Lattice Boltzmann Ocean Simulator

Postgraduate Programme Environmental Physics

Universität Bremen / Alfred Wegener Institut Bremerhaven

Submitted in partial fulfillment of the M.Sc. in Environmental Physics Degree

Supervisor: Prof. Gerrit Lohmann

Second Reviewer: Prof. Sergey Danilov

Dragos B. Chirila

Matriculation No. **2268745**

July 30, 2009

Contents

Declaration

I herewith declare that I did the written work on my own and only with the means indicated.

Date:_____ Signature:_____

Abstract

Lattice Boltzmann (LB) methods have recently emerged as a new class of viable simulation techniques for fluid flow problems. In the present work, we investigated the potential of this method for ocean dynamics simulation. First, we constructed a Lattice Boltzmann simulator using standard techniques and tested its behaviour on a well-known fluid mechanics problem (***3D*** Poiseuille flow). This was essentially a Direct Numerical Simulation (DNS) solver, as no turbulence model was included. The program was then extended using a Smagorinsky-type turbulence model (also documented in the literature), allowing the simulation of more realistic ocean dynamics. This refinement allowed us to study more complex cases, such as the lid-driven cavity. Nonetheless, oceanic flows require more involved turbulence parameterizations than the one introduced by Smagorinsky, due to the complications caused by the stronger stratification. Thus, the aim of the last part of the project was to introduce a new methodology for including these higher-order oceanic turbulence models into the standard LB algorithm. This part and the results therein also constitute the main contribution of this thesis. The new methodology allows LB to easily incorporate various turbulence models in common use in oceanography.

1 Introduction

The Earth climate is manifested through the dynamics and interactions of many subsystems[1]. Due to its high latent heat, water regulates most of the energy transfers between these components. As most of the water on the Earth is accumulated in the global ocean, this is also the largest heat reservoir on the Earth, bearing an important influence on climate variability. The characteristic timescale for changes in the oceanic circulation is in the range ($\mathbf{10^{-1}}$ yr,$\mathbf{10^{2}}$ yr), much larger than for the atmospheric circulation ($\mathbf{10^{-5}}$ yr, $\mathbf{10^{-1}}$ yr). In principle, this can be also an advantage for numerical simulations, as the Reynolds number is smaller compared to the atmospheric; however, longer integration times are also implied for oceanic dynamics simulations, which makes the two systems comparable in terms of computational demands.

As in the case of most fluids, the dynamics of the oceanic water is governed by the Navier-Stokes equations, whose nonlinear terms renders them very difficult to solve analytically in any non-trivial situation. Adding to this the high costs necessary for setting-up and quantifying relevant experimental systems[2], we conclude that the most accessible method of enhancing the knowledge in this field is through numerical simulations. Although the power of the computer systems is steadily increasing, the complexity of the ocean imposes high demand for efficient numerical methods.

Lattice Boltzmann Methods (henceforth LBM) represent relatively new techniques for fluid dynamics simulations, which have been extensively used for smaller-scale simulations. They emerged out of statistical mechanics rather than out of the macroscopic conservation laws[3]. The basic idea is to discretize both time and phase space, and also to separate the dynamics of the fluid particles into multiple scales, expressed through a collision and a streaming step. The streaming step involves only nearest neighbours, while the collision step is local and consists of a relaxation towards the local Boltzmann dis-

[1] These are usually classified as: atmosphere, hydrosphere, cryosphere, lithosphere and biosphere

[2] Controlled experiments are, indeed, out of the reach of geophysical fluid dynamicists in general and of oceanographers in particular. They are constrained to work with data that may be influenced by processes not completely understood. However, this is also one of the factors that make this field so fascinating.

[3] Which is the case for the Navier-Stokes equations [Landau and Lifshitz, 1987].

tribution, as required by Boltzmann's H-theorem [Ferziger and Kaper, 1972]. This leads to a conceptually simple but very powerful paradigm, which can be proved to effectively approximate the Navier-Stokes equations under the assumption of a low Mach number[4]. From a computational point of view, the locality of the algorithm leads to greater benefits from parallel computing compared to traditional Computational Fluid Dynamics (henceforth CFD), where an expensive global pressure correction step is usually required. Another benefit of the method relevant for the present project is the relative simplicity of implementing proper boundary conditions, which is of crucial importance in real-world oceanographic simulations, where the effects due to the coastline have to be resolved.

The aim of the present work was to extend the range of applicability of LBM to include problems in numerical oceanography. The first applications of the method to oceanographic problems were proposed by [Salmon, 1999a], [Salmon, 1999b] and [Wolf-Gladrow, 2000]. While the results reported by these authors are very promising, they are restricted either to the two-dimensional case or prototype 3D models with limiting assumptions and/or insufficient symmetry. To our knowledge, there has to date not been a concentrated effort on constructing a three-dimensional implementation targeting geophysical fluid dynamics. Adding this third dimension is a challenging yet highly necessary topic of research. Of course, the traditional oceanographic models have evolved simultaneously with computer hardware, and three-dimensional simulations are nowadays part of the standard requirements, as they provide additional information on the mechanisms driving the large-scale ocean circulation. Also, 3D simulations enable realistic modelling of the various fluxes between the ocean and the other components of climate. A successful LB-based model can also enable higher-resolution studies, due to the potentially greater per-gridpoint efficiency of the method.

The difficulty of the task consists of the fact that LBM assumes isotropy of both fluid viscosity and spatial resolution. Both of these assumptions are challenged by our current understanding of the oceanic system and our experience gained through modelling it with other numerical schemes; hence the need for fundamental changes to the LBM paradigm.

The present thesis is organized as follows: in **Section 2** we present a review of the theoretical concepts related to LBM, the most important aspects of the algorithm

[4]This assumption holds for the applications in geophysical fluid dynamics.

and some validation studies performed with our code; in **Section 3** we develop a new methodology for including turbulence closures in LBM, along with the proof that the new model does indeed solve the Reynolds-averaged Navier-Stokes equations; **Section 4** concludes the thesis and presents some outlook on the most probable pathways for future work.

2 Classical Lattice Boltzmann Methods

2.1 The Boltzmann Equation

One of the most significant theoretical breakthroughs in statistical physics was due to Ludwig Boltzmann (1864) (see [Boltzmann, 1995] for a recent reprint of his famous lectures on kinetic theory), who pioneered non-equilibrium statistical mechanics. Although Boltzmann's theory was conceived for diluted gases, the reader should be informed that during the last 50 years this treatment was extended to other classes of fluids as well. Indeed, the Lattice Boltzmann method, which is the focus of the current work, works in the assumption of a low Knudsen number[5], that is it fails exactly for diluted gases. However, for the sake of conciseness, only the original reasoning is presented in this subsection.

Boltzmann postulated that a gas was composed of a set of interacting particles, whose dynamics could be (at least in principle) modelled by classical dynamics. Due to the very large number of particles in such a system, a statistical approach was adopted, based on simplified physics composed of particle streaming in space and billiard-like inter-particle collisions (which are assumed elastic). Instrumental to the theory is the single-particle distribution function (hereafter SPDF), $\boldsymbol{f(\vec{x}, \vec{p}, t)}$ which represents the probability density of having a particle at the point $\boldsymbol{(\vec{x}, \vec{p})}$ in the phase space. Hence, the quantity

$$\boldsymbol{f(\vec{x}, \vec{p}, t) d\vec{x} d\vec{p}} \tag{1}$$

represents the probability of finding a particle inside an infinitesimal space cubelet centered around $\boldsymbol{\vec{x}}$, and inside an infinitesimal momentum-space cubelet around $\boldsymbol{\vec{p}}$ at any given time $\boldsymbol{t}$. In the presence of a body-force $\boldsymbol{\vec{F}}$, the SPDF will evolve according to

$$\boldsymbol{f(\vec{x} + d\vec{x}, \vec{p} + d\vec{p}, t + dt) d\vec{x} d\vec{p} = f(\vec{x}, \vec{p}, t) d\vec{x} d\vec{p}} \tag{2}$$

where

$$\boldsymbol{d\vec{x} = \frac{\vec{p}}{m} dt}$$

[5]The Knudsen number ($\boldsymbol{Kn}$) is a dimensionless quantity defined as the ratio of the mean free path length $\boldsymbol{\lambda}$ and a characteristic macroscopic lengthscale $\boldsymbol{L}$ of the process of interest.

and

$$d\vec{p} = \vec{F}dt.$$

If we also include the effect of the collisions, and denote by $\Gamma_+ d\vec{x}d\vec{p}dt$ the probability for a particle to start from outside the $d\vec{x} \times d\vec{p}$ domain and to enter this phase-space region during the infinitesimal time dt and by $\Gamma_- d\vec{x}d\vec{p}dt$ the probability for a particle to start from the $d\vec{x} \times d\vec{p}$ domain and leave this phase-space region during the infinitesimal time dt, the evolution of the SPDF becomes

$$f(\vec{x} + d\vec{x}, \vec{p} + d\vec{p}, t + dt)d\vec{x}d\vec{p} = f(\vec{x}, \vec{p}, t)d\vec{x}d\vec{p} + (\Gamma_+ - \Gamma_-)\, d\vec{x}d\vec{p}dt \quad (3)$$

Expanding the LHS into a Taylor series around the phase-space point $(\vec{x}, \vec{p}, t)$, we obtain:

$$f(\vec{x}+d\vec{x}, \vec{p}+d\vec{p}, t+dt)d\vec{x}d\vec{p} = f(\vec{x}, \vec{p}, t)d\vec{x}d\vec{p} + \left(\frac{\partial f}{\partial t}\right) dt + (\nabla_{\vec{x}} f)\cdot d\vec{x} + (\nabla_{\vec{p}} f)\cdot d\vec{p} + \ldots \quad (4)$$

Inserting Eq. (4) into Eq. (3) and cancelling terms, we easily obtain Boltzmann's Equation:

$$\boxed{\frac{\partial f}{\partial t} + \vec{u}\nabla_{\vec{x}} f + \vec{F}\nabla_{\vec{p}} f = \Gamma_+ - \Gamma_-} \quad (5)$$

where $\nabla_{\vec{x}}$ is the gradient operator in physical space and $\nabla_{\vec{p}}$ the same in momentum space.

For the sake of clarity, we have not written the collision operator explicitly yet. The important point is that the separation of the dynamics into collisions and streaming is already apparent from Eq. (5). The collision operator, which is in itself a complex integro-differential expression, reads

$$\Gamma_+ - \Gamma_- \equiv \int d^3\vec{u}_1 \int d\Omega\, \sigma(\Omega)\, |\vec{u} - \vec{u}_1|\, \left[f(\vec{u}')f(\vec{u}_1') - f(\vec{u})f(\vec{u}_1)\right] \quad (6)$$

where σ is the differential cross-section in the case of the 2-particle collisions (which is a function of the solid angle Ω), unprimed velocities are incoming (before collision) and primed velocities are outgoing (after collision).

A fundamental property of the collision operator [Cercignani, 1987] is that it conserves mass, momentum and kinetic energy (hence also a linear combination thereof). Also, it can be shown that the local Maxwell-Boltzmann distribution pertains to a certain class

of positive SPDFs for which the collision integral vanishes. This implies that, if this distribution is attained, we also have a state where incoming SPDFs exactly balance the outgoing ones, maintaining a local dynamic equilibrium. This observation is of paramount importance for our method, which uses the (discretized) Maxwell-Boltzmann distribution as the equilibrium distribution functions (hereafter EDFs).

Due to the complex expression for the collision operator, it became clear that approximations were desirable. It was also proven (see [Cercignani, 1990]) that such approximations were also reasonable, since the details of the two-body interaction are not likely to influence significantly experimentally-measured quantities. Hence, approximate collision operators were proposed, all of which had to $\boxed{1}$ conserve local mass, momentum and energy and $\boxed{2}$ develop a collisional contribution in Boltzmann's equation (5) which tends to a local Maxwellian distribution (which is required by Boltzmann's H-theorem - see [Wolf-Gladrow, 2000] for a full discussion). It was soon realized that a model developed at the middle of last century [Bhatnagar et al., 1954] (also known as Bhatnagar-Gross-Krook; hereafter BGK) satisfied both of these conditions. The basic idea was that each collision changes the SPDF by an amount which is proportional to the departure from the local Maxwellian distribution.

2.2 Theory of LBM

LBM evolved out of Lattice-Gas Cellular Automata (hereafter LGCA), statistical toy-models inspired by the Boltzmann theory which simulated a gas through particles at discrete points in space represented through Boolean variables. Following some physically-justified collision rules, these systems exhibited fluid-like behaviour. The prospect of useful fluid simulations became apparent after averaging over many simulation results using the same boundary conditions and forcing but different initializations of the gas.

LBM replaces the Boolean variables of LGCA [Wolf-Gladrow, 2000] with real-valued distribution functions $\boldsymbol{f_i(\vec{r})}$, representing the probability density of finding a particle in a certain region of the discretized phase-space. Another difference with respect to LGCA is the simplified collision operator [Bhatnagar et al., 1954].

Also, in contrast to LGCA, there are more choices of the underlying lattice. These are usually classified in the literature using the $\boldsymbol{D\alpha Q\beta}$-notation, where $\boldsymbol{\alpha}$ is an integer number denoting the space dimensionality and $\boldsymbol{\beta}$ is another integer indicating the number of discrete velocities (including the particle at rest) within the momentum discretization. Some restrictions still have to be fulfilled (especially Galilean and rotational invariance)[6] to ensure that a particular discretization can simulate the Navier-Stokes equations. Among the lattices in common use[7] there are the $\boldsymbol{D2Q9}$, $\boldsymbol{D3Q15}$, $\boldsymbol{D3Q19}$ and $\boldsymbol{D3Q27}$-models (see for example discussion in [He and Luo, 1997]). Since our primary interest was the $\boldsymbol{3D}$ case, we have chosen the $\boldsymbol{D3Q19}$ momentum discretization, which has better stability[8] than $\boldsymbol{D3Q15}$, while remaining less CPU-demanding than $\boldsymbol{D3Q27}$ at the price of negligible losses in accuracy. The discrete velocity directions for the $\boldsymbol{D2Q9}$ and $\boldsymbol{D3Q19}$ lattices are shown in Figs (1) and (2).

The macroscopic variables are defined as functions of the particle distribution func-

[6] A lattice with reduced symmetry can be (and has been) used, see [d'Humieres et al., 2001], where a $\boldsymbol{D3Q13}$-lattice is used. However, this approach also departs from the classical BGK-LBM dynamics.

[7] These are the lattices which satisfy these symmetry requirements; as an interesting side-note, the earlier LGCAs failed to recover the Navier-Stokes equations because an improper lattice was used (see [Succi, 2001], pp. 20-21). These models produced square vortices, which were clearly unphysical — one of the reasons why they were regarded as toy-models.

[8] The stability is determined using the von Neumann linear stability analysis (see for example [Wolf-Gladrow, 2000] for details of this procedure).

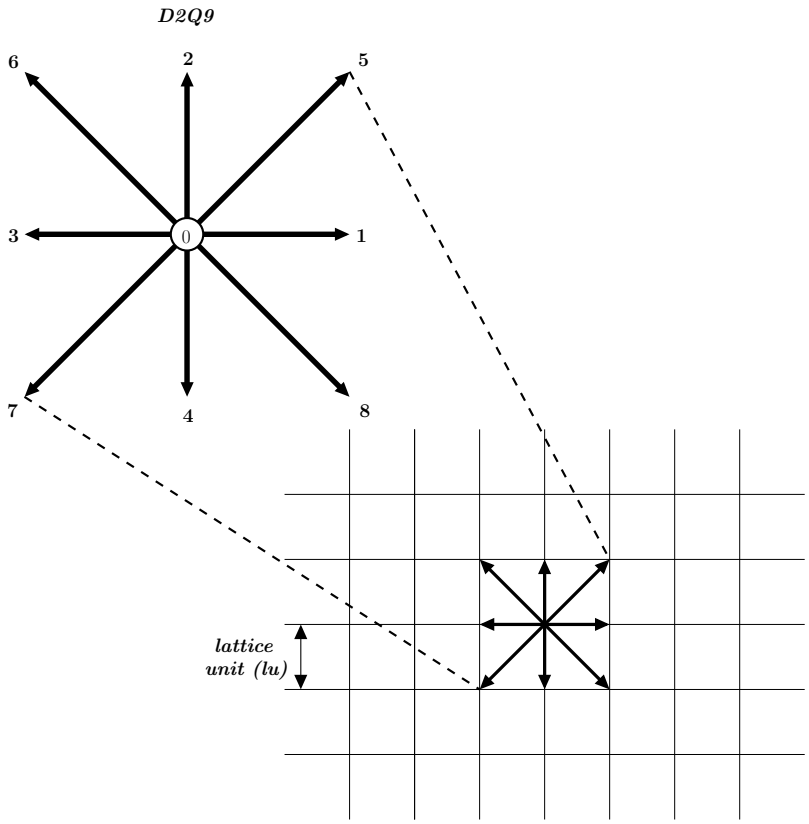

Figure 1: Discrete lattice velocities for the ***D2Q9*** model (adapted after [Sukop and Thorne, 2006])

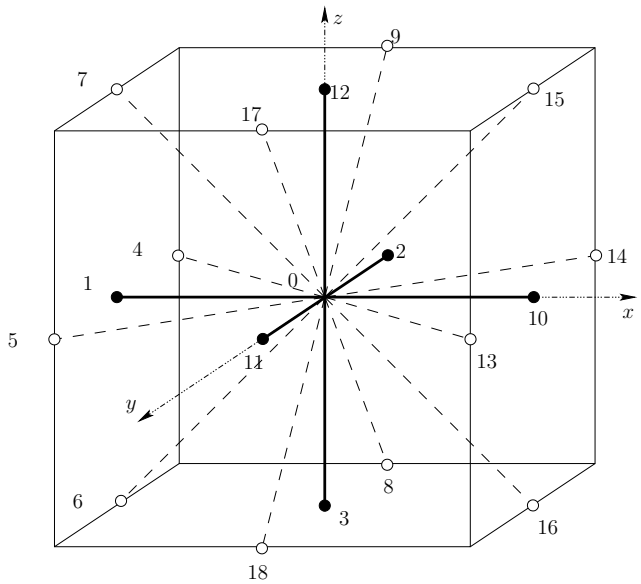

Figure 2: Discrete lattice velocities for the ***D3Q19*** model.

tions (hereafter DFs) according to:

$$\boldsymbol{\rho} = \sum_{\boldsymbol{i=0}}^{\boldsymbol{\beta-1}} \boldsymbol{f_i} \qquad \text{(macroscopic fluid density)} \tag{7}$$

and

$$\boldsymbol{\vec{u}} = \frac{\boldsymbol{1}}{\boldsymbol{\rho}} \sum_{\boldsymbol{i=0}}^{\boldsymbol{\beta-1}} \boldsymbol{f_i \vec{e}_i} \qquad \text{(macroscopic velocity).} \tag{8}$$

The DFs at each lattice point are updated using the equation:

$$\boxed{\underbrace{\boldsymbol{f_i(\vec{x} + \vec{e}_i \Delta t, t + \Delta t) = f_i(\vec{x}, t)}}_{\text{Streaming}} - \underbrace{\frac{[\boldsymbol{f_i(\vec{x}, t) - f_i^{eq}(\vec{x}, t)}]}{\boldsymbol{\tau}}}_{\text{Collision}},} \tag{9}$$

where $\boldsymbol{a} \in [\boldsymbol{0}, \boldsymbol{\beta - 1}]$ is an index spanning the (discretized) momentum space and $\boldsymbol{\tau}$ is a relaxation parameter, which is related to the fluid viscosity (more details about this will follow in this subsection).

This equation holds for lattice points within the fluid domain, but not for the domain boundaries, where boundary conditions compensate for the insufficient number of DFs (it does not make sense to stream DFs from walls towards the fluid). For this reason, the two steps (streaming & collision) are treated separately in actual numerical implementations.

The streaming step, where the DFs are translated to the neighbouring sites according to the respective discrete velocity direction, is illustrated in Fig. (3), in the $\boldsymbol{D2Q9}$ model for simplicity. The collision step (illustrated in Fig. [4]) consists of a re-distribution of the DFs towards the local discretized Maxwellian equilibrium DFs, but in such a way that local mass, momentum and energy are invariant, for reasons which were already explained in the previous subsection.

