: 10.1103/PhysRevE.65.056704.
- [56] A. Izmitli, D. C. Schwartz, M. D. Graham, and J. J. de Pablo. The effect of hydrodynamic interactions on the dynamics of DNA translocation through pores. *The Journal of Chemical Physics*, 128(8):085102, 2008. doi: 10.1063/1.2831777.
- [57] M. Junk and Z. Yang. One-point boundary condition for the lattice Boltzmann method. *Phys. Rev. E*, 72(6):066701, Dec 2005. doi: 10.1103/PhysRevE.72.066701.

- [58] D. Kehrwald. Lattice Boltzmann simulation of shear-thinning fluids. *J. Stat. Phys.*, 121:223–237(15), 2005. doi: doi:10.1007/s10955-005-5963-z.
- [59] R. Keunings. Finite element methods for integral viscoelastic fluids. *Rheology Reviews 2003*, pages 167–195, 2003.
- [60] R. Keunings. Micro-macro methods for the multiscale simulation of viscoelastic flow using molecular models of kinetic theory. *Rheology Reviews 2004*, pages 67–98, 2004.
- [61] V. I. Krylov. *Approximate Calculation of Integrals*. Macmillan, New York, 1962.
- [62] P. Lallemand, D. d’Humières, L.-S. Luo, and R. Rubinstein. Theory of the lattice Boltzmann method: Three-dimensional model for linear viscoelastic fluids. *Phys. Rev. E*, 67(2):021203, Feb 2003. doi: 10.1103/PhysRevE.67.021203.
- [63] R. G. Larson. *Constitutive Equations for Polymer Melts and Solutions*. Butterworths, Boston, 1988.
- [64] R. G. Larson. *The Structure and Rheology of Complex Fluids*. Oxford University Press, Oxford, 1998.
- [65] M. Laso and H. C. Öttinger. Calculation of viscoelastic flow using molecular models: the CONNFFESSIT approach. *J. Non-Newtonian Fluid Mech.*, 47:1–20, 1993.
- [66] J. Latt. *Hydrodynamic limit of lattice Boltzmann equations*. PhD dissertation, University of Geneva, Geneva, Switzerland, 2007. URL <http://www.unige.ch/cyberdocuments/theses2007/LattJ/meta.html>.
- [67] J. Latt and B. Chopard. A benchmark case for lattice Boltzmann: turbulent dipole-wall collision. *Int. J. Mod. Phys. C*, 18:619–626, 2007.
- [68] J. Latt and B. Chopard. Lattice Boltzmann method with regularized non-equilibrium distribution functions. *Math. Comp. Sim.*, 72:165–168, 2006.
- [69] J. Latt, B. Chopard, O. Malaspinas, M. Deville, and A. Michler. Straight velocity boundaries in the lattice Boltzmann method. *Phys. Rev. E*, 77(5):056703–+, May 2008. doi: 10.1103/PhysRevE.77.056703.

- [70] Y.-H. Lin. *Polymer Viscoelasticity : Basics, Molecular Theories and Experiments*. World Scientific Publishing, Singapore, 2003.
- [71] O. Malaspinas, G. Courbebaisse, and M. O. Deville. Simulation of generalized Newtonian fluids with the lattice Boltzmann method. *Int. J. of Mod. Phys. C*, 18:1939–1949, 2007. doi: 10.1142/S0129183107011832.
- [72] O. Malaspinas, M. O. Deville, and B. Chopard. Towards a physical interpretation of the entropic lattice Boltzmann method. *Phys. Rev E*, 78:066705, 2008.
- [73] T. C. B McLeish. Tube theory of entangled polymer dynamics. *Adv. Phys.*, 51:1379–1527, 2002.
- [74] G. R. McNamara and G. Zanetti. Use of the Boltzmann equation to simulate lattice-gas automata. *Phys. Rev. Lett.*, 61:2332–2335, 1988. doi: 10.1103/PhysRevLett.61.2332.
- [75] R. Mei, L.-S. Luo, P. Lallemand, and D. d’Humières. Consistent initial conditions for lattice Boltzmann simulations. *Comp. fluids*, 35:855–862, 2006. doi: doi:10.1016/j.compfluid.2005.08.008.
- [76] X. B. Nie, X. Shan, and H. Chen. Galilean invariance of lattice Boltzmann models. *Europhys. Lett.*, 81(3):34005 (6pp), 2008.
- [77] J. G. Oldroyd. On the Formulation of Rheological Equations of State. *Proc. R. Soc. London, Ser. A*, 200:523–541, 1950.
- [78] J. Onishi, Y. Chen, and H. Ohashi. A lattice Boltzmann model for polymeric liquids. *Prog. Comp. Fluid Dyn.*, 5:75–84, 2005. doi: 10.1504/PCFD.2005.005819.
- [79] J. Onishi, Y. Chen, and H. Ohashi. Dynamic simulation of multi-component viscoelastic fluids using the lattice Boltzmann method. *Physica A*, 362(1):84 – 92, 2006. ISSN 0378-4371. doi: DOI:10.1016/j.physa.2005.09.022.
- [80] H. C. Öttinger. *Stochastic processes in polymeric fluids : tools and examples for developing simulation algorithms*. Springer, Berlin, 1996.
- [81] R. Ouared and B. Chopard. Lattice Boltzmann simulations of blood flow: Non-Newtonian rheology and clotting processes. *J. Stat. Phys.*, 121:209–221(13), 2005. doi: doi:10.1007/s10955-005-8415-x.