The EDFs can be obtained from the local Maxwell-Boltzmann SPDF (see for example [He and Luo, 1997]); they are

$$\boldsymbol{f_i^{eq}(\vec{x}) = w_i \rho(\vec{x}) \left[1 + 3 \frac{\vec{e}_i \cdot \vec{u}}{c^2} + \frac{9}{2} \frac{(\vec{e}_i \cdot \vec{u})^2}{c^4} - \frac{3}{2} \frac{\vec{u}^2}{c^2} \right],} \tag{10}$$

where for the $\boldsymbol{D2Q9}$ model the weights are

$$\begin{cases} \boldsymbol{w_{i=0}} = \frac{4}{9} \\ \boldsymbol{w_{i=\{1..4\}}} = \frac{1}{9} \\ \boldsymbol{w_{i=\{5..8\}}} = \frac{1}{36} \end{cases} \tag{11}$$

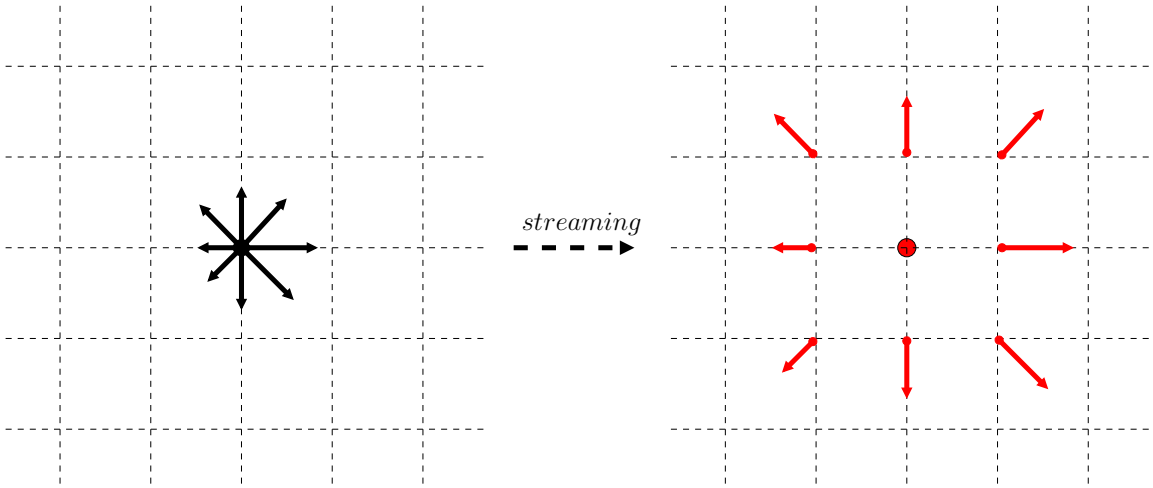

Figure 3: Illustration of the streaming process on a $\boldsymbol{D2Q9}$ lattice. Note that the magnitude of the DFs remain unchanged, but they move to a neighbouring node according to their direction.

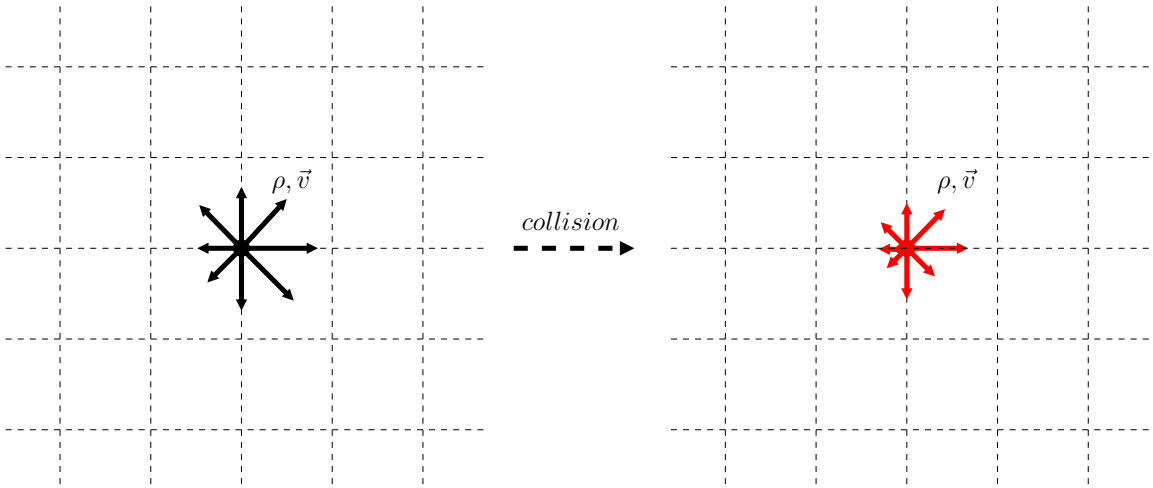

Figure 4: Illustration of the collision process on a $\boldsymbol{D2Q9}$ lattice. Note that the local density $\boldsymbol{\rho}$ and velocity $\boldsymbol{\vec{v}}$ are conserved, but the DFs change according to the relaxation-to-local-Maxwellian rule.

and $\boldsymbol{c}$ is the propagation speed on the lattice $(\mathbf{1}\textit{lattice spacing}/\textit{time step})$, taken as $\boldsymbol{c} = \mathbf{1}$ in most cases. For the $\boldsymbol{D3Q19}$ model, the weights are changed as follows:

$$\begin{cases} \boldsymbol{w_{i=0}} = \frac{\mathbf{1}}{\mathbf{3}} \\ \boldsymbol{w_{i=\{1..6\}}} = \frac{\mathbf{1}}{\mathbf{18}} \\ \boldsymbol{w_{i=\{7..18\}}} = \frac{\mathbf{1}}{\mathbf{36}} \end{cases} . \tag{12}$$

Under the afore-mentioned assumption of a low Mach number, and further taking $\boldsymbol{Kn}$[9], $\boldsymbol{\delta_t}$[10], $\boldsymbol{\delta_x}$[11] $\boldsymbol{\rightarrow 0}$, this model recovers the incompressible Navier-Stokes equations:

$$\boldsymbol{\nabla \cdot \vec{u} = 0} \ , \tag{13}$$

$$\boldsymbol{\rho \partial_t \vec{u} + \rho \vec{u} \nabla \cdot \vec{u} = -\nabla P + \rho \nu \nabla^2 \vec{u}} \tag{14}$$

with an isothermal equation of state:

$$\boldsymbol{P = c_s^2 \rho} \ , \tag{15}$$

where $\boldsymbol{P}$ is the pressure.

The viscosity of the fluid is related to the relaxation parameter $\boldsymbol{\tau}$ by the equation

$$\boldsymbol{\nu = c_s^2 \left(\tau - 1/2\right) \Rightarrow \tau = \frac{\nu}{c_s^2} + \frac{1}{2} \xrightarrow[c_s^2|_{D3Q19}=1/3]{} \tau_{D3Q19} = 3\nu + \frac{1}{2}} \tag{16}$$

The proof of these results follows from the Chapman-Enskog analysis, and is included in 3.4. The speed of sound $\boldsymbol{c_s}$ is a lattice-dependent quantity, which takes the value

$$\boldsymbol{c_s = \frac{1}{\sqrt{3}}}$$

for the popular $\boldsymbol{D2Q9}$ and $\boldsymbol{D3Q19}$ lattices.

Eq. (16) provides a straightforward method for adjusting the fluid viscosity in the model. It is obvious that $\boldsymbol{\tau \geq 0.5}$ is required in order to ensure a positive viscosity. The limit $\boldsymbol{\tau \rightarrow 0.5}$ corresponds to the inviscid flow, while the $\boldsymbol{\tau \rightarrow \infty}$ limit represents the Stokes (creeping) flow. While the later case poses no difficulty to the model[12], the former limit is problematic because stability issues appear if an insufficient lattice resolution is chosen. This is due to the fact that velocity gradients can become very large (especially in complex geometries, with high topography variations) and the model cannot dissipate the energy due to the very low viscosity[13]. Unfortunately, many of the flows of practical interest are turbulent, often with a low viscosity.

[9] The assumption of $\boldsymbol{Kn \equiv \frac{\lambda}{L} \rightarrow 0}$ is a requirement for continuum models to apply, hence it is not specific to LBM.

[10] Physical time unit, not the computational time unit $\boldsymbol{\Delta t}$, which is usually taken as $\boldsymbol{1}$.

[11] Physical space unit $\boldsymbol{\neq \Delta x}$, which is usually taken as $\boldsymbol{1}$.

[12] Strictly speaking, the Stokes limit presents its own issues, namely the slow convergence rate; however, in the present context, we concentrate on stability.

[13] The "brute-force" remedy of increasing the grid size improves the situation by effectively rescaling the velocity field, which automatically diminishes velocity gradients. However, this approach quickly becomes unfeasible.

To overcome this limitation, a turbulence model is necessary. The role of this procedure is to parameterize the turbulent energy dissipation in turbulent flows, where the larger eddies extract energy from the mean flow and ultimately transfer some of it to the smaller eddies which, in turn, pass the energy to even smaller eddies, and so on up to the smallest scales, where the eddies convert the kinetic energy into internal energy of the fluid. At this scales (also known as Kolmogorov scale), the viscous friction dominates the flow [Frisch, 1996].

In classical LB applications, a convenient method of modelling turbulent dissipation is through a locally-enhanced collision, which effectively stabilizes the simulation. A common (see [Thuerey, 2007]) procedure is the Smagorinsky sub-grid model [Smagorinsky, 1963], which, when adapted to LBM, consists of:

1. **evaluation of the local stress tensor**:

$$\boldsymbol{\Pi_{\alpha,\beta}} = \sum_{\boldsymbol{i=0}}^{\boldsymbol{\beta-1}} \boldsymbol{\vec{e}_{i,\alpha}\vec{e}_{i,\beta}(f_i - f_i^{eq})} \ , \tag{17}$$

   where $\boldsymbol{(\alpha,\beta) \in \{x,y,z\} \times \{x,y,z\}}$;

2. **computation of the enhanced relaxation time**:

$$\boldsymbol{\tau_S = 3(\nu + C^2 S) + \frac{1}{2}} \ , \tag{18}$$

$$\text{where: } \boldsymbol{S \equiv \frac{1}{6C^2}\left(\sqrt{\nu^2 + 18C^2\sqrt{\Pi_{\alpha,\beta}\Pi_{\alpha,\beta}}} - \nu\right)} . \tag{19}$$

   Proper values for the Smagorinsky constant $\boldsymbol{C}$ that are suitable for LBM have been published in the literature [Yu et al., 2005] and found to be close to $\mathbf{0.03}$. It can be observed that, as $\boldsymbol{S > 0, \forall f_i}$, the effect of the model will always be a higher local effective viscosity, which increases as the local stresses increase.

To conclude this discussion, it is worth mentioning that this procedure does indeed complicate the collision operator, losing some of the elegance of the LBM algorithm. However, the efficiency is increased because it allows one to work on much coarser grids compared to the original LBM, at the same value for viscosity [Thuerey, 2007].

Multi-Relaxation-Times (MRT) LBM. It is also worth mentioning that there exists another formulation of LBM, which uses instead of the simple BGK relaxation a set of relaxation times for the different hydrodynamic moments of the DFs [d'Humieres et al., 2002].

There is some freedom in choosing these multiple relaxation times, which can be used to stabilize the simulation. While this is a promising approach, this is out of the scope of the present thesis paper. This topic would be the one of the natural paths to investigate further.

2.3 Boundary conditions

Boundary Conditions (BC) form an important part of any numerical solution, as they can often affect the accuracy of the algorithm significantly. Thus, an introduction to LBM would not be complete without an exposition of the currently known methods and best practices for imposing the appropriate constraints on the fluid domain.

Periodic Boundary Conditions (hereafter BCs). The simplest type of boundary condition is the periodic one. In this case, the domain becomes folded along the direction of the periodic BC pair. From the perspective of ocean modelling, this kind of BC is, of course, only useful in preliminary tests, as it implies a high symmetry of the flow domain. Due to the way our streaming operator is implemented (that is, through circular shifts of the DF-arrays), this kind of BC does not require any special treatment (because from the point of view of the model there is no domain discontinuity) and is therefore also computationally the cheapest.

No-slip BCs. The most often used type of BC in LBM flows is the no-slip BC, especially the simple bounce-back rule, which is quite elegant and surprisingly accurate in most common applications. The basic idea is that the incoming DFs at a wall node are reflected back to the original fluid nodes, but with the direction rotated by π radians.

The bounce-back BC is one of the most advertised benefits of the Lattice Boltzmann method, as it is trivial to implement and it allows one to effortlessly introduce obstacles into the fluid domain (for example, by using a global Boolean field; bounce-back is performed for all of the wall lattice points, whose flag was "flipped" to solid during the initialization stage). However, the BC has been proven to be only first-order accurate in time and space [Pan et al., 2006]. A straightforward improvement is to consider the wall-fluid interface to be situated halfway between the wall and fluid lattice nodes

[Ziegler, 1993]. This simple translation (which is actually nothing more than a slightly different post-processing procedure), commonly referred to as half-way bounce-back in the literature, is illustrated in Fig. (5). However, even for this scheme the accuracy becomes first-order when the fluid boundaries are tilted with respect to the lattice directions [Cornubert et al., 1991] and more advanced interpolation schemes are necessary. It is important to note that, in actual numerical implementation, the streaming step is also performed at the solid nodes marked as no-slip, but at these points the collision step is replaced by the bounce-back procedure. Nonetheless, it does not make sense to evaluate the macroscopic fields at these points, as they are in reality situated outside the fluid domain.

Slip BCs. The slip BC is similar to the no-slip one, except that the DFs are reflected in a mirror-like fashion instead of bounce-back. A clear treatment of this can be found in [Succi, 2001].

Velocity and Pressure BC. Sometimes, we need to be able to model flows with prescribed velocity or pressure profiles; since LBM operates from the point of view of the particle distribution function, these are often referred to as von Neumann and Dirichlet BCs respectively. Such constraints are necessary, for example, when the simulation domain communicates with other, not simulated but parameterized flow domains. The prescribed velocity or pressure add **2** (in $\boldsymbol{D2Q9}$ — **3** in $\boldsymbol{D3Q19}$) and, respectively **1** equation for determining the unknown DFs (which would hypothetically have to come from "outside" the fluid domain). In the case of the velocity BC, the additional equations are actually enough to solve for the unknown DFs (in $\boldsymbol{D2Q9}$ only, but not in $\boldsymbol{D3Q19}$); for the pressure BC, the system of equations is not closed for all lattices, and additional constitutive equations are necessary. A central idea in this direction is that of the bounce-back of the non-equilibrium part of the DFs in the normal direction (also known as the Zou-He assumption after its proponents [Zou and He, 1997]). Hence, for a northern boundary (on a $\boldsymbol{D2Q9}$-lattice), one would write:

$$\boldsymbol{f_2 - f_2^{eq} = f_4 - f_4^{eq}} \tag{20}$$

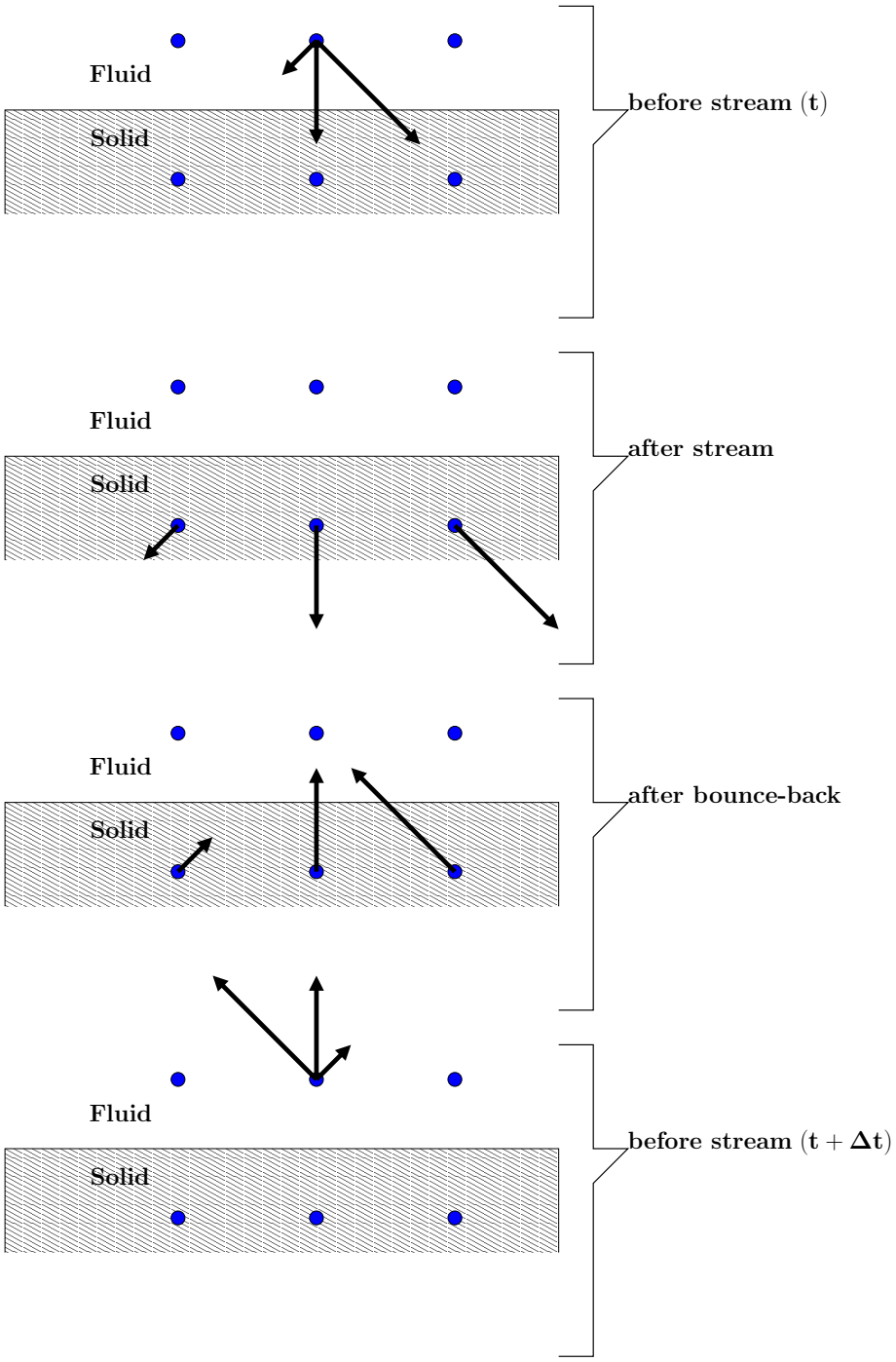

Figure 5: Illustration of the half-way bounce-back algorithm for the ***D2Q9*** model (adapted after [Sukop and Thorne, 2006]). Note that the collision step is replaced by the bounce-back step at the solid nodes in the immediate vicinity of the fluid domain. Information from these nodes is also discarded during post-processing, as the wall position is taken as half-way between them and the adjacent fluid nodes.

While these considerations are sufficient in $\boldsymbol{2D}$, problems arise in $\boldsymbol{3D}$, where there are $\mathbf{5}$ unknown DFs and the system of equations is again not closed, even including the Zou-He assumption. Additional constraints have been proposed [Chen and Martinez, 1996], but these are rather "symptomatic" fixes. A very accurate, recent approach to this problem is due to [Latt and Chopard, 2008] (also known as regularized BC). The authors first apply the bounce-back of off-equilibrium parts for all of the unknown DFs, then evaluate the local stress tensor (see Eq. (17)), and ultimately use this to compute new values for all of the DFs, according to the rule:

$$\boldsymbol{f_i^{reg} = f_i^{eq}(\rho, \vec{u}) + \frac{w_i}{2c_s^4} Q_i : \Pi^{(1)}, \quad i \in [0, \beta - 1]} \ , \tag{21}$$

where ":" stands for the contraction of two tensors[14] and the tensor $\boldsymbol{Q_i}$ is defined as:

$$\boldsymbol{Q_i \equiv \vec{e}_i \vec{e}_i - c_s^2 I} \tag{22}$$

The expression for the corrections are derived by the authors through a rigorous multiscale expansion and also found to be quite accurate in actual numerical simulations.

[14] Which produces a scalar from any two tensors of same dimensions (say $\boldsymbol{T}$ and $\boldsymbol{U}$), according to the rule $\boldsymbol{T : U \equiv \sum_{i=1}^{3} \sum_{j=1}^{3} T_{ij} U_{ji}}$.

2.4 Numeric implementation

2.4.1 Outline of the algorithm

The basic algorithm is listed below:

Initialize Macroscopic Quantities ($\boldsymbol{\rho}, \vec{\boldsymbol{u}}$)
Compute Equilibrium DFs ($\boldsymbol{f_i^{eq}}$) based on initial Macroscopic Fields

Initialize DFs with Equilibrium DFs
for $\boldsymbol{tStep}$ = 1 to tMax **do**
  Boundaries
  Compute Macroscopic Quantities ($\boldsymbol{\rho}, \vec{\boldsymbol{u}}$)
  Compute Equilibrium DFs ($\boldsymbol{f_i^{eq}}$)
  Local Collision Step
  Streaming Step
end for

2.4.2 General principles of the implementation

When it comes to the actual implementation, there are many ways of translating the theoretical algorithm into computer instructions. We present here some of the general ideas particular to our own implementation.

First of all, the computational domain consists of a regular, 3D Cartesian grid. This implies that any system to be simulated should be inscribed within such a domain. Depending on the number of divisions along each side of the 3D rectangle, the spatial resolution can be adjusted. At the current stage of the implementation, isotropic space spacing is used, that means

$$\boldsymbol{\Delta x \equiv \Delta y \equiv \Delta z} \ . \tag{23}$$

In principle, at each point of the rectangular computational domain, we hold an integer value representing the type of the cell (which can be fluid, fluid boundary —hereafter fluidB — or solid/air) and several (19 in the $\boldsymbol{D3Q19}$-model) floating-point values for the

discrete DFs. However, since the DFs are only useful for the fluid and the fluidB nodes[15], it would be a waste of computer memory to store them all at each lattice point. We circumvent this problem by implementing a dynamic memory allocation scheme for the DFs at each lattice node, which allows us to not allocate memory where it is unnecessary (specifically for solid/air cells which play no role in the simulation). The fluidB cells play a similar role to the "ghost layer" described in other implementations [Succi, 2001].

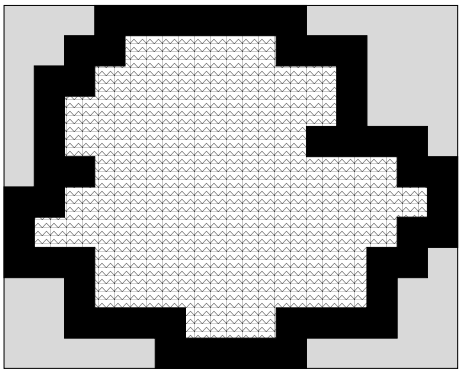

Figure 6: Sample irregular domain (***2D*** for clarity). Fluid is hashed, boundary layer is black and solid regions (for which no memory is allocated to store the discretized DFs) are gray.

Once this datastructure has been set for the particular geometry to be simulated (in the `init` subroutine), the actual LBM algorithm is applied to the lattice.

2.5 Refinements of the model

As described up to now, the model can already simulate some simple flows. However, for more complex simulations, additional improvements are required. Those which have been already implemented are described below. Others, which await to be added, are discussed in the concluding chapter of the present thesis.

2.5.1 Increasing the accuracy of our numerical implementation

Experimentation with the model showed slight non-conservation of the total mass of the system over large integration times. Further investigations revealed that the problem

[15]The air is not directly simulated, but its effect of forcing the ocean surface are included through a tangential velocity-type boundary condition.

consisted of an incorrect calculation of the equilibrium DFs. While this is more of a numerical effect than a theoretical observation, we will develop this further as we consider it to be important nonetheless, due to the major improvement observed in the subsequent model runs.

Essentially, the problem was a roundoff error encountered due to the addition of 2 numbers of very different orders of magnitude. In Eq. (10), the second and third terms inside the brackets are much smaller than **1** (by definition, the model requires them to be $\sim \boldsymbol{Ma} \ll \mathbf{1}$). This implies that they have a very different mantissa in the floating-point representation. Hence, upon addition, the computer truncates the result, effectively diminishing the influence of the smaller term.

The remedy [Skordos, 1993] consists of reformulating the model such that only the fluctuations are manipulated. Therefore, one can define:

$$\boldsymbol{h_i} \equiv \boldsymbol{f_i} - \boldsymbol{w_i \rho_0} \ , \tag{24}$$

where $\boldsymbol{\rho_0}$ is a reference density; the equivalent LBM equation reads

$$\boldsymbol{h_i(\vec{x} + \vec{e_i}\Delta t) = h_i(\vec{x} + \vec{e_i}\Delta t) - \frac{[h_i(\vec{x}, t) - h_i^{eq}(\vec{x}, t)]}{\tau}} \ , \tag{25}$$

where the macroscopic variables are recovered from the modified DFs through

$$\boldsymbol{\rho = \rho_0 + \sum h_i \ \ ; \ \ \vec{u} = \frac{\sum h_i \vec{e}_i}{\rho_0 + \sum h_i}} \tag{26}$$

and with the modified equilibrium DFs

$$\boldsymbol{h_i^{eq} = w_i(\rho - \rho_0) + w_i \rho_0 \left[3\frac{\vec{e_i} \cdot \vec{u}}{c^2} + \frac{9}{2}\frac{(\vec{e_i} \cdot \vec{u})^2}{c^4} - \frac{3}{2}\frac{\vec{u}^2}{c^2} \right]} \ . \tag{27}$$

Since this is mainly an algorithmic modification, we did not formulate the theory in terms of the new variables. However, the interested reader should be aware of the fact that this small artifice can have far-reaching implications in the actual numerical implementation, where many of the steps (specifically: macroscopic variable computation, collision and also the boundary conditions) have to be devised by taking this issue into consideration. Switching from a **32**-bit to a **64**-bit computer system did not eliminate the problem. For more details on this topic, see for example [Dahlquist and Björck, 2008].

2.5.2 Proper treatment of the forces

An important aspect when long-term integrations are desired is a proper introduction of body forces into the model. The simplest approach [Sukop and Thorne, 2006] is to add another term in the expression of the macroscopic velocity $\vec{u}$ which is then used during the evaluation of the equilibrium DFs, as follows

$$\vec{u}^* = \vec{u} + \frac{\tau \vec{F}}{\rho} , \tag{28}$$

where $\vec{F}$ is the applied body force.

While simple, this approach is known to be unstable from the work of Wolf-Gladrow [Wolf-Gladrow, 2000]. Moreover, Guo et. al. [Guo and Zhao, 2002] proved this approach to be correct only in the limiting case, when the force has very low spatial and temporal variations (practically - when the body-force is constant): if this is not true, the Navier-Stokes equation with a body force is not recovered, but another (unphysical) equation is solved instead.

According to [Guo and Zhao, 2002], the correct force treatment involves:

1. Modifying the LBM evolution equation (9) into:

$$\boxed{\underbrace{f_i(\vec{x} + \vec{e_i}\Delta t, t + \Delta t) = f_i(\vec{x}, t)}_{\text{Streaming}} - \underbrace{\frac{[f_i(\vec{x}, t) - f_i^{eq}(\vec{x}, t)]}{\tau}}_{\text{Collision}} + \underbrace{\Delta t \xi_i}_{\text{Forcing}} ,} \tag{29}$$

   where ξ_i are direction-specific forcing terms, computed from the actual force [16] $\vec{F}$, defined as

$$\xi_i = \left(1 - \frac{1}{2\tau}\right) w_i \left[\frac{\vec{e_i} - \vec{u}}{c^2} + \frac{(\vec{e_i} \cdot \vec{u})}{c^4} \cdot \vec{e_i}\right] \cdot \vec{F} . \tag{30}$$

2. Modifying the formula for calculating the local fluid momentum as:

$$\vec{u} = \frac{1}{\rho} \left(\sum_{i=0}^{\beta-1} f_i \vec{e_i} + \frac{\Delta t}{2} \vec{F} \right) , \tag{31}$$

   and leaving the density evaluation formula unchanged.

A delicate topic in this context is related to the proper introduction of the Coriolis force: due to the fact that this force is velocity dependent, the values of the velocity at

[16] This force should be scaled accordingly to the lattice dimensions.

the next timestep are needed to evaluate the forces at the current timestep, essentially rendering the method implicit in theory. [Wolf-Gladrow, 2000] proposed a method to bypass this limitation using a predictor-corrector step, obtaining promising results.

2.6 Validation cases and preliminary results

Novel CFD methods are only taken seriously if they are able to reproduce some standard benchmark flow problems. These are cases when the solutions have been extensively investigated, either analytically or numerically, using more traditional methods. These test problems are, of course, not fixed, as they are chosen to stress the kind of simulations that the new model is supposed to address. We have chosen a problem from classical CFD (Poiseuille flow) to quantify the accuracy of the plain LB algorithm and we also illustrate some turbulent flow simulations which can only be performed once the Smagorinsky turbulence model is enabled.

2.6.1 $3D$ Poiseuille flow

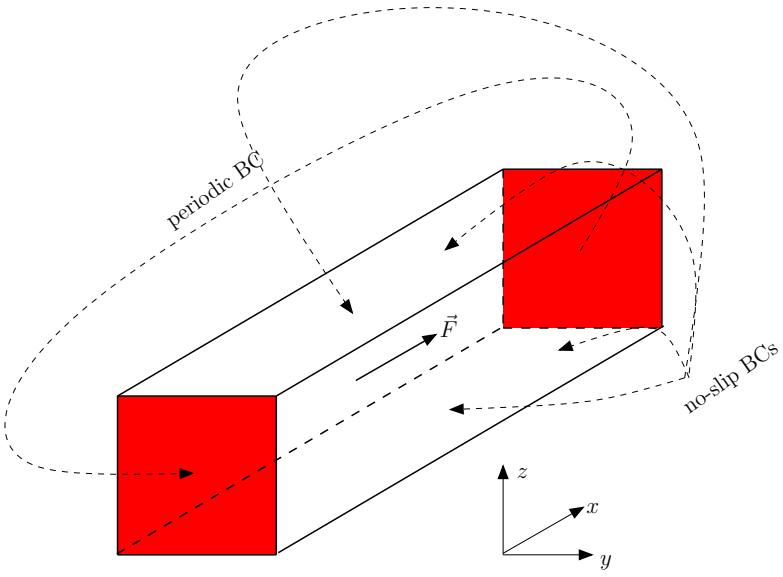

Figure 7: Geometry for $\mathbf{3D}$ Poiseuille problem. For the rectangular computational domain, the periodic BCs are highlighted in red; no-slip BCs were imposed for the rest of the faces.

Description of the case. The geometry of the problem is shown schematically in Fig. (7). Along the $\boldsymbol{X}$-direction, periodic boundary conditions were applied, while for all the remaining faces of the computational domain no-slip boundary conditions were used. This geometry is commonly referred to as the 3D Poiseuille flow, in analogy to the classical 2D, steady-state flow between two infinite planes. The number of grid points along the $\boldsymbol{X}$ direction was fixed to **xMax=10**. Due to the periodic boundaries, this number should not influence the flow.

Initially, the entire fluid volume is at rest. A constant and homogeneous body force along the $\boldsymbol{X}$-direction is then applied, and the Lattice Boltzmann algorithm is applied repeatedly until a steady-state is reached. The non-dimensional kinematic viscosity of the flow was fixed at $\boldsymbol{\nu = 0.14}$, the non-dimensional magnitude of the body force was $\boldsymbol{F_x = |\vec{F}| = 0.1}$, and the non-dimensional cross-section was $\mathbf{1 \times 1}$. Based on these numbers, the analytical solution for the velocity at the center of the channel was evaluated.

For each numerical experiment, we have taken $\boldsymbol{\delta_t = \delta_x^2}$ for reasons of numerical accuracy (this choice guarantees both the stability of the simulation and the recovery of the incompressible Navier-Stokes equations — see [Latt, 2008]). Based on a given grid resolution in the $\boldsymbol{Y}$ and $\boldsymbol{Z}$ directions (these were the same), we computed the scaled force term using

$$\boldsymbol{f_{LBM} = \frac{F_x}{yMax^3}} \, . \tag{32}$$

As the timestep scales with $\boldsymbol{\delta_x^2}$, for any two experiments we also adjusted the number of iterations to simulate the same non-dimensional time using

$$\boldsymbol{N_{it,2} = N_{it,1} \left(\frac{yMax_2}{yMax_1} \right)^2} \, . \tag{33}$$

From the steady-state solution, the maximum value on the line with **y=yMax/2**, **z=zMax/2** is taken and re-scaled to non-dimensional units using

$$\boldsymbol{u}_{\text{numeric, non-dimensional}} = \boldsymbol{yMax} \times \boldsymbol{u}_{\text{numeric,lattice units}} \, . \tag{34}$$

The result was compared with the analytic solution.

Then, the number of grid points along the $\boldsymbol{Y}$ and $\boldsymbol{Z}$-direction were changed (keeping $\boldsymbol{yMax = zMax}$), and the procedure was repeated.

Analytic solutions. [Krueger et al., 2009] presented the following series solution for the problem:

$$u_x(y,z) = \frac{\hat{u}_x}{\Sigma}\left[z(H-z) - \frac{8H^2}{\pi^3}\sum_n^{odd}\frac{1}{n^3}\frac{\cosh(n\pi(y-W/2)/H)}{\cosh(n\pi W/2H)}sin\left(\frac{n\pi z}{H}\right)\right] \tag{35}$$

where **H=zMax-2**, **W=yMax-2**, $\hat{u}_x = \Sigma\frac{F_x}{2\nu}$ is the velocity at the center of the channel with $\vec{F} = (F_x, 0, 0)$=body force acting on the fluid. The expression for this term stems from the fact that, at steady-state, the pressure gradient is canceled by the viscous shearing at the center of the channel:

$$\frac{\partial p}{\partial x} = F_x = -2\nu\frac{\hat{u}}{\Sigma} \ .$$

$$\Sigma = \frac{H^2}{4} - \frac{8H^2}{\pi^3}\sum_n^{odd}\frac{1}{n^3}\frac{\sin(n\pi/2)}{\cosh(n\pi W/2H)}$$

is a geometry-dependent normalization factor.

In practice, the series has to be truncated. We have chosen to stop the summation when terms of the sum become smaller than $\mathbf{10^{-8}}$.

Comparison of semi-analytic and numeric results

xMax	yMax	zMax	$\hat{u}_{\text{theoretical}}$ [$\times 10^{-2}$]	$\hat{u}_{\text{numeric}}$ [$\times 10^{-2}$]	relative error in $\hat{u}$ [%]
10	**20**	**20**	**5.2622**	**5.2493**	**0.245**
10	**30**	**30**	**5.2622**	**5.2562**	**0.114**
10	**40**	**40**	**5.2622**	**5.2586**	**0.068**
10	**50**	**50**	**5.2622**	**5.2598**	**0.046**
10	**60**	**60**	**5.2622**	**5.2603**	**0.036**

The relative error ϵ scales approximately $\sim \delta_x^2$, as it can be observed in Fig. [8].

We also present the simulated steady-state mid-x-section velocity profile in Fig. [11].

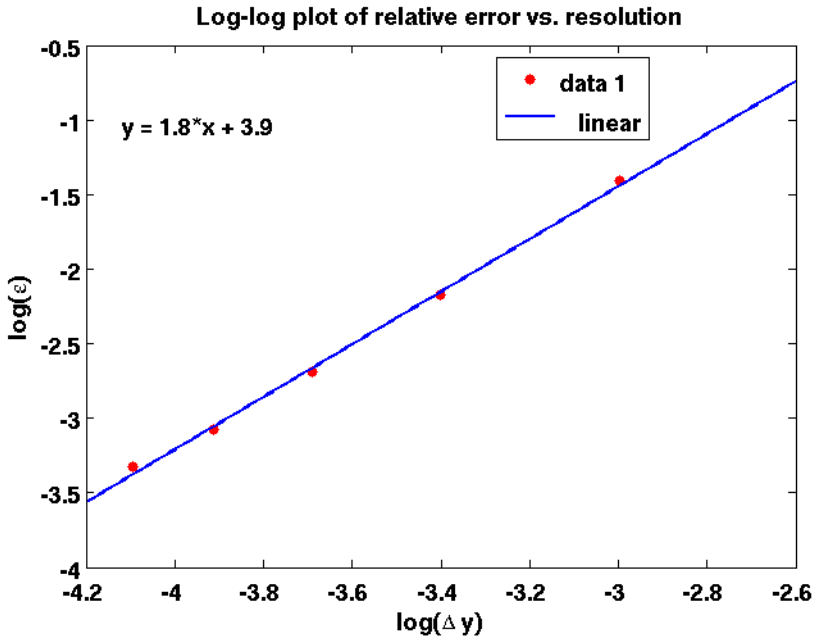

Figure 8: Plot of the relative error for midpoint velocity $\boldsymbol{\epsilon}$ vs. square of resolution $\boldsymbol{\delta_x^2}$. The linear trend was determined using standard least-squares regression.

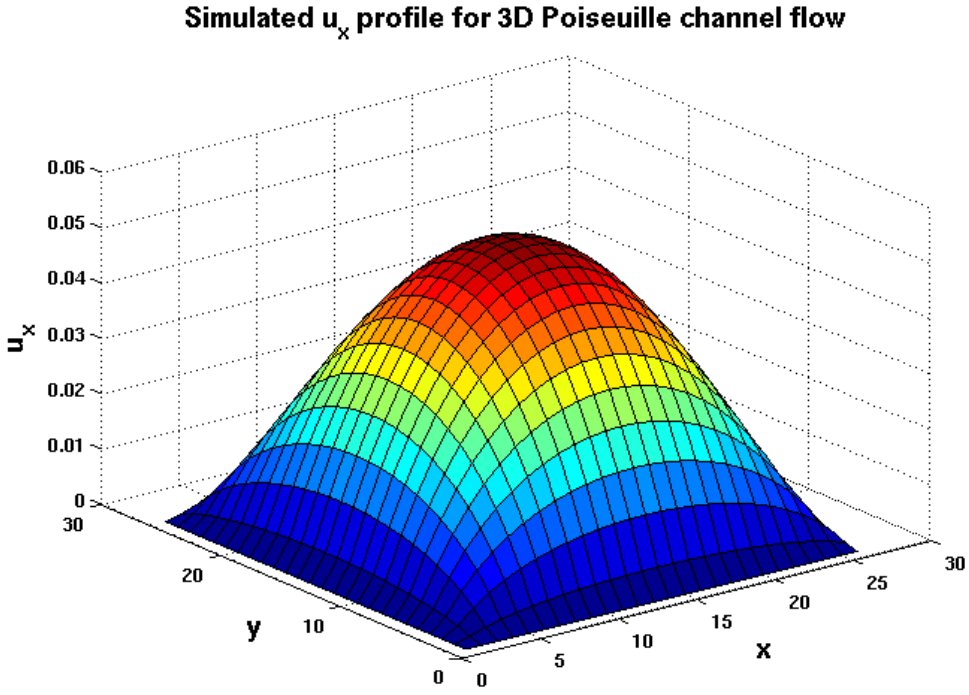

Figure 9: Surface plot of the $\boldsymbol{U_x}$ profile for the 3D Poiseuille flow problem at steady-state. The color map is also presenting the magnitude of the $\boldsymbol{U_x}$ velocity, for better observation.

2.6.2 Turbulent $3D$ lid-driven cavity flow

In the second example, the turbulent flow in a closed cavity was simulated. This was not possible without the introduction of the Smagorinsky turbulence model that was previously described. The geometry of the case is described in Fig. [10]; all of the walls of the cubic fluid domain were set to no-slip (half-way bounce-back), except the top wall, on which a constant tangential velocity along the $\boldsymbol{X}$-direction was given (sliding lid), according to the BC described in [Latt and Chopard, 2008]. For visualisation purposes, a number of tracers were passively advected by the flow, using a simple, $\mathbf{1}^{\boldsymbol{st}}$-order Eulerian scheme. Since the advection scheme for the tracers needs information on the fluid velocity at off-lattice sites also, we used the trilinear interpolation algorithm to obtain a piece-wise continuous approximation for each velocity component from the LB solver.

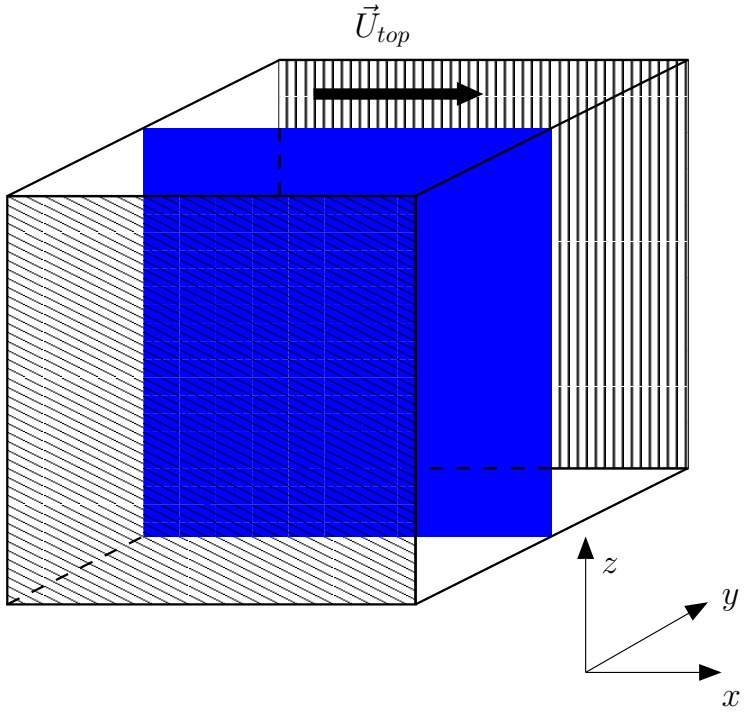

Figure 10: Geometry for $\mathbf{3D}$ lid-driven cavity. No-slip BCs used for all walls, except the top one, which is a velocity BC. In blue, the section along which the passive tracers were advected is shown.