- [82] R. G. Owens and T. N. Phillips. *Computational Rheology*. Imperial College Press, London, 2002.
- [83] A. Peterlin. Streaming birefringence of soft linear macromolecules with finite chain length. *Polymer*, 2:257 – 264, 1961. ISSN 0032-3861. doi: DOI:10.1016/0032-3861(61)90029-5.
- [84] R. J. Poole, M. A. Alves, and P. J. Oliveira. Purely elastic flow asymmetries. *Phys. Rev. Lett.*, 99(16):164503, 2007. doi: 10.1103/PhysRevLett.99.164503.
- [85] W. Press, S. Teukolsky, W. Vetterling, and B. Flannery. *Numerical Recipes in C++: The Art of Scientific Computing*. Cambridge University Press, Cambridge, 2002.
- [86] The OpenLB project, . <http://www.lbmethod.org/openlb>.
- [87] The Palabos project, . <http://www.lbmethod.org/palabos>.
- [88] Y.-H. Qian and Y.-F. Deng. A lattice BGK model for viscoelastic media. *Phys. Rev. Lett.*, 79(14):2742–2745, Oct 1997. doi: 10.1103/PhysRevLett.79.2742.
- [89] T. Reis and T. N. Phillips. Lattice Boltzmann model for simulating immiscible two-phase flows. *J. Phys. A: Math. Theor.*, 40(14):4033–4053, 2007. URL <http://stacks.iop.org/1751-8121/40/4033>.
- [90] X. Shan and H. Chen. A general multiple-relaxation-time Boltzmann collision model. *Int. J. of Mod. Phys C*, 18:635, 2007.
- [91] X. Shan and H. Chen. Lattice Boltzmann model for simulating flows with multiple phases and components. *Phys. Rev. E*, 47:1815–1819, 1993. doi: 10.1103/PhysRevE.47.1815.
- [92] 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, March 2006. doi: 10.1017/S0022112005008153.
- [93] P. A. Skordos. Initial and boundary conditions for the lattice Boltzmann method. *Phys. Rev. E*, 48:4823–4842, 1993.
- [94] S. Succi. *The lattice Boltzmann equation for fluid dynamics and beyond*. Oxford University Press, Oxford, 2001.

- [95] B. Thomases and M. Shelley. Emergence of singular structures in Oldroyd-B fluids. *Physics of Fluids*, 19(10):103103, 2007. doi: 10.1063/1.2783426.
- [96] M.F. Tomé, L. Grossi, A. Castelo, J.A. Cuminato, S. McKee, and K. Walters. Die-swell, splashing drop and a numerical technique for solving the Oldroyd-B model for axisymmetric free surface flows. *J. Non-Newtonian Fluid Mech.*, 141(2-3):148 – 166, 2007. doi: DOI:10.1016/j.jnnfm.2006.09.008.
- [97] H. K. Versteeg and W. Malalasekera. *An Introduction to Computational Fluid Dynamics: The Finite Volume Method*. Addison-Wesley, Reading, 1995.
- [98] A. Vikhansky. Lattice-Boltzmann method for yield-stress liquids. *J. Non-Newtonian Fluid Mech.*, 155(3):95 – 100, 2008. ISSN 0377-0257. doi: DOI:10.1016/j.jnnfm.2007.09.001.
- [99] O. Walsh. Eddy solutions of the Navier-Stokes equations. In J. G. Heywood, K. Masuda, R. Rautmann, and S. A. Solonnikov, editors, *The Navier-Stokes Equations II – Theory and Numerical Methods*, volume 1530 of *Lecture Notes in Mathematics*, pages 305–309. Springer-Verlag, 1991.
- [100] D. A. Wolf-Gladrow. *Lattice-Gas Cellular Automata and Lattice Boltzmann Models: An Introduction*. Springer-Verlag, Berlin, 2000.
- [101] J. R. Womersley. Method for the calculation of velocity, rate of flow and viscous drag in arteries when the pressure gradient is known. *J. Physiol.*, 127:553–563, 1955.
- [102] C.-S. Yih. *Fluid Mechanics*. McGraw-Hill, London, 1969.
- [103] O. C. Zienkiewicz, R. L. Taylor, and P. Nithiarasu. *The Finite Element Method for Fluid Dynamics, Sixth Edition*. Butterworth-Heinemann, Oxford, 6th edition, December 2005. ISBN 0750663227.
- [104] Q. Zou and X. He. On pressure and velocity boundary conditions for the lattice Boltzmann BGK model. *Phys. Fluids*, 9:1592–1598, 1997.
- [105] R. Zwanzig. *Nonequilibrium statistical mechanics*. Oxford University Press, New York, 2001.