In Fig. [11], we present $\mathbf{4}$ snapshots[17] of the passive tracer field for the problem. The

[17] ordered from left to right and from up to down, based on the time

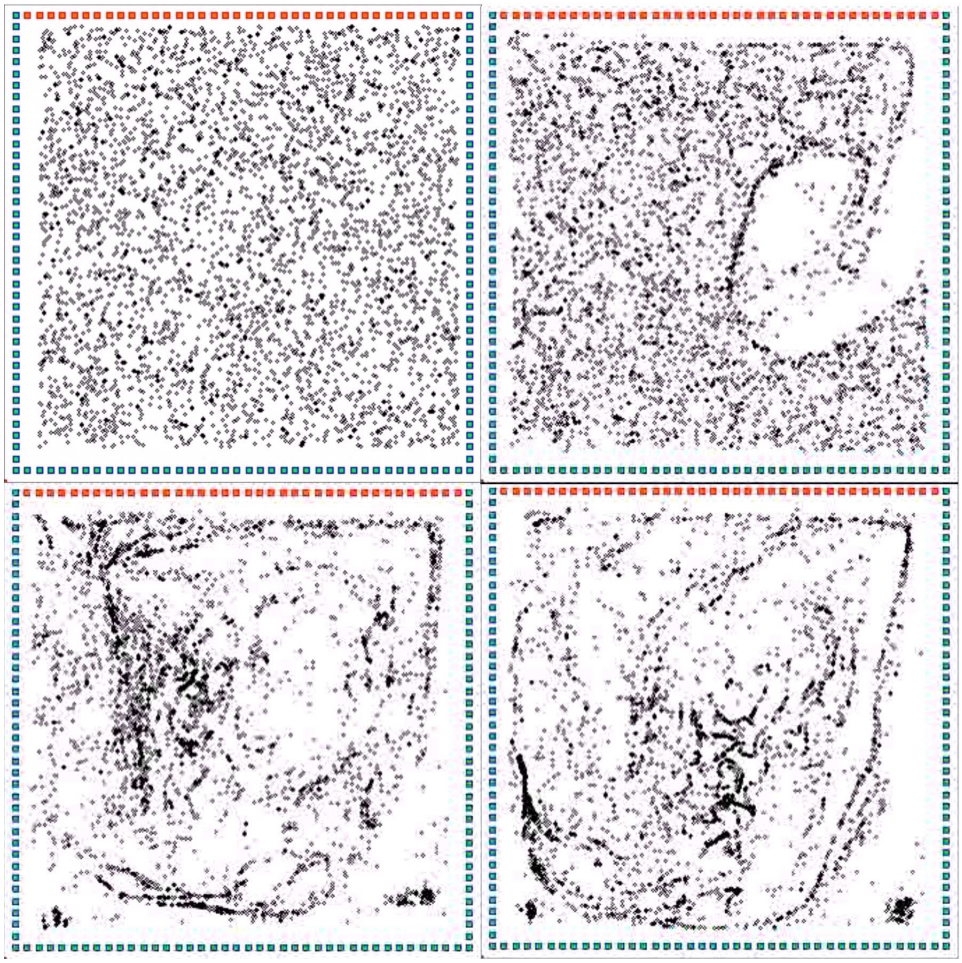

Figure 11: Snapshots at $\boldsymbol{t = 0}$ (upper-left), $\boldsymbol{t = 750}$ (upper-right), $\boldsymbol{t = 1500}$ (lower-left) and $\boldsymbol{t = 2250}$ (lower-right) of the passive tracers on a sectioning plane parallel to the $\boldsymbol{X}$-axis. The Reynolds number of the simulation is $\boldsymbol{Re \approx 1.7 \times 10^6}$; the red points represent the sliding wall, while the blue ones represent the no-slip walls of the cavity.

simulation becomes stable due to the locally-enhanced viscosity coefficient. However, the fluid dynamics remains turbulent even for long simulation times, which reflects the fact that LBM with the Smagorinsky parameterization forms essentially an LES solver (see discussion at the beginning of 3).

3 Reynolds-averaged Lattice Boltzmann Method

3.1 Turbulence modelling for the Navier-Stokes equations

The dynamics of most fluids encountered in nature is described by the Navier-Stokes equations. As already discussed, due to the nonlinear nature of these equations, the only viable method of obtaining their solution for non-trivial geometries is to recurse to numerical methods. The continuous PDEs are then solved on a discretized spatial mesh (for example finite differences, finite volumes or finite elements) and at discrete time points. Even this approach is not devoid of difficulties, as very often the dynamics is highly turbulent. It has been shown [Canuto, 1994] that the computational effort scales as a polynomial in the Reynolds number, because the turbulent scales become smaller as the degree of turbulence increases. This result comes from Kolmogorov's theory of turbulence (see for example [Frisch, 1995]). It is important in the present context to remind the reader the main picture emerging from this theory. Kolmogorov taught us that the energy is first introduced through a series of large eddies. These, in turn, transfer energy with almost negligible dissipation to smaller eddies. The process is repeated for smaller-and-smaller eddies, but as the characteristic length-scale decreases the Reynolds number $\boldsymbol{Re} = \boldsymbol{UL}/\boldsymbol{\nu}$ increases and the viscous effects gain importance and eventually dominate the flow at very small scales[18].

One can classify the wide variety of numerical methods for solving the NSE by the approach they use to deal with the description of turbulence. These classes of methods are (roughly) listed in Table [1], and briefly described below:

- Direct Numerical Simulation (DNS): is conceptually the simplest approach in that it tries to resolve all of the turbulent motions in the fluid. While this method is a useful tool in the exploration of turbulence and can also be used to study small-scale flows, it is unfortunately very costly, and even on today's most powerful supercomputers it can only reach a very low Reynolds number.

- Large Eddy Simulation (LES): only attempts to resolve the large (energy-containing) eddies, and includes the effects of smaller eddies only indirectly, through a set of

[18]These are also called Kolmogorov or *dissipative* length-scales

parameterizations which describe the rate of energy dissipation in the fluid. The methodology is derived by applying a spatial filter on the governing fluid equations [Hou et al., 1994]. This is a useful approach when some of the effects of the eddies are of interest, but when the entire energy cascade is not the focus of the study or is too costly. Hence, LES can be viewed as an intermediate approach - a trade-off between flow details and computational resources. However, this approach is still not popular in environmental fluid dynamics because LES lose their computational efficiency in the case of stably stratified fluids, as is often the case in the ocean: due to the stability of the fluid, the largest, energy-containing eddies are still rather small. LES nonetheless will resolve them, leading to a spatial resolution that, even on today's fastest supercomputers, is not viable for global circulation models.

- Reynolds-averaged Navier-Stokes (RANS): represents in a sense the opposite of DNS, in that turbulent motion is not resolved at all, but only the evolution of the averaged fields is studied. One has to be very careful in choosing the definition of the averaging procedure. The exact formulation is in terms of *ensemble averages*, where the macroscopic fields (velocity, density or pressure) obtained through a RANS simulation represent the average at a certain time of many realizations of the flow configuration, all subject to the same initial and boundary conditions, and subject to the same body forces. The link with real fluid flows is provided by the *ergodic theorem*, which states that the ensemble averages can be interpreted as time averages if the timescales to be resolved are much larger than the timescales over which one performs the averaging [Huang, 1987]. The entire turbulent behavior is not resolved with this approach, as turbulence appears only through parameterizations. Thus, RANS cannot teach us anything about the turbulence itself, as inevitably relies on experiments, analytic studies or finer (and, consequently, smaller scale) numerical studies which prescribe the actual parameterization that are used.

	Traditional CFD	Lattice Boltzmann Methods
DNS	Yes	Yes
LES	Yes	Yes
RANS	Yes	**Not previously attempted (our main contribution)**

Table 1: Rough classification of the majority of computational fluid dynamics numerical methods

We concentrated in the present study on applying the RANS method to oceanographic flows. Despite it's limitations, this approach is a reasonable one in our present case because we are often interested in the evolution of the flows on timescales of weeks to decades. Furthermore, there have been many studies quantifying aspects of the turbulence, and several parameterizations are especially popular and used in forecast models with reasonable prediction power [Palmer and Hagedorn, 2006]. The method was, to our knowledge, not been previously coupled with the LBM method, which is usually investigated in the context of smaller Reynolds-numbers occurring in engineering problems. The rest of the chapter is structured as follows: in **3.2**, we describe the basic concepts of RANS. In **3.3**, we apply this method to the LB equations, which will lead to a formulation which can be coupled to the current turbulence models. In **3.4**, we describe the Chapman-Enskog expansion, which is commonly used in determining the macroscopic equations reproduced by LB methods [Chen and Doolen, 1998]. Ideas from this procedure will then be used in **3.5** to prove that the Reynolds-averaged LB actually recovers the corresponding Reynolds-averaged Navier-Stokes equations.

3.2 Reynolds-averaging procedure[19]

The Reynolds-averaging procedure was developed at the beginning of the $\mathbf{20}^{\boldsymbol{th}}$-century. The main idea is to consider statistical properties of the flow at each point in the physical space, including correlations and autocorrelations of various physical properties. In fact, we only discuss a certain class of these models, usually denoted as *one-point closures*, where correlations between the flow characteristics between different points are neglected.

[19]This subsection is inspired mostly by [Burchard, 2002].

Two-point closures have also been proposed [Orszag, 1977], but they are not so widespread due to the increased computational demands.

As previously mentioned, the RA procedure decomposes any prognostic variable $\boldsymbol{\psi(\vec{x}, t)}$ into a mean-field $\boldsymbol{\Psi \equiv \langle \psi \rangle}$ and a fluctuating field $\boldsymbol{\psi'}$, that is:

$$\boldsymbol{\psi = \Psi + \psi'} \tag{36}$$

Usually, such a decomposition is only well-defined if there exists a spectral gap between the slow- and fast-timescales, which is often not the case for turbulent flows. The problem can be solved, however, if a suitable averaging technique is chosen, such as the *ensemble averaging* approach. In that case, assuming a large number $\boldsymbol{n}$ of realizations of the same flow configuration with identical constraints, we can define the ensemble average at a certain space-time coordinates $\boldsymbol{(t_0, x_0, y_0, z_0)}$ as:

$$\boldsymbol{\Psi(t_0, x_0, y_0, z_0) \equiv \langle \psi(t_0, x_0, y_0, z_0) \rangle = \lim_{n \to \infty} \frac{1}{n} \sum_{i=1}^{n} \psi_i(t_0, x_0, y_0, z_0)} \tag{37}$$

This definition has been shown [Mohammadi and Pironneau, 1994] to have the following **4** important properties:

1. Linearity:

$$\boldsymbol{\langle \psi + \lambda \phi \rangle = \langle \psi \rangle + \lambda \langle \phi \rangle} \tag{38}$$

2. Averages and derivatives commute:

$$\boldsymbol{\langle \partial_{x_i} \psi \rangle = \partial_{x_i} \langle \psi \rangle} \tag{39}$$

3. Double averages:

$$\boldsymbol{\langle \langle \psi \rangle \rangle = \langle \psi \rangle} \tag{40}$$

4. Product averages:

$$\boldsymbol{\langle \psi \langle \phi \rangle \rangle = \langle \psi \rangle \langle \phi \rangle} \tag{41}$$

By definition, we also have the properties:

$$\boldsymbol{\langle \psi \rangle = \Psi} \tag{42}$$

and:

$$\langle \boldsymbol{\psi'} \rangle = \mathbf{0} \tag{43}$$

While this averaging procedure is strictly valid, it is not directly useful in real climatology due to the obvious lack of parallel realizations of the flow dynamics. However, if within the temporal and spatial intervals considered the turbulence of the flow can be taken as approximately stationary and homogeneous, the ensemble averages can be linked to time averages by virtue of the *ergodic hypothesis*, which predicts that over long enough timescales a stationary chaotic process will visit all of its micro-states.

3.3 Reynolds-averaged Lattice Boltzmann Model (RALB)

The Reynolds-averaging procedure is now applied to the Lattice Boltzmann model. Effectively, we separate the dynamics of the distribution functions into a long-timescale component and a short-timescale, highly-fluctuating component. The result of this analysis will be a new set of equations for the evolution of the long-timescale component, which can be immediately related to the evolution of the mean fields we are interested in. The Lattice Boltzmann Equation (LBE) reads (on a $\boldsymbol{D}$-dimensional lattice with $\boldsymbol{N}$ discretized velocity vectors):

$$\boldsymbol{f_i}(\vec{\boldsymbol{x}} + \vec{\boldsymbol{e_i}}, \boldsymbol{t+1}) = \boldsymbol{f_i}(\vec{\boldsymbol{x}}, \boldsymbol{t}) + \boldsymbol{\Omega_i}(\boldsymbol{f_{0..N-1}}(\vec{\boldsymbol{x}}, \boldsymbol{t})) \tag{44}$$

where the macroscopic variables are defined as:

$$\begin{cases} \boldsymbol{\rho} = \sum_i \boldsymbol{f_i} \\ \boldsymbol{\rho \vec{u}} = \sum_i \vec{\boldsymbol{e_i}} \boldsymbol{f_i} \end{cases} \tag{45}$$

and $\boldsymbol{\Omega_i}$ is the collision operator.

We introduce a decomposition of each $\boldsymbol{f_i}$ into a mean field $\boldsymbol{F_i}$ and a fluctuating part $\boldsymbol{h_i}$ with:

$$\boldsymbol{f_i} = \boldsymbol{F_i} + \boldsymbol{h_i} \tag{46}$$

$$\langle \boldsymbol{f_i} \rangle \equiv \boldsymbol{F_i} \tag{47}$$

$$\langle \boldsymbol{h_i} \rangle \equiv \mathbf{0} \tag{48}$$

and all the additional properties of the ensemble averages [Eq. (38)-(41)]. As justified above, in the subsequent discussion $\langle\cdot\rangle$ represent time averages.

Additionally, we define mean macroscopic variables:

$$\boldsymbol{R} \equiv \langle \boldsymbol{\rho} \rangle = \left\langle \sum_i \boldsymbol{f_i} \right\rangle = \sum_i \langle \boldsymbol{f_i} \rangle = \sum_i \boldsymbol{F_i} \tag{49}$$

$$\boldsymbol{R}\vec{\boldsymbol{U}} \equiv \langle \boldsymbol{\rho}\vec{\boldsymbol{u}} \rangle = \left\langle \sum_i \vec{\boldsymbol{e_i}} \boldsymbol{f_i} \right\rangle = \sum_i \vec{\boldsymbol{e_i}} \langle \boldsymbol{f_i} \rangle = \sum_i \vec{\boldsymbol{e_i}} \boldsymbol{F_i} \tag{50}$$

Inserting the decomposition into the LBE:

$$\boldsymbol{F_i}(\vec{\boldsymbol{x}}+\vec{\boldsymbol{e_i}}, \boldsymbol{t}+\boldsymbol{1}) + \boldsymbol{h_i}(\vec{\boldsymbol{x}}+\vec{\boldsymbol{e_i}}, \boldsymbol{t}+\boldsymbol{1}) = \boldsymbol{F_i}(\vec{\boldsymbol{x}}, \boldsymbol{t}) + \boldsymbol{h_i}(\vec{\boldsymbol{x}}, \boldsymbol{t}) + \boldsymbol{\Omega_i}(\boldsymbol{F_{0..18}}(\vec{\boldsymbol{x}}, \boldsymbol{t}) + \boldsymbol{h_{0..18}}(\vec{\boldsymbol{x}}, \boldsymbol{t})) \tag{51}$$

Plugging-in the simplified BGK collision operator:

$$\boldsymbol{\Omega_i} = -\frac{\boldsymbol{1}}{\boldsymbol{\lambda}} \left[\boldsymbol{f_i} - \boldsymbol{f_i^{eq}}(\boldsymbol{f_{0..18}}(\vec{\boldsymbol{x}}, \boldsymbol{t})) \right] \tag{52}$$

we obtain:

$$\begin{aligned} \boldsymbol{F_i}(\vec{\boldsymbol{x}} + \vec{\boldsymbol{e_i}}, \boldsymbol{t}+\boldsymbol{1}) + \boldsymbol{h_i}(\vec{\boldsymbol{x}} + \vec{\boldsymbol{e_i}}, \boldsymbol{t}+\boldsymbol{1}) = \boldsymbol{F_i}(\vec{\boldsymbol{x}}, \boldsymbol{t}) + \boldsymbol{h_i}(\vec{\boldsymbol{x}}, \boldsymbol{t}) - \\ -\frac{\boldsymbol{1}}{\boldsymbol{\lambda}} \left\{ \boldsymbol{F_i}(\vec{\boldsymbol{x}}, \boldsymbol{t}) + \boldsymbol{h_i}(\vec{\boldsymbol{x}}, \boldsymbol{t}) - \boldsymbol{f_i^{eq}}(\boldsymbol{f_{0..18}}(\vec{\boldsymbol{x}}, \boldsymbol{t})) \right\} \end{aligned} \tag{53}$$

Seeking an equation for the long-timescale evolution of the mean-fields[20], we will take the ensemble average of Eq. [53]:

$$\begin{aligned} \boldsymbol{F_i}(\vec{\boldsymbol{x}} + \vec{\boldsymbol{e_i}}, \boldsymbol{t}+\boldsymbol{1}) + \cancel{\langle \boldsymbol{h_i}(\vec{\boldsymbol{x}} + \vec{\boldsymbol{e_i}}, \boldsymbol{t}+\boldsymbol{1}) \rangle} = \\ \boldsymbol{F_i}(\vec{\boldsymbol{x}}, \boldsymbol{t}) + \cancel{\langle \boldsymbol{h_i}(\vec{\boldsymbol{x}}, \boldsymbol{t}) \rangle} - \frac{\boldsymbol{1}}{\boldsymbol{\lambda}} \left\{ \boldsymbol{F_i} + \cancel{\langle \boldsymbol{h_i}(\vec{\boldsymbol{x}}, \boldsymbol{t}) \rangle} - \langle \boldsymbol{f_i^{eq}} \rangle \right\} \\ \Rightarrow \boldsymbol{F_i}(\vec{\boldsymbol{x}} + \vec{\boldsymbol{e_i}}, \boldsymbol{t}+\boldsymbol{1}) = \boldsymbol{F_i}(\vec{\boldsymbol{x}}, \boldsymbol{t}) - \frac{\boldsymbol{1}}{\boldsymbol{\lambda}} \left\{ \boldsymbol{F_i} - \langle \boldsymbol{f_i^{eq}} \rangle \right\} \end{aligned} \tag{54}$$

Notice that the corresponding RALB-equation for the mean fields is very similar to the original LBE. The only missing link in the new formulation is $\langle \boldsymbol{f_i^{eq}} \rangle$, which we calculate in the next subsubsection.

[20] As a clarification, the mean fields can be thought of as running mens of the macroscopic variables with averaging intervals over short timescales, but measured at discrete, long-timescale intervals.

3.3.1 Calculation of $\langle f_i^{eq} \rangle$

The usual expression of the equilibrium distribution function used in most LB models is the $\mathbf{2^{nd}}$-order truncated Maxwellian distribution:

$$f_i^{eq} = \rho t_i \left\{ 1 + 3(\vec{e_i} \cdot \vec{u}) + \frac{9}{2}(\vec{e_i} \cdot \vec{u})^2 - \frac{3}{2}\vec{u}^2 \right\} \tag{55}$$

where t_i are some properly-chosen weights, which are in fact the stationary fluid equilibrium distribution functions[21].

Taking the ensemble average of Eq. [55]:

$$\langle f_i^{eq} \rangle = \underbrace{\langle \rho t_i \rangle}_{E_1} + \underbrace{3 \langle \rho t_i (\vec{e_i} \cdot \vec{u}) \rangle}_{E_2} + \underbrace{\frac{9}{2} \langle \rho t_i (\vec{e_i} \cdot \vec{u})^2 \rangle}_{E_3} - \underbrace{\frac{3}{2} \langle \rho t_i \vec{u}^2 \rangle}_{E_4} \tag{56}$$

Let us now evaluate the sub-expressions:

$$\mathbf{E_1} \equiv \langle \rho t_i \rangle = t_i \langle \rho \rangle = R t_i \tag{57}$$

$$\begin{aligned}
\mathbf{E_2} &\equiv 3 \langle \rho t_i (\vec{e_i} \cdot \vec{u}) \rangle = 3 t_i \langle \rho (e_{ix} u + e_{iy} v + e_{iz} w) \rangle \\
&= 3 t_i (e_{ix} \langle \rho u \rangle + e_{iy} \langle \rho v \rangle + e_{iz} \langle \rho w \rangle) \\
&= 3 t_i \left[e_{ix} \langle (R + \rho')(U + u') \rangle + e_{iy} \langle (R + \rho')(V + v') \rangle + e_{iz} \langle (R + \rho')(W + w') \rangle \right] \\
&= 3 t_i [e_{ix} (\langle RU \rangle + \langle R u' \rangle + \langle \rho' U \rangle + \langle \rho' u' \rangle) + \\
&\quad e_{iy} (\langle RV \rangle + \langle R v' \rangle + \langle \rho' V \rangle + \langle \rho' v' \rangle) + \\
&\quad e_{iz} (\langle RW \rangle + \langle R w' \rangle + \langle \rho' W \rangle + \langle \rho' w' \rangle)] \\
&= 3 t_i [e_{ix} (RU + R\cancel{\langle u' \rangle} + U\cancel{\langle \rho' \rangle} + \langle \rho' u' \rangle + \\
&\quad e_{iy} (RV + R\cancel{\langle v' \rangle} + V\cancel{\langle \rho' \rangle} + \langle \rho' v' \rangle + \\
&\quad e_{iz} (RW + R\cancel{\langle w' \rangle} + W\cancel{\langle \rho' \rangle} + \langle \rho' w' \rangle] \\
&= 3 t_i R (\vec{e_i} \cdot \vec{U}) + 3 t_i (e_{ix} \langle \rho' u' \rangle + e_{iy} \langle \rho' v' \rangle + e_{iz} \langle \rho' w' \rangle)
\end{aligned}$$

Since the method is only applicable in the incompressible limit, we neglect the $\langle \rho' u'_\alpha \rangle$-terms, finally obtaining:

$$E_2 \approx 3 t_i R (\vec{e_i} \cdot \vec{U}) \tag{58}$$

[21] From this point of view, the remaining terms in Eq. [55] can be viewed as perturbations around the stationary limit. The low order of the truncation also justifies why the simple LB method is only applicable to low Mach number-flows.

$$
\begin{aligned}
\mathbf{E_3} &\equiv \frac{9}{2}\left\langle \rho t_i (\vec{e_i}\cdot\vec{u})^2 \right\rangle = \frac{9t_i}{2}\left\langle \rho (\vec{e_i}\cdot\vec{u})^2 \right\rangle \approx \frac{9t_i}{2} R \left\langle (e_{ix}u + e_{iy}v + e_{iz}w)^2 \right\rangle \\
&= \frac{9}{2} R t_i \left[\left\langle e_{ix}^2 u^2 + e_{iy}^2 v^2 + e_{iz}^2 w^2 + 2e_{ix}e_{iy}uv + 2e_{ix}e_{iz}uw + 2e_{iy}e_{iz}vw \right\rangle \right] \\
&= \frac{9}{2} R t_i \left[e_{ix}^2 \left\langle u^2 \right\rangle + e_{iy}^2 \left\langle v^2 \right\rangle + e_{iz}^2 \left\langle w^2 \right\rangle + 2e_{ix}e_{iy} \left\langle uv \right\rangle + 2e_{ix}e_{iz} \left\langle uw \right\rangle + 2e_{iy}e_{iz} \left\langle vw \right\rangle \right]
\end{aligned}
$$

Plugging-in the decompositions for the velocity components, we further obtain:

$$
\begin{aligned}
\boldsymbol{E_3} &= \frac{9}{2} R t_i \Big[e_{ix}^2 \left\langle U^2 + 2Uu' + u'^2 \right\rangle + e_{iy}^2 \left\langle V^2 + 2Vv' + v'^2 \right\rangle + \\
&\quad e_{iz}^2 \left\langle W^2 + 2Ww' + w'^2 \right\rangle + \\
&\quad 2e_{ix}e_{iy} \left\langle UV + Uv' + Vu' + u'v' \right\rangle + \\
&\quad 2e_{ix}e_{iz} \left\langle UW + Uw' + Wu' + u'w' \right\rangle + \\
&\quad 2e_{iy}e_{iz} \left\langle VW + Vw' + Wv' + v'w' \right\rangle \Big] \\
&= \frac{9}{2} R t_i \Bigg[e_{ix}^2 \left(\underbrace{\left\langle U^2 \right\rangle}_{U^2} + 2U\cancel{\left\langle u' \right\rangle} + \left\langle u'^2 \right\rangle \right) + e_{iy}^2 \left(\underbrace{\left\langle V^2 \right\rangle}_{V^2} + 2V\cancel{\left\langle v' \right\rangle} + \left\langle v'^2 \right\rangle \right) + \\
&\quad e_{iz}^2 \left(\underbrace{\left\langle W^2 \right\rangle}_{W^2} + 2W\cancel{\left\langle w' \right\rangle} + \left\langle w'^2 \right\rangle \right) + \\
&\quad 2e_{ix}e_{iy} \left(\underbrace{\left\langle UV \right\rangle}_{UV} + U\cancel{\left\langle v' \right\rangle} + V\cancel{\left\langle u' \right\rangle} + \left\langle u'v' \right\rangle \right) + \\
&\quad 2e_{ix}e_{iz} \left(\underbrace{\left\langle UW \right\rangle}_{UW} + U\cancel{\left\langle w' \right\rangle} + W\cancel{\left\langle u' \right\rangle} + \left\langle u'w' \right\rangle \right) + \\
&\quad 2e_{iy}e_{iz} \left(\underbrace{\left\langle VW \right\rangle}_{VW} + V\cancel{\left\langle w' \right\rangle} + W\cancel{\left\langle v' \right\rangle} + \left\langle v'w' \right\rangle \right) \Bigg]
\end{aligned}
$$

thus:

$$
\begin{aligned}
\boldsymbol{E_3} &= \frac{9}{2} R t_i \Bigg\{ \left(\vec{e_i}\cdot\vec{U} \right)^2 + \Big[e_{ix}^2 \left\langle u'^2 \right\rangle + e_{iy}^2 \left\langle v'^2 \right\rangle + e_{iz}^2 \left\langle w'^2 \right\rangle + \\
&\quad 2e_{ix}e_{iy} \left\langle u'v' \right\rangle + 2e_{iy}e_{iz} \left\langle u'w' \right\rangle + 2e_{iy}e_{iz} \left\langle v'w' \right\rangle \Big] \Bigg\}
\end{aligned} \tag{59}
$$

$$
\begin{aligned}
\mathbf{E_4} &\equiv -\frac{3}{2}\left\langle \rho t_i \vec{u}^2 \right\rangle \approx -\frac{3}{2} t_i R \left\langle \vec{u}^2 \right\rangle \\
&= -\frac{3}{2} t_i R \left\{ \left\langle (U+u')^2 \right\rangle + \left\langle (V+v')^2 \right\rangle + \left\langle (W+w')^2 \right\rangle \right\} \\
&= -\frac{3}{2} t_i R \Bigg\{ \underbrace{\left\langle U^2 \right\rangle}_{U^2} + 2U \cancel{\langle u' \rangle} + \left\langle u'^2 \right\rangle + \underbrace{\left\langle V^2 \right\rangle}_{V^2} + 2V \cancel{\langle v' \rangle} + \left\langle v'^2 \right\rangle + \\
&\quad \underbrace{\left\langle W^2 \right\rangle}_{W^2} + 2W \cancel{\langle w' \rangle} + \left\langle w'^2 \right\rangle \Bigg\}
\end{aligned}
$$

$$
\boldsymbol{E_4 = -\frac{3}{2} t_i R \left\{ \vec{U}^2 + \left\langle u'^2 \right\rangle + \left\langle v'^2 \right\rangle + \left\langle w'^2 \right\rangle \right\}} \tag{60}
$$

Summing Eq. [57],[58],[59] and [60], we obtain the final expression for the Reynolds-averaged equilibrium distribution function:

$$
\boxed{\boldsymbol{\left\langle f_i^{eq} \right\rangle = R t_i \left\{ 1 + 3(\vec{e_i} \cdot \vec{U}) + \frac{9}{2}(\vec{e_i} \cdot \vec{U})^2 - \frac{3}{2}\vec{U}^2 \right\} + \Gamma_i}} \tag{61}
$$

where:

$$
\begin{aligned}
\boldsymbol{\Gamma_i} &\equiv \frac{3}{2} R t_i \Big[(3e_{ix}^2 - 1)\left\langle u'^2 \right\rangle + (3e_{iy}^2 - 1)\left\langle v'^2 \right\rangle + (3e_{iz}^2 - 1)\left\langle w'^2 \right\rangle + \\
&\quad 6 e_{ix} e_{iy} \left\langle u'v' \right\rangle + 6 e_{ix} e_{iz} \left\langle u'w' \right\rangle + 6 e_{iy} e_{iz} \left\langle v'w' \right\rangle \Big]
\end{aligned} \tag{62}
$$

is an additional term containing the $\mathbf{2^{nd}}$-order correlations due to turbulence. We notice that, in analogy to the Reynolds-averaging of the Navier-Stokes equation, the resulting equation for the time evolution of the mean field is very similar to the original equations, except that a few additional terms which are products of velocity cross-correlations are emerging. In order to obtain a solvable system of equations, additional closure relations are needed for the $\left\langle u'_\alpha u'_\beta \right\rangle$-terms. Although these closures remain an active field of research, several formulations are in widespread use in the field of numerical oceanography and yield reasonable results (to be discussed in **??**).

Another interesting observation is that the effect of turbulence on the mean fields behaves like a special type of forcing. Since the second-moments are proportional to

the stresses within the fluid, which is dependent on the local velocity gradients at each lattice point, these force-like terms are also time-dependent. This force-like behavior of the Reynolds stresses also suggests a reasonable interpretation of our result: as will be discussed later, most of the turbulence closures consider the components of the Reynolds stress tensor to be proportional to velocity gradients; it is only natural to consider the effects of the unresolved dissipative scales as quasi-forces which tend to smoothen these gradients.

3.4 Chapman-Enskog procedure

In the next subsection, we will derive the macroscopic equations corresponding to our Reynolds-averaged Lattice Boltzmann equation. On that course, it is useful to outline first the basic steps of the Chapman-Enskog (CE) expansion, which provides the fundamental link between kinetic theory and hydrodynamics. We will illustrate the procedure for the classical, single relaxation time LB scheme with BGK collision operator.

The main idea of the CE expansion is to separate the fluid motion into several timescales, to mimick the real fluids, where some processes (for example convection) are much faster than others (such as diffusion). Thus, a formal series is introduced for time:

$$\frac{\partial}{\partial t} = \epsilon \frac{\partial}{\partial t_1} + \epsilon^2 \frac{\partial}{\partial t_2} + h.o.t. \tag{63}$$

Note: we will neglect in our subsequent analysis higher order terms (h.o.t.), with $\epsilon^n,\ n > 2$.

For the spatial gradients, a single scale is used because all processes occur on roughly the same spatial extent:

$$\nabla = \epsilon \nabla_1 + h.o.t. \tag{64}$$

Also, the probability distribution function is written as a formal series:

$$f_i = f_i^{(0)} + \epsilon f_i^{(1)} + \epsilon^2 f_i^{(2)} + h.o.t. \tag{65}$$

The equations should be thought of as *formal expansions*, in the sense that the power of ϵ are just labels representing the magnitude of the terms they are multiplied with

(higher power of ϵ denotes smaller terms). This allows us to discard the formal expansion parameter at a future step by simply setting it to $\mathbf{1}$ in one of the equations.

Plugging Eq. [63]-[65] into the LB equation [44] with the BGK collision term (Eq. [52]), we obtain:

$$f_i(\vec{x}+\vec{e_i},t+1) = f_i(\vec{x},t) - \frac{1}{\tau}\left(\epsilon f_i^{(1)} + \epsilon^2 f_i^{(2)} + h.o.t.\right) \tag{66}$$

We also apply a Taylor expansion in space and time to the LHS:

$$f_i(\vec{x}+\vec{e_i},t+1) = f_i(\vec{x},t)+\frac{\partial}{\partial t}f_i(\vec{x},t)+\vec{e_i}\cdot\nabla f_i(\vec{x},t)+\frac{1}{2}\left(\frac{\partial}{\partial t}+\vec{e_i}\cdot\nabla\right)^2 f_i(\vec{x},t)+h.o.t. \tag{67}$$

Plugging Eq. [67] into Eq. [66], we obtain:

$$\underbrace{\left[\frac{\partial}{\partial t}+\vec{e_i}\cdot\nabla+\frac{1}{2}(\frac{\partial}{\partial t}+\vec{e_i}\cdot\nabla)^2\right]}_{\equiv\mathcal{L}} f_i = -\frac{1}{\tau}\left(\epsilon f_i^{(1)} + \epsilon^2 f_i^{(2)} + h.o.t.\right) \tag{68}$$

The operator $\mathcal{L}$ can be expanded as:

$$\begin{aligned}\mathcal{L} &= \frac{\partial}{\partial t}+\vec{e_i}\cdot\nabla+\frac{1}{2}\left[\frac{\partial^2}{\partial t^2}+2\frac{\partial}{\partial t}(\vec{e_i}\cdot\nabla)+(\vec{e_i}\cdot\nabla)(\vec{e_i}\cdot\nabla)\right]\\ \Rightarrow \mathcal{L} &= \frac{\partial}{\partial t}+\vec{e_i}\cdot\nabla+\frac{1}{2}\vec{e_i}\vec{e_i}:\nabla\nabla+\vec{e_i}\cdot\nabla+\frac{1}{2}\frac{\partial^2}{\partial t^2}\end{aligned} \tag{69}$$

where $\vec{A}\vec{B}\Big|_{\alpha\beta} \equiv A_\alpha B_\beta$ represents the dyadic product and $\tilde{T}:\tilde{S} \equiv \sum_{\alpha,\beta} A_{\alpha\beta}B_{\alpha\beta}$ represents the tensor contraction operator.

Plugging the expanded operator and also the series for f_i:

$$\underbrace{\left(\epsilon\frac{\partial}{\partial t_1}+\epsilon^2\frac{\partial}{\partial t_2}\right)\left(f_i^{(0)}+\epsilon f_i^{(1)}+\epsilon^2 f_i^{(2)}\right)}_{A}+\underbrace{\epsilon\vec{e_i}\cdot\nabla_1\left(f_i^{(0)}+\epsilon f_i^{(1)}+\epsilon^2 f_i^{(2)}\right)}_{B}+$$

$$\underbrace{\frac{1}{2}\epsilon^2\vec{e_i}\vec{e_i}:\nabla\nabla\left(f_i^{(0)}+\epsilon f_i^{(1)}+\epsilon^2 f_i^{(2)}\right)}_{C}+\underbrace{\epsilon\vec{e_i}\nabla\left(\epsilon\frac{\partial}{\partial t_1}+\epsilon^2\frac{\partial}{\partial t_2}\right)\left(f_i^{(0)}+\epsilon f_i^{(1)}+\epsilon^2 f_i^{(2)}\right)}_{D}+$$

$$\underbrace{\frac{1}{2}\left(\epsilon\frac{\partial}{\partial t_1}+\epsilon^2\frac{\partial}{\partial t_2}\right)^2\left(f_i^{(0)}+\epsilon f_i^{(1)}+\epsilon^2 f_i^{(2)}\right)}_{E} = -\frac{1}{\tau}\left(\epsilon f_i^{(1)}+\epsilon^2 f_i^{(2)}\right)$$

Expanding the sub-expressions and neglecting the terms beyond $\mathbf{2}^{nd}$-order:

$$\begin{aligned}
\boldsymbol{A} &= \epsilon\frac{\partial f_i^{(0)}}{\partial t_1} + \epsilon^2\frac{\partial f_i^{(1)}}{\partial t_1} + \epsilon^2\frac{\partial f_i^{(0)}}{\partial t_2} \\
\boldsymbol{B} &= \epsilon\vec{e_i}\cdot\nabla_1 f_i^{(0)} + \epsilon^2\vec{e_i}\cdot\nabla_1 f_i^{(1)} \\
\boldsymbol{C} &= \frac{1}{2}\epsilon^2\vec{e_i}\vec{e_i} : \nabla\nabla f_i^{(0)} \\
\boldsymbol{D} &= \epsilon^2\vec{e_i}\cdot\nabla_1\frac{\partial f_i^{(0)}}{\partial t_1} \\
\boldsymbol{E} &= \frac{1}{2}\left(\epsilon^2\frac{\partial^2}{\partial t_1^2} + \underbrace{\epsilon^4\frac{\partial^2}{\partial t_2^2}}_{\approx 0} + \underbrace{2\epsilon^3\frac{\partial^2}{\partial t_1 \partial t_2}}_{\approx 0}\right)\left(f_i^{(0)} + \epsilon f_i^{(1)} + \epsilon^2 f_i^{(2)}\right) \approx \frac{1}{2}\epsilon^2\frac{\partial^2 f_i^{(0)}}{\partial t_1^2}
\end{aligned}$$

Re-inserting the sub-expressions, we obtain:

$$\begin{aligned}
\underline{\epsilon\frac{\partial f_i^{(0)}}{\partial t_1}} + \epsilon^2\frac{\partial f_i^{(1)}}{\partial t_1} + \epsilon^2\frac{\partial f_i^{(0)}}{\partial t_2} + \underline{\epsilon\vec{e_i}\cdot\nabla_1 f_i^{(0)}} + \epsilon^2\vec{e_i}\cdot\nabla_1 f_i^{(1)} + \\
\frac{1}{2}\epsilon^2\vec{e_i}\vec{e_i} : \nabla\nabla f_i^{(0)} + \epsilon^2\vec{e_i}\cdot\nabla_1\frac{\partial f_i^{(0)}}{\partial t_1} + \frac{1}{2}\epsilon^2\frac{\partial^2 f_i^{(0)}}{\partial t_1^2} &= \underline{-\frac{1}{\tau}\epsilon f_i^{(1)}} - \frac{1}{\tau}\epsilon^2 f_i^{(2)}
\end{aligned}$$

To a consistent order, we can transform this relation into two equalities, one for each order of $\boldsymbol{\epsilon}$. The $\mathbf{1}^{st}$-order terms (underlined in the equation above) yield:

$$\boxed{\frac{\partial f_i^{(0)}}{\partial t_1} + \vec{e_i}\cdot\nabla_1 f_i^{(0)} = -\frac{1}{\tau}f_i^{(1)}} \tag{70}$$

and the corresponding $\mathbf{2}^{nd}$-order equation:

$$\begin{aligned}
\frac{\partial f_i^{(1)}}{\partial t_1} + \frac{\partial f_i^{(0)}}{\partial t_2} + \vec{e_i}\cdot\nabla_1 f_i^{(1)} + \frac{1}{2}\vec{e_i}\vec{e_i} : \nabla\nabla f_i^{(0)} + \\
\vec{e_i}\cdot\nabla_1\frac{\partial f_i^{(0)}}{\partial t_1} + \frac{1}{2}\frac{\partial^2 f_i^{(0)}}{\partial t_1^2} &= -\frac{1}{\tau}f_i^{(2)}
\end{aligned} \tag{71}$$

Also, from Eq. [70]:

$$\begin{aligned}
\left(\frac{1}{2}\vec{e_i}\vec{e_i} : \nabla\nabla f_i^{(0)} + \vec{e_i}\cdot\nabla_1\frac{\partial f_i^{(0)}}{\partial t_1} + \frac{1}{2}\frac{\partial^2 f_i^{(0)}}{\partial t_1^2}\right) &\equiv \\
\frac{1}{2}\left(\frac{\partial}{\partial t_1} + \vec{e_i}\cdot\nabla_1\right)^2 f_i^{(0)} &= \\
-\frac{1}{2\tau}\left(\frac{\partial}{\partial t_1} + \vec{e_i}\cdot\nabla_1\right) f_i^{(1)}
\end{aligned} \tag{72}$$

Inserting Eq. [72] into Eq. [71], we obtain:

$$\boxed{\frac{\partial f_i^{(0)}}{\partial t_2} + \left(1 - \frac{1}{2\tau}\right)\left(\frac{\partial}{\partial t_1} + \vec{e_1} \cdot \nabla_1\right) f_i^{(1)} = -\frac{1}{\tau} f_i^{(2)}} \tag{73}$$

Finally, the equations governing the evolution of macroscopic fields can be derived by taking the $\mathbf{1}^{st}$ and second moments of the distribution functions w.r.t. the discretized velocity space.