ORESTIS PILEAS MALASPINAS

55, rue de Monthoux
CH-1201 Genève
Suisse

Tel.: +41 22 732 28 59 (home)
+41 79 420 33 89 (mobile)

e-mail: orestis.malaspinas@epfl.ch

Date of birth : 16 December 1980

Nationality : Greek and Swiss

Languages : Greek, French, English, German (school level)

Education

Currently PHD IN FLUID MECHANICS – Ecole Polytechnique Fédérale de
Lausanne, CH
07/2004 MASTER IN PHYSICS – University of Geneva, CH
06/1999 HIGH SCHOOL DIPLOMA – Collège de Candolle, Geneva, CH

Employments

08/05 - now ASSISTANT
Ecole Polytechnique Fédérale de Lausanne, CH
10/04 - 05/05 RESEARCH ASSISTANT
Geneva University Hospital, CH
08/03 - 07/04 PHYSICS ASSISTANT
(teaching of physics experiments)
Collège Emilie-Gourd, Geneva, CH
10/01 - 07/03 LABORATORY ASSISTANT
(teaching of physics experiments for biologists)
University of Geneva, CH

Teaching experience

- Spring 2009 LATTICE BOLTZMANN METHOD FOR FLUID FLOWS
Ecole Polytechnique Fédérale de Lausanne, CH
- Spring 2006-09 HYDRODYNAMICS
Ecole Polytechnique Fédérale de Lausanne, CH
- Spring 2006-08 CONTINUUM MECHANICS
Ecole Polytechnique Fédérale de Lausanne, CH

Peer Review

Comp. & Fluids, Phys. Rev. E, Phys. Rev. Lett.

Publications

Research Articles

- [1] O. Malaspinas & R. Sturani, *Detecting a stochastic background of gravitational waves by correlating n detectors*, *Class. and Quantum Grav.*, **23**, 319-327, (2006).
- [2] O. Malaspinas, G. Courbebaisse & M. Deville, *Simulation of Generalized Newtonian Fluids with the Lattice Boltzmann Method*, *Int. J. of Mod. Phys. C*, **18**, 1939-1944, (2007).
- [3] J. Latt, B. Chopard, O. Malaspinas, M. O. Deville & A. Michler, *Straight velocity boundaries in the lattice Boltzmann method*, *Phys. Rev. E*, **77**, 056703, (2008).
- [4] O. Malaspinas, M. O. Deville & B. Chopard, *Towards a physical interpretation of the entropic lattice Boltzmann method*, *Phys. Rev. E*, **78**, 066705, (2008).

Theses

- [1] O. Malaspinas, *Détection de fonds stochastiques d'ondes gravitationnelles avec N détecteurs*, Master Thesis, University of Geneva, 2004
- [2] O. Malaspinas, *Lattice Boltzmann method for the simulation of viscoelastic fluid flows*, PhD Thesis, Ecole Polytechnique Fédérale de Lausanne, 2009

Conference Presentations

- Summer 2006 DISCRETE SIMULATION OF FLUID DYNAMICS (DSFD)
Geneva, CH
- Summer 2007 DISCRETE SIMULATION OF FLUID DYNAMICS (DSFD)
Banff, CA
- Summer 2008 INTERNATIONAL CONFERENCE FOR MESOSCOPIC METHODS IN
ENGINEERING AND SCIENCE (ICMMES)
Amsterdam, NL
- Autumn 2008 ANNUAL MEETING OF THE LEONHARD EULER CENTER SWISS
ERCOFTAC PILOT CENTER
Zürich, CH

Contributed Poster Sessions

- Fall 2007 AMERICAN GEOPHYSICAL UNION (AGU) FALL MEETING
San Francisco, CH
- Summer 2008 INTERNATIONAL ASSOCIATION OF VOLCANOLOGY AND CHEM-
ISTRY OF THE EARTH'S INTERIOR GENERAL ASSEMBLY
Reykjavik, IS

Software

- Co-developer of Palabos, Open-Source C++ Parallel Lattice Boltzmann Solver,
<http://www.lbmethod.org/palabos>.
- Co-maintainer of the lattice Boltzmann website,
<http://www.lbmethod.org>.

Programming languages and scientific software

C++, Fortran, Pascal; Maple, Matlab.

Other

- 10/01 - 10/03 President of the Physics Students Association
University of Geneva, CH
- 09/99 - 06/03 Trainer of the water polo school of Carouge Natation
Former member of Switzerland's water polo National team