3.4.1 Derivation of the continuity equation

For the continuity equation, we first take the first moment of Eq. [70]:

$$\begin{aligned}
\sum_i (70) \quad &\Rightarrow \quad \sum_i \frac{\partial}{\partial t_1} f_i^{(0)} + \sum_i \vec{e_i} \cdot \nabla_1 f_i^{(0)} = -\frac{1}{\tau} \underbrace{\sum_i f_i^{(1)}}_{=0} \\
&\Rightarrow \quad \frac{\partial}{\partial t_1} \underbrace{\sum_i f_i^{(0)}}_{=\rho} + \underline{\sum_i \vec{e_i} \cdot \nabla_1 f_i^{(0)}} = 0
\end{aligned}$$

Since the discretized velocity vectors $\vec{e_i}$ are not depending on the spatial coordinates, the second term in the LHS is equivalent to:

$$\begin{aligned}
\nabla_1 \cdot \underbrace{\left(\sum_i \vec{e_i} f_i^{(0)}\right)}_{=\rho\vec{u}} &= \sum_i \nabla_1 \cdot \left(\vec{e_i} f_i^{(0)}\right) \\
&= \sum_i \left(\frac{\partial}{\partial x_1}, \frac{\partial}{\partial y_1}, \frac{\partial}{\partial z_1}\right) \begin{pmatrix} e_{ix} f_i^{(0)} \\ e_{iy} f_i^{(0)} \\ e_{iz} f_i^{(0)} \end{pmatrix} \\
&= \sum_i \left(e_{ix} \frac{\partial f_i^{(0)}}{\partial x_1} + e_{iy} \frac{\partial f_i^{(0)}}{\partial y_1} + e_{iz} \frac{\partial f_i^{(0)}}{\partial z_1}\right) \\
&= \underline{\sum_i \vec{e_i} \cdot \nabla_1 f_i^{(0)}}
\end{aligned}$$

With the last result, we obtain:

$$\boxed{\frac{\partial \rho}{\partial t_1} + \nabla_1 \cdot (\rho \vec{u}) = 0} \tag{74}$$

Taking the $\mathbf{1}^{st}$ moment of Eq. [73]:

$$\sum_i (73) \;\Rightarrow\; \sum_i \frac{\partial f_i^{(0)}}{\partial t_2} + \left(1 - \frac{1}{2\tau}\right) \sum_i \left(\frac{\partial}{\partial t_1} + \vec{e}_i \cdot \nabla_1\right) f_i^{(1)} = -\frac{1}{\tau} \underbrace{\sum_i f_i^{(2)}}_{=0}$$

$$\Rightarrow\; \frac{\partial \rho}{\partial t_2} + \left(1 - \frac{1}{2\tau}\right) \frac{\partial}{\partial t_1} \underbrace{\sum_i f_i^{(1)}}_{=0} + \left(1 - \frac{1}{2\tau}\right) \underbrace{\underbrace{\sum_i \vec{e}_i \cdot \nabla_1 f_i^{(1)}}_{= \nabla_1 \cdot \sum_i \vec{e}_i f_i^{(1)}}}_{=0} = 0$$

which leads to:

$$\boxed{\frac{\partial \rho}{\partial t_2} = 0} \tag{75}$$

Adding Eq. [74] and [75] and setting the *formal* expansion parameter $\epsilon \equiv 1$, we obtain:

$$\boxed{\frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \vec{u}) = 0} \tag{76}$$

which is nothing else than the *continuity equation*.

3.4.2 Derivation of the momentum equation

For the momentum equation, we now take the *second* moments of Eq. [70]:

$$\sum_i \vec{e}_i (70) \;\Rightarrow\; \sum_i \vec{e}_i \frac{\partial f_i^{(0)}}{\partial t_1} + \sum_i \vec{e}_i (\vec{e}_i \cdot \nabla_1 f_i^{(0)}) = -\frac{1}{\tau} \underbrace{\sum_i \vec{e}_i f_i^{(1)}}_{=0}$$

$$\Rightarrow\; \frac{\partial}{\partial t_1} \underbrace{\sum_i \vec{e}_i f_i^{(0)}}_{=\rho \vec{u}} + \nabla_1 \cdot \tilde{\Pi}^{(0)} = 0$$

$$\Rightarrow\; \frac{\partial \rho \vec{u}}{\partial t_1} + \nabla_1 \cdot \tilde{\Pi}^{(0)} = 0, \tag{77}$$

with

$$\Pi_{\alpha\beta}^{(0)} = \sum_i e_{i\alpha} e_{i\beta} f_i^{(0)} \tag{78}$$

Also, taking the second moment of Eq. [73]:

$$\sum_i \vec{e_i}(73) \;\Rightarrow\; \underbrace{\sum_i \vec{e_i}\frac{\partial f_i^{(0)}}{\partial t_2}}_{=\frac{\partial \rho \vec{u}}{\partial t_2}} + \left(1-\frac{1}{2\tau}\right)\underbrace{\sum_i \vec{e_i}\frac{\partial f_i^{(1)}}{\partial t_1}}_{=0} +$$
$$\left(1-\frac{1}{2\tau}\right)\underbrace{\sum_i \vec{e_i}\left(\vec{e_i}\cdot\nabla_1 f_i^{(1)}\right)}_{=\nabla_1\cdot\tilde{\Pi}^{(1)}} = -\frac{1}{\tau}\underbrace{\sum_i \vec{e_i} f_i^{(2)}}_{=0}$$
$$\Rightarrow\; \frac{\partial \rho \vec{u}}{\partial t_2} + \left(1-\frac{1}{2\tau}\right)\nabla_1\cdot\tilde{\Pi}^{(1)} = 0\ , \tag{79}$$

where

$$\Pi^{(1)}_{\alpha\beta} = \sum_i e_{i\alpha} e_{i\beta} f_i^{(1)}\ . \tag{80}$$

As for the continuity equation, adding Eq. [77] and [79] and setting the expansion parameter to $\epsilon = 1$, we obtain:

$$\boxed{\frac{\partial \rho \vec{u}}{\partial t} + \nabla\cdot\left[\tilde{\Pi}^{(0)} + \left(1-\frac{1}{2\tau}\right)\tilde{\Pi}^{(1)}\right] = 0} \tag{81}$$

It is interesting to observe that up to this point it was not necessary to know the specific functional form of the equilibrium distribution. Based only on the fact that its first two moments equal the ones of the instantaneous distribution functions set (which is the same as saying that LB collisions conserve mass and momentum), we obtained the correct form of the continuity equation. The situation is a little more complicated for the momentum equation. As it can be observed, Eq. [81] is still not identical to the Navier-Stokes formulation. In order to establish this missing agreement, we will need indeed to use the specific expression of f_i^{eq} to calculate the equilibrium ($\tilde{\Pi}^{(0)}$) and 1^{st}-order perturbation ($\tilde{\Pi}^{(1)}$) of the momentum flux tensor. These calculations are rather lengthy and are thus relegated to the **Appendix**. However, it will be useful in the later parts of our discussion to note that a key part in the calculations is played by several important symmetries of the usual[22] LB lattice vectors $\vec{e_i}$ and weights t_i in the

[22] The $D3Q19$-lattice, which is the one we used throughout the present study, also possesses these properties.

equilibrium distribution functions [Latt, 2007]:

$$\sum_i \boldsymbol{t_i} = \mathbf{1} \tag{82}$$

$$\sum_i \boldsymbol{t_i e_{i\alpha}} = \mathbf{0} \tag{83}$$

$$\sum_i \boldsymbol{t_i e_{i\alpha} e_{i\beta}} = \boldsymbol{c_s^2 \delta_{\alpha\beta}} \tag{84}$$

$$\sum_i \boldsymbol{t_i e_{i\alpha} e_{i\beta} e_{i\gamma}} = \mathbf{0} \tag{85}$$

$$\sum_i \boldsymbol{t_i e_{i\alpha} e_{i\beta} e_{i\gamma} e_{i\delta}} = \boldsymbol{c_s^4(\delta_{\alpha\beta}\delta_{\gamma\delta} + \delta_{\alpha\gamma}\delta_{\beta\delta} + \delta_{\alpha\delta}\delta_{\beta\gamma})} \tag{86}$$

$$\sum_i \boldsymbol{t_i e_{i\alpha} e_{i\beta} e_{i\gamma} e_{i\delta} e_{i\epsilon}} = \mathbf{0} \tag{87}$$

Using these properties and the expression for $\boldsymbol{f_i^{(eq)}}$, it is a matter of algebra to evaluate the stress tensor (see **Appendix** for a complete derivation) as:

$$\boldsymbol{\Pi_{\alpha\beta}^{(0)}} = \frac{\boldsymbol{\rho}}{\mathbf{3}}\boldsymbol{\delta_{\alpha\beta}} + \boldsymbol{\rho u_\alpha u_\beta} \tag{88}$$

$$\boldsymbol{\Pi_{\alpha\beta}^{(1)}} = -\frac{\boldsymbol{\rho\tau}}{\mathbf{3}}\left(\frac{\boldsymbol{\partial u_\beta}}{\boldsymbol{\partial x_{\alpha 1}}} + \frac{\boldsymbol{\partial u_\alpha}}{\boldsymbol{\partial x_{\beta 1}}}\right) \tag{89}$$

Inserting Eq. (89) and (89) into Eq. (81), we obtain[23] for the $\boldsymbol{\alpha}$-component of fluid velocity:

$$\frac{\boldsymbol{\partial(\rho u_\alpha)}}{\boldsymbol{\partial t}} + \sum_\beta \frac{\boldsymbol{\partial}}{\boldsymbol{\partial x_\beta}}\left(\frac{\boldsymbol{\rho}}{\mathbf{3}}\boldsymbol{\delta_{\alpha\beta}} + \boldsymbol{\rho u_\alpha u_\beta}\right) - \frac{\boldsymbol{\tau}}{\mathbf{3}}\left(\mathbf{1} - \frac{\mathbf{1}}{\boldsymbol{2\tau}}\right)\sum_\gamma \frac{\boldsymbol{\partial}}{\boldsymbol{\partial x_\gamma}}\left[\boldsymbol{\rho}\left(\frac{\boldsymbol{\partial u_\gamma}}{\boldsymbol{\partial x_\alpha}} + \frac{\boldsymbol{\partial u_\alpha}}{\boldsymbol{\partial x_\gamma}}\right)\right] = \mathbf{0} \tag{90}$$

In the incompressible limit, the equation becomes:

$$\frac{\boldsymbol{\partial u_\alpha}}{\boldsymbol{\partial t}} + (\boldsymbol{\vec{u}\cdot\nabla})\boldsymbol{u_\alpha} = -\frac{\mathbf{1}}{\boldsymbol{\rho_0}}\frac{\boldsymbol{\partial(\rho_0/3)}}{\boldsymbol{\partial x_\alpha}} + \frac{\mathbf{1}}{\mathbf{3}}\left(\mathbf{1} - \frac{\mathbf{1}}{\boldsymbol{2\tau}}\right)\boldsymbol{\nabla^2 u_\alpha}\ ,$$

[23] The divergence operator reduces by one the dimensionality of the quantity it is applied to, hence in our case it will transform the tensors to vectors

which is precisely the Navier-Stokes[24] equation for incompressible flows

$$\boxed{\frac{\partial u_\alpha}{\partial t} + (\vec{u}\cdot\nabla)u_\alpha = -\frac{1}{\rho_0}\frac{\partial p}{\partial x_\alpha} + \nu\nabla^2 u_\alpha}\,, \tag{91}$$

where we identified: $\boxed{p \equiv \frac{\rho}{3}}$; $\boxed{\nu \equiv \frac{1}{3}\left(1 - \frac{1}{2\tau}\right)}$. (92)

3.5 Linking RALB and RANS

In this subsection, we will use the Chapman-Enskog (CE) analysis to derive the macroscopic equations for our RALB model. In fact, a full repetition of the CE analysis is not necessary, as it will become clear shortly.

First of all, let us remark that the properties [82] to [87] are also valid for our new model, since the structure of the lattice and the weights remain the same as in traditional LB theory.

We need to check whether the additional terms $\mathbf{\Gamma}_i$ (introduced by the Reynolds-averaging procedure) have any contribution to the mass and momentum of the fluid. For this, we will calculate the $\mathbf{1}^{st}$ and $\mathbf{2}^{nd}$ moments of $\mathbf{\Gamma}_i$:

$$\begin{aligned}
\sum_i \mathbf{\Gamma}_i &= \frac{3R}{2}\left\{\sum_i t_i(3e_{ix}^2-1)\langle u'^2\rangle + \sum_i t_i(3e_{iy}^2-1)\langle v'^2\rangle + \sum_i t_i(3e_{iz}^2-1)\langle w'^2\rangle + \right.\\
&\quad \left. 6\sum_i t_i e_{ix}e_{iy}\langle u'v'\rangle + 6\sum_i t_i e_{ix}e_{iz}\langle u'w'\rangle + 6\sum_i t_i e_{iy}e_{iz}\langle v'w'\rangle\right\}\\
&= \frac{3R}{2}\left\{3\langle u'^2\rangle\underbrace{\sum_i t_i e_{ix}e_{ix}}_{=c_s^2} - \langle u'^2\rangle\underbrace{\sum_i t_i}_{=1} + 3\langle v'^2\rangle\underbrace{\sum_i t_i e_{iy}e_{iy}}_{=c_s^2} - \langle v'^2\rangle\underbrace{\sum_i t_i}_{=1} + \right.\\
&\quad 3\langle w'^2\rangle\underbrace{\sum_i t_i e_{iz}e_{iz}}_{=c_s^2} - \langle w'^2\rangle\underbrace{\sum_i t_i}_{=1} +\\
&\quad \left. 6\langle u'v'\rangle\cancel{\sum_i t_i e_{ix}e_{iy}} + 6\langle u'w'\rangle\cancel{\sum_i t_i e_{ix}e_{iz}} + 6\langle v'w'\rangle\cancel{\sum_i t_i e_{iy}e_{iz}}\right\}\\
&= \frac{3R}{2}\left\{\langle u'^2\rangle(3c_s^2-1) + \langle v'^2\rangle(3c_s^2-1) + \langle w'^2\rangle(3c_s^2-1)\right\}
\end{aligned}$$

[24] The equation of state $p = p(\rho)$ and the expression for the kinematic viscosity ν are given above in the specific case of the ***D3Q19***-lattice. However, the expressions can be generalized to other lattices, leading to Eqs (15) and (16).

But, for the $\boldsymbol{D3Q19}$-lattice, $\boldsymbol{c_s^2 = 1/3 \Rightarrow 3c_s^2 - 1 = 0}$, hence:

$$\boxed{\sum_i \boldsymbol{\Gamma}_i = \mathbf{0}} \tag{93}$$

The $\boldsymbol{x}$-component of the second moment reads:

$$\begin{aligned}
\left(\sum_i \vec{\boldsymbol{e}}_i \boldsymbol{\Gamma}_i\right)_{\boldsymbol{x}} &\equiv \sum_i \boldsymbol{e}_{i\boldsymbol{x}} \boldsymbol{\Gamma}_i \\
&= \frac{\boldsymbol{3R}}{\mathbf{2}} \Bigg\{ \sum_i \boldsymbol{t}_i \boldsymbol{e}_{i\boldsymbol{x}} (\mathbf{3}\boldsymbol{e}_{i\boldsymbol{x}}^{\mathbf{2}} - \mathbf{1}) \left\langle \boldsymbol{u}'^{\mathbf{2}} \right\rangle + \sum_i \boldsymbol{t}_i \boldsymbol{e}_{i\boldsymbol{x}} (\mathbf{3}\boldsymbol{e}_{i\boldsymbol{y}}^{\mathbf{2}} - \mathbf{1}) \left\langle \boldsymbol{v}'^{\mathbf{2}} \right\rangle + \\
&\quad \sum_i \boldsymbol{t}_i \boldsymbol{e}_{i\boldsymbol{x}} (\mathbf{3}\boldsymbol{e}_{i\boldsymbol{z}}^{\mathbf{2}} - \mathbf{1}) \left\langle \boldsymbol{w}'^{\mathbf{2}} \right\rangle + \mathbf{6} \sum_i \boldsymbol{t}_i \boldsymbol{e}_{i\boldsymbol{x}} \boldsymbol{e}_{i\boldsymbol{x}} \boldsymbol{e}_{i\boldsymbol{y}} \left\langle \boldsymbol{u}'\boldsymbol{v}' \right\rangle + \\
&\quad \mathbf{6} \sum_i \boldsymbol{t}_i \boldsymbol{e}_{i\boldsymbol{x}} \boldsymbol{e}_{i\boldsymbol{x}} \boldsymbol{e}_{i\boldsymbol{z}} \left\langle \boldsymbol{u}'\boldsymbol{w}' \right\rangle + \mathbf{6} \sum_i \boldsymbol{t}_i \boldsymbol{e}_{i\boldsymbol{x}} \boldsymbol{e}_{i\boldsymbol{y}} \boldsymbol{e}_{i\boldsymbol{z}} \left\langle \boldsymbol{v}'\boldsymbol{w}' \right\rangle \Bigg\} \\
&= \frac{\boldsymbol{3R}}{\mathbf{2}} \Bigg\{ \mathbf{3} \left\langle \boldsymbol{u}'^{\mathbf{2}} \right\rangle \underbrace{\sum_i \boldsymbol{t}_i \boldsymbol{e}_{i\boldsymbol{x}} \boldsymbol{e}_{i\boldsymbol{x}} \boldsymbol{e}_{i\boldsymbol{x}}}_{=\mathbf{0}} - \left\langle \boldsymbol{u}'^{\mathbf{2}} \right\rangle \underbrace{\sum_i \boldsymbol{t}_i \boldsymbol{e}_{i\boldsymbol{x}}}_{=\mathbf{0}} + \\
&\quad \mathbf{3} \left\langle \boldsymbol{v}'^{\mathbf{2}} \right\rangle \underbrace{\sum_i \boldsymbol{t}_i \boldsymbol{e}_{i\boldsymbol{x}} \boldsymbol{e}_{i\boldsymbol{y}} \boldsymbol{e}_{i\boldsymbol{y}}}_{=\mathbf{0}} - \left\langle \boldsymbol{v}'^{\mathbf{2}} \right\rangle \underbrace{\sum_i \boldsymbol{t}_i \boldsymbol{e}_{i\boldsymbol{x}}}_{=\mathbf{0}} + \\
&\quad \mathbf{3} \left\langle \boldsymbol{w}'^{\mathbf{2}} \right\rangle \underbrace{\sum_i \boldsymbol{t}_i \boldsymbol{e}_{i\boldsymbol{x}} \boldsymbol{e}_{i\boldsymbol{z}} \boldsymbol{e}_{i\boldsymbol{z}}}_{=\mathbf{0}} - \left\langle \boldsymbol{w}'^{\mathbf{2}} \right\rangle \underbrace{\sum_i \boldsymbol{t}_i \boldsymbol{e}_{i\boldsymbol{x}}}_{=\mathbf{0}} + \\
&\quad \mathbf{6} \left\langle \boldsymbol{u}'\boldsymbol{v}' \right\rangle \underbrace{\sum_i \boldsymbol{t}_i \boldsymbol{e}_{i\boldsymbol{x}} \boldsymbol{e}_{i\boldsymbol{x}} \boldsymbol{e}_{i\boldsymbol{y}}}_{=\mathbf{0}} + + \mathbf{6} \left\langle \boldsymbol{u}'\boldsymbol{w}' \right\rangle \underbrace{\sum_i \boldsymbol{t}_i \boldsymbol{e}_{i\boldsymbol{x}} \boldsymbol{e}_{i\boldsymbol{x}} \boldsymbol{e}_{i\boldsymbol{z}}}_{=\mathbf{0}} + \\
&\quad \mathbf{6} \left\langle \boldsymbol{v}'\boldsymbol{w}' \right\rangle \underbrace{\sum_i \boldsymbol{t}_i \boldsymbol{e}_{i\boldsymbol{x}} \boldsymbol{e}_{i\boldsymbol{y}} \boldsymbol{e}_{i\boldsymbol{z}}}_{=\mathbf{0}} \Bigg\} = \mathbf{0}
\end{aligned}$$

Due to symmetry conditions, similar relations hold for the other components of the second moment of $\boldsymbol{\Gamma_i}$, hence:

$$\boxed{\sum_i \vec{\boldsymbol{e}}_i \boldsymbol{\Gamma}_i = \mathbf{0}} \tag{94}$$

To conclude, we have shown that the additional terms due to the turbulent stresses in the RALB-model have no contribution to the mass and momentum during the collision. It will be shown that they **do** influence the *evolution* equations, in the sense that they introduce an additional mechanism for momentum dissipation.

Since the emergence of the continuity equation from the Chapman-Enskog analysis is independent on the actual functional form of the equilibrium distribution function, it is obvious that the RALB will exhibit the same macroscopic continuity equation. The only change where deviations from the standard model occur are in the momentum flux tensor or, in other words, in its sub-components $\boldsymbol{\Pi}^{(0)}_{\alpha\beta}$ and $\boldsymbol{\Pi}^{(1)}_{\alpha\beta}$.

3.5.1 Correction for $\boldsymbol{\Pi}^{(0)}_{\alpha\beta}$

The correction for the equilibrium momentum flux tensor is:

$$\boldsymbol{\Delta\Pi}^{(0)}_{\alpha\beta} = \sum_i \boldsymbol{e}_{i\alpha}\boldsymbol{e}_{i\beta}\boldsymbol{\Gamma}_i \overset{Not}{\equiv} \frac{3\boldsymbol{R}}{2}\boldsymbol{E} \tag{95}$$

where:

$$\begin{aligned}
\boldsymbol{E} \equiv\ & \sum_i \boldsymbol{t}_i\boldsymbol{e}_{i\alpha}\boldsymbol{e}_{i\beta}(3\boldsymbol{e}_{ix}^2-1)\left\langle \boldsymbol{u}'^2\right\rangle + \sum_i \boldsymbol{t}_i\boldsymbol{e}_{i\alpha}\boldsymbol{e}_{i\beta}(3\boldsymbol{e}_{iy}^2-1)\left\langle \boldsymbol{v}'^2\right\rangle + \\
& \sum_i \boldsymbol{t}_i\boldsymbol{e}_{i\alpha}\boldsymbol{e}_{i\beta}(3\boldsymbol{e}_{iz}^2-1)\left\langle \boldsymbol{w}'^2\right\rangle + 6\sum_i \boldsymbol{t}_i\boldsymbol{e}_{i\alpha}\boldsymbol{e}_{i\beta}\boldsymbol{e}_{ix}\boldsymbol{e}_{iy}\left\langle \boldsymbol{u}'\boldsymbol{v}'\right\rangle + \\
& 6\sum_i \boldsymbol{t}_i\boldsymbol{e}_{i\alpha}\boldsymbol{e}_{i\beta}\boldsymbol{e}_{ix}\boldsymbol{e}_{iz}\left\langle \boldsymbol{u}'\boldsymbol{w}'\right\rangle + 6\sum_i \boldsymbol{t}_i\boldsymbol{e}_{i\alpha}\boldsymbol{e}_{i\beta}\boldsymbol{e}_{iy}\boldsymbol{e}_{iz}\left\langle \boldsymbol{v}'\boldsymbol{w}'\right\rangle \\
=\ & 3\left\langle \boldsymbol{u}'^2\right\rangle \underbrace{\sum_i \boldsymbol{t}_i\boldsymbol{e}_{i\alpha}\boldsymbol{e}_{i\beta}\boldsymbol{e}_{ix}\boldsymbol{e}_{ix}}_{\boldsymbol{E}_{xx}} - \left\langle \boldsymbol{u}'^2\right\rangle \underbrace{\sum_i \boldsymbol{t}_i\boldsymbol{e}_{i\alpha}\boldsymbol{e}_{i\beta}}_{=c_s^2\delta_{\alpha\beta}} + \\
& 3\left\langle \boldsymbol{v}'^2\right\rangle \underbrace{\sum_i \boldsymbol{t}_i\boldsymbol{e}_{i\alpha}\boldsymbol{e}_{i\beta}\boldsymbol{e}_{iy}\boldsymbol{e}_{iy}}_{\boldsymbol{E}_{yy}} - \left\langle \boldsymbol{v}'^2\right\rangle \underbrace{\sum_i \boldsymbol{t}_i\boldsymbol{e}_{i\alpha}\boldsymbol{e}_{i\beta}}_{=c_s^2\delta_{\alpha\beta}} + \\
& 3\left\langle \boldsymbol{w}'^2\right\rangle \underbrace{\sum_i \boldsymbol{t}_i\boldsymbol{e}_{i\alpha}\boldsymbol{e}_{i\beta}\boldsymbol{e}_{iz}\boldsymbol{e}_{iz}}_{\boldsymbol{E}_{zz}} - \left\langle \boldsymbol{w}'^2\right\rangle \underbrace{\sum_i \boldsymbol{t}_i\boldsymbol{e}_{i\alpha}\boldsymbol{e}_{i\beta}}_{=c_s^2\delta_{\alpha\beta}} + \\
& 6\left\langle \boldsymbol{u}'\boldsymbol{v}'\right\rangle \underbrace{\sum_i \boldsymbol{t}_i\boldsymbol{e}_{i\alpha}\boldsymbol{e}_{i\beta}\boldsymbol{e}_{ix}\boldsymbol{e}_{iy}}_{\boldsymbol{E}_{xy}} + 6\left\langle \boldsymbol{u}'\boldsymbol{w}'\right\rangle \underbrace{\sum_i \boldsymbol{t}_i\boldsymbol{e}_{i\alpha}\boldsymbol{e}_{i\beta}\boldsymbol{e}_{ix}\boldsymbol{e}_{iz}}_{\boldsymbol{E}_{xz}} + 6\left\langle \boldsymbol{v}'\boldsymbol{w}'\right\rangle \underbrace{\sum_i \boldsymbol{t}_i\boldsymbol{e}_{i\alpha}\boldsymbol{e}_{i\beta}\boldsymbol{e}_{iy}\boldsymbol{e}_{iz}}_{\boldsymbol{E}_{yz}}
\end{aligned}$$

We can then proceed in evaluating the sub-expressions (making heavy use of Eq. [86]):

$$\boldsymbol{E}_{xx} \equiv \sum_i \boldsymbol{t}_i\boldsymbol{e}_{i\alpha}\boldsymbol{e}_{i\beta}\boldsymbol{e}_{ix}\boldsymbol{e}_{ix} = c_s^4\left(\boldsymbol{\delta}_{\alpha\beta}\underbrace{\boldsymbol{\delta}_{xx}}_{=1} + 2\boldsymbol{\delta}_{\alpha x}\boldsymbol{\delta}_{\beta x}\right)$$

hence:

$$\boldsymbol{E}_{xx} = c_s^4(\boldsymbol{\delta}_{\alpha\beta} + 2\boldsymbol{\delta}_{\alpha x}\boldsymbol{\delta}_{\beta x})$$

and similarly:

$$\boldsymbol{E_{yy} = c_s^4(\delta_{\alpha\beta} + 2\delta_{\alpha y}\delta_{\beta y})}$$

$$\boldsymbol{E_{zz} = c_s^4(\delta_{\alpha\beta} + 2\delta_{\alpha z}\delta_{\beta z})}$$

For the off-diagonal terms:

$$\boldsymbol{E_{xy} \equiv \sum_i t_i e_{i\alpha} e_{i\beta} e_{ix} e_{iy} = c_s^4 \left(\delta_{\alpha\beta} \underbrace{\delta_{xy}}_{=0} + \delta_{\alpha x}\delta_{\beta y} + \delta_{\alpha y}\delta_{\beta x} \right)}$$

thus:

$$\boldsymbol{E_{xy} = c_s^4(\delta_{\alpha x}\delta_{\beta y} + \delta_{\alpha y}\delta_{\beta x})}$$

and similarily:

$$\boldsymbol{E_{xz} = c_s^4(\delta_{\alpha x}\delta_{\beta z} + \delta_{\alpha z}\delta_{\beta x})}$$

$$\boldsymbol{E_{yz} = c_s^4(\delta_{\alpha y}\delta_{\beta z} + \delta_{\alpha z}\delta_{\beta y})}$$

Plugging-in all of the sub-expressions, we obtain:

$$\begin{aligned}
\boldsymbol{\Delta\Pi_{\alpha\beta}^{(0)}} \;=\; & \boldsymbol{\frac{3R}{2}} \Bigg\{ \left\langle \boldsymbol{u'^2} \right\rangle \left[\underbrace{\boldsymbol{3c_s^4}}_{\boldsymbol{1/3}} (\boldsymbol{\delta_{\alpha\beta} + 2\delta_{\alpha x}\delta_{\beta x}}) - \boldsymbol{c_s^2\delta_{\alpha\beta}} \right] + \\
& \left\langle \boldsymbol{v'^2} \right\rangle \left[\underbrace{\boldsymbol{3c_s^4}}_{\boldsymbol{1/3}} (\boldsymbol{\delta_{\alpha\beta} + 2\delta_{\alpha y}\delta_{\beta y}}) - \boldsymbol{c_s^2\delta_{\alpha\beta}} \right] + \\
& \left\langle \boldsymbol{w'^2} \right\rangle \left[\underbrace{\boldsymbol{3c_s^4}}_{\boldsymbol{1/3}} (\boldsymbol{\delta_{\alpha\beta} + 2\delta_{\alpha z}\delta_{\beta z}}) - \boldsymbol{c_s^2\delta_{\alpha\beta}} \right] + \\
& \boldsymbol{6 \left\langle u'v' \right\rangle} \underbrace{\boldsymbol{c_s^4}}_{\boldsymbol{1/9}} (\boldsymbol{\delta_{\alpha x}\delta_{\beta y} + \delta_{\alpha y}\delta_{\beta x}}) + \\
& \boldsymbol{6 \left\langle u'w' \right\rangle} \underbrace{\boldsymbol{c_s^4}}_{\boldsymbol{1/9}} (\boldsymbol{\delta_{\alpha x}\delta_{\beta z} + \delta_{\alpha z}\delta_{\beta x}}) + \\
& \boldsymbol{6 \left\langle v'w' \right\rangle} \underbrace{\boldsymbol{c_s^4}}_{\boldsymbol{1/9}} (\boldsymbol{\delta_{\alpha y}\delta_{\beta z} + \delta_{\alpha z}\delta_{\beta y}}) \Bigg\}
\end{aligned}$$

$$
\begin{aligned}
\Rightarrow \Delta\Pi^{(0)}_{\alpha\beta} \;=\; & \frac{\cancel{3}R}{\cancel{2}}\cdot\frac{\cancel{2}}{\cancel{3}}\left\{\left\langle u'^2\right\rangle\delta_{\alpha x}\delta_{\beta x}+\left\langle v'^2\right\rangle\delta_{\alpha y}\delta_{\beta y}+\left\langle w'^2\right\rangle\delta_{\alpha z}\delta_{\beta z}\right\}+\\
& \frac{\cancel{3}R}{\cancel{2}}\cdot\frac{\cancel{6}}{\cancel{9}}\Big\{\left\langle u'v'\right\rangle\left(\delta_{\alpha x}\delta\beta y+\delta_{\alpha y}\delta_{\beta x}\right)+\\
& \left\langle u'w'\right\rangle\left(\delta_{\alpha x}\delta\beta z+\delta_{\alpha z}\delta_{\beta x}\right)+\\
& \left\langle v'w'\right\rangle\left(\delta_{\alpha y}\delta\beta z+\delta_{\alpha z}\delta_{\beta y}\right)\Big\}
\end{aligned}
$$

Therefore:

$$
\begin{aligned}
\Delta\Pi^{(0)}_{\alpha\beta} \;=\; & R\Big\{\left\langle u'^2\right\rangle\delta_{\alpha x}\delta_{\beta x}+\left\langle v'^2\right\rangle\delta_{\alpha y}\delta_{\beta y}+\left\langle w'^2\right\rangle\delta_{\alpha z}\delta_{\beta z}+\\
& \left\langle u'v'\right\rangle\left(\delta_{\alpha x}\delta_{\beta y}+\delta_{\alpha y}\delta_{\beta x}\right)+\left\langle u'w'\right\rangle\left(\delta_{\alpha x}\delta_{\beta z}+\delta_{\alpha z}\delta_{\beta x}\right)+\\
& \left\langle v'w'\right\rangle\left(\delta_{\alpha x}\delta_{\beta z}+\delta_{\alpha z}\delta_{\beta y}\right)\Big\}
\end{aligned}
$$

The components of this tensor can be easily evaluated, leading to:

$$
\Delta\Pi^{(0)}\equiv\left(\sum_i e_{i\alpha}e_{i\beta}\Gamma_i\right)=R\begin{pmatrix}\left\langle u'^2\right\rangle & \left\langle u'v'\right\rangle & \left\langle u'w'\right\rangle\\ \left\langle u'v'\right\rangle & \left\langle v'^2\right\rangle & \left\langle v'w'\right\rangle\\ \left\langle u'w'\right\rangle & \left\langle v'w'\right\rangle & \left\langle w'^2\right\rangle\end{pmatrix} \tag{96}
$$

3.5.2 Correction for $\Pi^{(1)}_{\alpha\beta}$

The correction in the $\mathbf{1}^{st}$-order momentum flux tensor reads:

$$
\Delta\Pi^{(1)}_{\alpha\beta}=\Pi^{(1)}_{\alpha\beta}\Big|_{RALB}-\Pi^{(1)}_{\alpha\beta}\Big|_{LB}\equiv\sum_i e_{i\alpha}e_{i\beta}\,F^{(1)}_i\Big|_{RALB}-\sum_i e_{i\alpha}e_{i\beta}\,F^{(1)}_i\Big|_{LB} \tag{97}
$$

It is not straightforward how to compute this term exactly. This is due to the fact that, while it is easy to separate the non-equilibrium contribution from the equilibrium one, it is not easy to distinguish between the corrections forming the non-equilibrium part. As in the case of the standard CE analysis (see 6), we use the approximation[25]:

$$
F^{(1)}_i\approx F^{neq}_i\equiv F_i-F^{(0)}_i \tag{98}
$$

Since the only change in $\tilde{\mathbf{\Pi}}^{(1)}$ with our model is the additional term in the equilibrium distribution function, we have:

$$
\Delta\Pi^{(1)}_{\alpha\beta}\approx-\sum_i e_{i\alpha}e_{i\beta}\Gamma_i=-\Delta\Pi^{(0)}_{\alpha\beta} \tag{99}
$$

[25]This approximation is often encountered in the literature, see for example [Krueger et al., 2009]

Plugging Eq. [96] and [99] into [81], we notice that the momentum equation at the macroscale gains an additional term:

$$\boldsymbol{\Delta}\left\{\boldsymbol{\nabla}\cdot\left[\tilde{\boldsymbol{\Pi}}^{(0)}+\left(1-\frac{\mathbf{1}}{\boldsymbol{2\tau}}\right)\tilde{\boldsymbol{\Pi}}^{(1)}\right]\right\}=\boldsymbol{\nabla}\cdot\left[\boldsymbol{\Delta}\tilde{\boldsymbol{\Pi}}^{(0)}+\left(1-\frac{\mathbf{1}}{\boldsymbol{2\tau}}\right)\boldsymbol{\Delta}\tilde{\boldsymbol{\Pi}}^{(1)}\right]=\frac{\mathbf{1}}{\boldsymbol{2\tau}}\boldsymbol{\nabla}\cdot\boldsymbol{\Delta}\tilde{\boldsymbol{\Pi}}^{(0)} \tag{100}$$

Hence, the new macroscopic momentum equation for our RALB model is:

$$\begin{aligned}\frac{\boldsymbol{\partial RU_\alpha}}{\boldsymbol{\partial t}}+\sum_{\boldsymbol{\beta}}\boldsymbol{U_\beta}\frac{\boldsymbol{\partial RU_\alpha}}{\boldsymbol{\partial x_\beta}} \;=\; &-\frac{\boldsymbol{\partial P}}{\boldsymbol{\partial x_\alpha}}+\boldsymbol{\nu}\sum_{\boldsymbol{\beta}}\frac{\boldsymbol{\partial}}{\boldsymbol{\partial x_\beta}}\left[\boldsymbol{R}\left(\frac{\boldsymbol{\partial U_\beta}}{\boldsymbol{\partial x_\alpha}}+\frac{\boldsymbol{\partial U_\alpha}}{\boldsymbol{\partial x_\beta}}\right)\right]- \\ &\frac{\boldsymbol{R}}{\boldsymbol{2\tau}}\boldsymbol{\nabla}\cdot\begin{pmatrix}\langle\boldsymbol{u'^2}\rangle & \langle\boldsymbol{u'v'}\rangle & \langle\boldsymbol{u'w'}\rangle\\ \langle\boldsymbol{u'v'}\rangle & \langle\boldsymbol{v'^2}\rangle & \langle\boldsymbol{v'w'}\rangle\\ \langle\boldsymbol{u'w'}\rangle & \langle\boldsymbol{v'w'}\rangle & \langle\boldsymbol{w'^2}\rangle\end{pmatrix}\end{aligned} \tag{101}$$

which is the same expression as the RANS formulation if we absorb the denominator $\boldsymbol{2\tau}$ in the last term into redefined turbulent second-order moments.

4 Conclusions and Outlook

The current work can be divided roughly into two sections. In the first part, a three-dimensional implementation of the Lattice Boltzmann algorithm was developed, including many of the improvements proposed in the literature in the recent years with respect to boundary conditions, inclusion of forcing terms, subgrid-scale modelling and algorithm optimizations. The algorithm was tested and found to be in good quantitative agreement with analytic solutions for the three-dimensional variant of the Poiseuille flow problem. Also, the ability of the Smagorinsky turbulence model to stabilize the flow was illustrated qualitatively through turbulent simulations of the classical lid-driven $\boldsymbol{3D}$ cavity problem.

However, the implementation is still not easily applicable to oceanic flows, mostly due to the fact that LBM was studied mostly for the case of fluids with no preferential stratification, which is hardly the case in oceanography. This revealed the need for a systematic method for incorporating common oceanography-specific turbulence closure models into the framework of LBM. This problem was therefore the second major topic of the work. A successful procedure was developed, based on the Reynolds-averaging of the discretized Boltzmann equation. In this new model, the second-order turbulent moments appear through an additional term in the expression of the equilibrium distribution functions. Also included was the proof (using the Chapman-Enskog procedure) that the new model (RALBM) recovers the Navier-Stokes equations with arbitrary turbulence parametrizations in the appropriate limits. The derivation of the macroscopic equations for RALBM also illustrated the need to scale the second-order turbulent fluxes before incorporating them in LBM.

A natural extension of our work would be to use the new RALBM model in conjunction with various turbulence parametrizations. From the many options available in the literature, one can choose the simple downgradient parametrizations or the ones due to [Mellor and Yamada, 1982]. These would allow the simulation of the three-dimensional western boundary current intensification, which was only possible in the two-dimensional case using standard LBM due to the lack of the anisotropic eddy diffusivities.

Another aspect that remains to be investigated would be the enhancement of the model to include scalar fields in the Boussinesq approximation. LBM models for the

advection-diffusion equation exist [Wolf-Gladrow, 2000], and the same Reynolds-averaging procedure should be in principle applicable to incorporate parametrizations for the heat and salinity fluxes also.

As it was already mentioned, the classical bounce-back scheme has limited accuracy, especially when applied to curved boundaries. One of the causes for this is the fact that the physical (in general - curved) bounding domain is essentially approximated by a set of cubes during the initial phase of geometry definition. This unphysical "ruggedness" of the boundary increases the drag at the boundary and, according to Newton's law of reciprocal action, also affects the flow of the fluid. Therefore, a proper treatment of the actual domain geometry is recommended. The problem with this is that most often the location of the solid boundary does not coincide with a lattice grid point. This difficulty is usually addressed with interpolation schemes (see for example [Guo and Zheng, 2002]).

In its original form, the LBM works on uniform cartesian grids, with equal spacing in each of the spatial directions. However, an efficient solver should include the ability to choose a different spatial resolution in some directions, and also to specify whole regions with refined grids. The first requirement is quite relevant in ocean simulations, where the fluid domain has a vertical scale much smaller than the horizontal scale: if the vertical transport is to be modeled with any degree of accuracy, the grid resolution is to be chosen in the order of $\sim \mathbf{100}\boldsymbol{m}$, which requires a prohibitive computational cost if isotropic cartesian grids are used. A promising approach to address this kind of problem was published in [Shu et al., 2001], who used Taylor expansions to construct interpolation schemes for non-uniform grids. The authors only discussed $\mathbf{2}\boldsymbol{D}$ cases, therefore our task would be to extend the treatment to three dimensions. Although the interpolations themselves increase the computational cost per lattice point, the efficiency of the solver would be significantly increased due to the possibility of employing larger spatial scales in the vertical directions.

5 Acknowledgements

The completion of this work would not have been possible without the help of many persons. I would especially like to thank Prof. Gerrit Lohmann for the complete support and encouragement and for creating an environment where I could refine and follow my scientific interests. Also, Dr. Sergey Danilov has helped me in many ways, through the invaluable discussions and pen-and-pencil lessons which helped me observe the bigger picture and often offered pointers to further study that I may have missed otherwise. Prof. Dirk Olbers and Prof. Dieter Wolf-Gladrow and Prof. Manfred Mudelsee were kind to review parts of my work and offer helpful comments.

6 Appendix: Evaluation of stress tensor in CE expansion

In the present section, we offer a detailed derivation of the equilibrium and $\mathbf{1}^{st}$-order perturbation of the momentum flux tensor. Certainly, these calculations can bear no claim for originality. However, we believe that such an exposition is useful (especially for the newcomers to the field), as it is often only cursorily discussed in the literature. In addition, the derivation is also relevant to our Reynolds-averaged LB model, as it indicates how the changes to the LB evolution equation influence the resulting macroscopic equations.

6.1 Calculation of equilibrium stress tensor $\Pi^{(0)}_{\alpha\beta}$

$$
\begin{aligned}
\Pi^{(0)}_{\alpha\beta} \equiv \sum_i e_{i\alpha} e_{i\beta} f_i^{(0)} = \\
\rho \Bigg\{ \underbrace{\boxed{\textstyle\sum_i e_{i\alpha} e_{i\beta} t_i}}_{\equiv A_{\alpha\beta}} + \\
\underbrace{\boxed{3\left[u\sum_i e_{i\alpha}e_{i\beta}e_{ix}t_i + v\sum_i e_{i\alpha}e_{i\beta}e_{iy}t_i + w\sum_i e_{i\alpha}e_{i\beta}e_{iz}t_i\right]}}_{\equiv B_{\alpha\beta}} + \\
\underbrace{\boxed{\begin{aligned}&\frac{9}{2}\Big[u^2\sum_i e_{i\alpha}e_{i\beta}e_{ix}e_{ix}t_i + v^2\sum_i e_{i\alpha}e_{i\beta}e_{iy}e_{iy}t_i + w^2\sum_i e_{i\alpha}e_{i\beta}e_{iz}e_{iz}t_i + \\ &2uv\sum_i e_{i\alpha}e_{i\beta}e_{ix}e_{iy}t_i + 2uw\sum_i e_{i\alpha}e_{i\beta}e_{ix}e_{iz}t_i + 2vw\sum_i e_{i\alpha}e_{i\beta}e_{iy}e_{iz}t_i\Big] +\end{aligned}}}_{\equiv C_{\alpha\beta}} \\
\underbrace{\boxed{-\frac{3}{2}\vec{u}^2\sum_i e_{i\alpha}e_{i\beta}t_i}}_{\equiv D_{\alpha\beta}} \Bigg\}
\end{aligned}
\tag{102}
$$

Next, evaluate the sub-expressions:

$$
\begin{aligned}
\boldsymbol{A_{\alpha\beta}} &= \sum_i \boldsymbol{e_{i\alpha}e_{i\beta}t_i} = \boldsymbol{c_s^2\delta_{\alpha\beta}} = \boldsymbol{\frac{\delta_{\alpha\beta}}{3}} \\
\boldsymbol{B_{\alpha\beta}} &= \boldsymbol{0}\ (\boldsymbol{3^{rd}}\text{-order moments cancel by definition on the lattice}) \\
\boldsymbol{C_{\alpha\beta}} &= \boldsymbol{\frac{1}{2}\left[u^2(\delta_{\alpha\beta}+2\delta_{\alpha x}\delta_{\beta x})+v^2(\delta_{\alpha\beta}+2\delta_{\alpha y}\delta_{\beta y})+w^2(\delta_{\alpha\beta}+2\delta_{\alpha z}\delta_{\beta z})\right]+} \\
&\quad \boldsymbol{uv(\delta_{\alpha x}\delta_{\beta y}+\delta_{\alpha y}\delta_{\beta x})+uw(\delta_{\alpha x}\delta_{\beta z}+\delta_{\alpha z}\delta_{\beta x})+vw(\delta_{\alpha y}\delta_{\beta z}+\delta_{\alpha z}\delta_{\beta y})} \\
\boldsymbol{D_{\alpha\beta}} &= \boldsymbol{-\frac{1}{2}(u^2+v^2+w^2)\delta_{\alpha\beta}}
\end{aligned}
$$

Plugging these into Eq. (102), we obtain:

$$
\boxed{\boldsymbol{\Pi^{(0)}_{\alpha\beta} = \frac{\rho}{3}\delta_{\alpha\beta} + \rho u_\alpha u_\beta}} \tag{103}
$$

6.2 Calculation of 1^{st}-order perturbation of the stress tensor $\boldsymbol{\Pi^{(1)}_{\alpha\beta}}$

For evaluating this term, we need an estimate for $\boldsymbol{f_i^{(1)}}$. Using Eq. (70) into Eq. (80):

$$
\begin{aligned}
\boldsymbol{\Pi^{(1)}_{\alpha\beta}} &= \boldsymbol{-\tau\left\{\sum_i e_{i\alpha}e_{i\beta}\frac{\partial f_i^{(0)}}{\partial t_1} + \sum_i e_{i\alpha}e_{i\beta}\sum_\gamma e_{i\gamma}\frac{\partial f_i^{(0)}}{\partial x_\gamma}\right\}} \\
&= \boldsymbol{-\tau\Bigg\{\frac{\partial \Pi^{(0)}}{\partial t_1} + \frac{\partial}{\partial x_1}\underbrace{\sum_i e_{i\alpha}e_{i\beta}e_{ix}f_i^{(0)}}_{\equiv F_1} +} \\
&\quad \boldsymbol{\frac{\partial}{\partial y_1}\underbrace{\sum_i e_{i\alpha}e_{i\beta}e_{iy}f_i^{(0)}}_{\equiv F_2} + \frac{\partial}{\partial z_1}\underbrace{\sum_i e_{i\alpha}e_{i\beta}e_{iz}f_i^{(0)}}_{\equiv F_3}\Bigg\}}
\end{aligned} \tag{104}
$$

The $\boldsymbol{3^{rd}}$-order moments of can be easily computed using the expression for $\boldsymbol{f_i^{(0)}}$. We only show in detail the calculation of $\boldsymbol{F_1}$, and only state the end results for the other two moments.

$$
\begin{aligned}
\boldsymbol{F_1} &\equiv \boldsymbol{\sum_i e_{i\alpha}e_{i\beta}e_{ix}f_i^{(0)}} \\
&= \boldsymbol{\sum_i e_{i\alpha}e_{i\beta}e_{ix}\left\{1+3(\vec{e_i}\cdot\vec{u})+\frac{9}{2}(\vec{e_i}\cdot\vec{u})^2-\frac{3}{2}\vec{u}^2\right\}\rho t_i}
\end{aligned}
$$

Further expansions yield:

$$\begin{aligned} \boldsymbol{F_1} \;=\; & \boldsymbol{\rho} \underbrace{\sum_i \boldsymbol{e_{i\alpha} e_{i\beta} e_{ix} t_i}}_{\equiv 0} + \boldsymbol{3\rho} \Big\{ \boldsymbol{u} \underbrace{\sum_i \boldsymbol{e_{i\alpha} e_{i\beta} e_{ix} e_{ix} t_i}}_{\boldsymbol{1/9(\delta_{\alpha\beta} + 2\delta_{\alpha x}\delta_{\beta x})}} + \\ & \boldsymbol{v} \underbrace{\sum_i \boldsymbol{e_{i\alpha} e_{i\beta} e_{ix} e_{iy} t_i}}_{\boldsymbol{1/9(\delta_{\alpha x}\delta_{\beta y} + \delta_{\alpha y}\delta_{\beta x})}} + \boldsymbol{w} \underbrace{\sum_i \boldsymbol{e_{i\alpha} e_{i\beta} e_{ix} e_{iz} t_i}}_{\boldsymbol{1/9(\delta_{\alpha x}\delta_{\beta z} + \delta_{\alpha z}\delta_{\beta x})}} + \\ & \underbrace{\cdot \qquad\qquad \cdot \qquad\qquad \cdot}_{\boldsymbol{3^{rd}} \text{ and } \boldsymbol{5^{th}} \text{ moments, which cancel due to symmetries}} \Big\} \end{aligned} \tag{105}$$

Finally, we obtain

$$\boldsymbol{F_1} = \frac{\boldsymbol{\rho}}{\boldsymbol{3}} \begin{pmatrix} \boldsymbol{3u} & \boldsymbol{v} & \boldsymbol{w} \\ \boldsymbol{v} & \boldsymbol{u} & \boldsymbol{0} \\ \boldsymbol{w} & \boldsymbol{0} & \boldsymbol{u} \end{pmatrix} \tag{106}$$

and after more (similar) algebra

$$\boldsymbol{F_2} = \frac{\boldsymbol{\rho}}{\boldsymbol{3}} \begin{pmatrix} \boldsymbol{v} & \boldsymbol{u} & \boldsymbol{0} \\ \boldsymbol{u} & \boldsymbol{3v} & \boldsymbol{w} \\ \boldsymbol{0} & \boldsymbol{w} & \boldsymbol{v} \end{pmatrix} \tag{107}$$

$$\boldsymbol{F_3} = \frac{\boldsymbol{\rho}}{\boldsymbol{3}} \begin{pmatrix} \boldsymbol{w} & \boldsymbol{0} & \boldsymbol{u} \\ \boldsymbol{0} & \boldsymbol{w} & \boldsymbol{v} \\ \boldsymbol{u} & \boldsymbol{v} & \boldsymbol{3w} \end{pmatrix} \tag{108}$$

Plugging these expressions back into Eq. (104), we obtain:

$$\boldsymbol{\Pi^{(1)}} = -\boldsymbol{\tau} \Big\{ \underbrace{\frac{\boldsymbol{\partial}}{\boldsymbol{\partial t_1}} \Big[\frac{\boldsymbol{\rho}}{\boldsymbol{3}} \begin{pmatrix} \boldsymbol{1} & \boldsymbol{0} & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{1} & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{0} & \boldsymbol{1} \end{pmatrix}}_{\boldsymbol{= 0\ (\partial_{t_1}\rho = 0)}} + \boldsymbol{\rho} \begin{pmatrix} \boldsymbol{u^2} & \boldsymbol{uv} & \boldsymbol{uw} \\ \boldsymbol{uv} & \boldsymbol{v^2} & \boldsymbol{vw} \\ \boldsymbol{uw} & \boldsymbol{vw} & \boldsymbol{w^2} \end{pmatrix} \Big] + \frac{\boldsymbol{\rho}}{\boldsymbol{3}} \boldsymbol{\tilde{M}} \Big\}, \tag{109}$$

where by $\boldsymbol{\tilde{M}}$ we have denoted

$$\boldsymbol{M} = \begin{pmatrix} \boldsymbol{3\frac{\partial u}{\partial x_1} + \frac{\partial v}{\partial y_1} + \frac{\partial w}{\partial z_1}} & \boldsymbol{\frac{\partial v}{\partial x_1} + \frac{\partial u}{\partial y_1}} & \boldsymbol{\frac{\partial w}{\partial x_1} + \frac{\partial u}{\partial z_1}} \\ \boldsymbol{\frac{\partial v}{\partial x_1} + \frac{\partial u}{\partial y_1}} & \boldsymbol{\frac{\partial u}{\partial x_1} + 3\frac{\partial v}{\partial y_1} + \frac{\partial w}{\partial z_1}} & \boldsymbol{\frac{\partial w}{\partial y_1} + \frac{\partial v}{\partial z_1}} \\ \boldsymbol{\frac{\partial w}{\partial x_1} + \frac{\partial u}{\partial z_1}} & \boldsymbol{\frac{\partial w}{\partial y_1} + \frac{\partial v}{\partial z_1}} & \boldsymbol{\frac{\partial u}{\partial x_1} + \frac{\partial v}{\partial y_1} + 3\frac{\partial w}{\partial z_1}} \end{pmatrix} \tag{110}$$

In the incompressible limit, $\boldsymbol{\nabla_1 \cdot \vec{u} \approx 0}$, hence

$$\boldsymbol{M_{\alpha\beta} \approx \left(\frac{\partial u_\beta}{\partial x_{\alpha 1}} + \frac{\partial u_\alpha}{\partial x_{\beta 1}} \right)} \tag{111}$$

We then have

$$\boldsymbol{\Pi^{(1)}_{\alpha\beta}} = -\boldsymbol{\tau} \left\{ \frac{\boldsymbol{\delta_{\alpha\beta}}}{\boldsymbol{3}} \underbrace{\frac{\boldsymbol{\partial \rho}}{\boldsymbol{\partial t_1}}}_{\boldsymbol{\approx 0}} + \boldsymbol{u_\alpha u_\beta} \underbrace{\frac{\boldsymbol{\partial \rho}}{\boldsymbol{\partial t_1}}}_{\boldsymbol{\approx 0}} + \boldsymbol{\rho} \underbrace{\frac{\boldsymbol{\partial u_\alpha u_\beta}}{\boldsymbol{\partial t_1}}}_{\boldsymbol{\approx 0} \text{ (low Ma)}} + \frac{\boldsymbol{\rho}}{\boldsymbol{3}} \left(\frac{\boldsymbol{\partial u_\beta}}{\boldsymbol{\partial x_{\alpha 1}}} + \frac{\boldsymbol{\partial u_\alpha}}{\boldsymbol{\partial x_{\beta 1}}} \right) \right\}$$

hence

$$\boxed{\boldsymbol{\Pi^{(1)}_{\alpha\beta} \approx -\frac{\rho\tau}{3} \left(\frac{\partial u_\beta}{\partial x_{\alpha 1}} + \frac{\partial u_\alpha}{\partial x_{\beta 1}} \right)}} \tag{112}$$

References

[Bhatnagar et al., 1954] Bhatnagar, P., Gross, E. P., and Krook, M. K. (1954). A model for collision process in gases. i. small amplitude processes in charged and neutral one-component system. *Phys. Rev*, 94:511.

[Boltzmann, 1995] Boltzmann, L. (1995). *Lectures on Gas Theory.* Dover Publ. New York. ISBN 978-0486684550.

[Burchard, 2002] Burchard, H. (2002). Applied turbulence modelling in marine waters. *Lecture Notes In Earth Sciences, 100.*

[Canuto, 1994] Canuto, V. M. (1994). Large-eddy simulation of turbulence - a subgrid scale-model including shear, vorticity, rotation, and buoyancy. *Astrophysical Journal*, 428(2, Part 1):729–752.

[Cercignani, 1987] Cercignani, C. (1987). *The Boltzmann equation and its applications.* Springer New York. ISBN 978-0387966373.

[Cercignani, 1990] Cercignani, C. (1990). *Mathematical methods in kinetic theory.* Plenum, 2 edition. ISBN 978-0306434600.

[Chen and Doolen, 1998] Chen, S. and Doolen, G. D. (1998). Lattice Boltzmann method for fluid flows. *Annu. Rev. Fluid Mech.*, 30:329–364.

[Chen and Martinez, 1996] Chen, S. and Martinez, D. (1996). On boundary conditions in lattice Boltzmann methods. *Phys. Fliuds*, 8(9):2527–2536.

[Cornubert et al., 1991] Cornubert, R., d'Humieres, D., and Levermore, D. (1991). A knudsen layer theory for lattice gases. *Physica D*, 47:241.

[Dahlquist and Björck, 2008] Dahlquist, G. and Björck, Å. (2008). *Numerical Methods in Scientific Computing*, volume 1. Society for Industrial and Applied Mathematics, Philadelphia, PA, USA.

[d'Humieres et al., 2001] d'Humieres, D., Bouzidi, M., and Lallemand, P. (2001). Thirteen-velocity three-dimensional lattice boltzmann model. *PRE*, 63(6, Part 2).

[d'Humieres et al., 2002] d'Humieres, D., Ginzburg, I., Krafczyk, M., Lallemand, P., and Luo, L. S. (2002). Multiple-relaxation-time lattice Boltzmann models in three dimensions. *Phil. Trans. R. Soc. A*, 360:437–451.

[Ferziger and Kaper, 1972] Ferziger, J. H. and Kaper, H. G. (1972). *Mathematical theory of transport processes in gases.* North-Holland Publishing Company. ISBN 0-7204-2046-6.

[Frisch, 1995] Frisch, U. (1995). *Turbulence: The Legacy of A. N. Kolmogorov.* Cambridge University Press. ISBN 0521457130.

[Frisch, 1996] Frisch, U. (1996). *Turbulence: the legacy of A.N. Kolmogorov.* ISBN 0-521-45103-5.

[Guo and Zhao, 2002] Guo, Z. and Zhao, T. S. (2002). Lattice Boltzmann model for incompressible flows through porous media. *Phys. Rev. E*, 66:66304.

[Guo and Zheng, 2002] Guo, Z. and Zheng, C. (2002). An extrapolation method for boundary conditions in lattice Boltzmann method. *Phys. Fliuds*, 14(6):2007–2010.

[He and Luo, 1997] He, X. and Luo, L. S. (1997). Theory of the lattice Boltzmann method: From the Boltzmann equation to the lattice Boltzmann equation. *Phys. Rev. E*, 56(6):6811–6817.

[Hou et al., 1994] Hou, S., Sterling, J., Chen, S., and Doolen, G. D. (1994). A lattice boltzmann subgrid model for high reynolds number flows.

[Huang, 1987] Huang, K. (1987). *Statistical Mechanics (2nd edition).* Wiley & Sons. ISBN 978-0471815181.

[Krueger et al., 2009] Krueger, T., Varnik, F., and Raabe, D. (2009). Shear stress in lattice boltzmann simulations. *PRE*, 79(4, Part 2).

[Landau and Lifshitz, 1987] Landau, L. D. and Lifshitz, E. M. (1987). *Course of Theoretical Physics: Fluid Mechanics (Volume 6).* Butterworth-Heinemann, 2 edition. ISBN 978-0750627672.

[Latt, 2007] Latt, J. (2007). *Hydrodynamics limit of lattice Boltzmann equation.* PhD thesis, University of geneva.

[Latt, 2008] Latt, J. (2008). Choice of units in lattice boltzmann simulations. Technical report, The Institute of Mechanical Engineering of the Ecole Polytechnique Federale de Lausanne.

[Latt and Chopard, 2008] Latt, J. and Chopard, B. (2008). Straight velocity boundaries in the lattice Boltzmann method. *Phys. Rev. E*, 77:056703–1:16.

[Mellor and Yamada, 1982] Mellor, G. L. and Yamada, T. (1982). Development of a turbulence closure model for geophysical fluid problems. *Rev. Geophys. Space Phys.*, 20:851–875.

[Mohammadi and Pironneau, 1994] Mohammadi, B. and Pironneau, O. (1994). *Analysis of the K-epsilon turbulence model.* John Wiley, Chichester.

[Orszag, 1977] Orszag, S. A. (1977). Statistical theory of turbulence. *Fluid Dynamics 1973*, pages 237–374.

[Palmer and Hagedorn, 2006] Palmer, T. and Hagedorn, R. (2006). *Predictability of Weather and Climate.* Cambridge University Press. ISBN 0521848822.

[Pan et al., 2006] Pan, C., Luo, L. S., and Miller, C. T. (2006). An evaluation of lattice Boltzmann schemes for porous medium flow simulation. *Comput. Fluids*, 35:898–909.

[Salmon, 1999a] Salmon, R. (1999a). The lattice Boltzmann method as a basis for ocean circulation modeling. *J. Marine Res.*, 57:503–535.

[Salmon, 1999b] Salmon, R. (1999b). Lattice Boltzmann solutions of the three-dimensional planetary geostrophic equations. *J. Marine Res.*, 57:847–884.

[Shu et al., 2001] Shu, C., Chew, Y. T., and Niu, X. D. (2001). Least-squares-based lattice Boltzmann method: A meshless approach for simulation of flows with complex geometry. *Phys. Rev. E*, 64:045701.

[Skordos, 1993] Skordos, P. A. (1993). Initial and boundary conditions for the lattice Boltzmann method. *Phys. Rev. E*, 48(6):4823–4842.

[Smagorinsky, 1963] Smagorinsky, J. (1963). General circulation experiments with the primitive equations. *Mon. Wea. Rev.*, 91:99–164.

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

[Sukop and Thorne, 2006] Sukop, M. and Thorne, D. T. (2006). *Lattice Boltzmann Modeling: an introduction for geoscientists and engineers*. Springer Verlag, 1 edition. ISBN 978-3-540-27981-5.

[Thuerey, 2007] Thuerey, N. (2007). *Physically based animation of free surface flows with the Lattice Boltzmann Method*. PhD thesis, Der Technischen Fakultaet der Universitaet Erlangen-Nuernberg.

[Wolf-Gladrow, 2000] Wolf-Gladrow, D. (2000). *Lattice-Gas Cellular Automata and Lattice Boltzmann Models - An Introduction*. Springer Verlag, 1 edition. ISBN 978-3-540-66973-9.

[Yu et al., 2005] Yu, H., Girimaji, S., and Luo, L. S. (2005). Lattice Boltzmann simulations of decaying homogeneous isotropic turbulence. *Phys. Rev. E*, 71:016708–1:5.

[Ziegler, 1993] Ziegler, D. P. (1993). Boundary conditions for the lattice Boltzmann simulations. *J. Stat. Phys.*, 71:1171.

[Zou and He, 1997] Zou, Q. and He, X. (1997). On pressure and velocity boundary conditions for the lattice Boltzmann bgk model. *Phys. Fliuds*, 9(6):1591–1598.