



UNIVERSIDAD
NACIONAL
DE COLOMBIA

A LATTICE-BOLTZMANN MODEL FOR THE ADVECTION-DIFFUSION EQUATION IN GENERALIZED COORDINATES

Juliana García Sarmiento

Universidad Nacional de Colombia
Facultad de Ciencias, Departamento de Física
Bogotá, Colombia
2019

A LATTICE-BOLTZMANN MODEL FOR THE ADVECTION-DIFFUSION EQUATION IN GENERALIZED COORDINATES

Juliana García Sarmiento

Thesis presented as a partial requirement to obtain the degree of:
M.Sc. in Physics

Advisor:
Dr. Rer. Nat. José Daniel Muñoz Castaño

Research line:
Computational Physics
Research Group:
Simulation of Physical Systems - SSF UN

Universidad Nacional de Colombia
Facultad de Ciencias, Departamento de Física
Bogotá, Colombia
2019

*A mis padres y familia, por apoyarme siempre
y a mis abuelos por acompañarme desde donde
estén*

*“One thing I have learned in a long life: That
all our science, measured against reality, is
primitive and childlike - and yet it is the most
precious thing we have.”*

Albert Einstein

Acknowledgements

First of all, I thank Professor José Daniel Muñoz Castaño for giving me the opportunity to belong to the Physical Systems Simulation group (SSF+UN). I feel a great appreciation and a deep admiration towards him, which has made working with him a great experience. I also thank Miller Mendoza for his brilliant ideas for this work and for making my stay at the ETH, Zürich an unforgettable experience.

I thank Professor Hans J. Herrmann and all Comphys group at the Institute for Building Materials, ETH Zürich, for hosting me and helping during the six months of my internship and for enriching my training both professionally and personally.

I thank all the members of the Physical Systems Simulation Group (SSF+UN) for having shared so many things and for the discussions that in some way enriched the understanding of this problem.

I thank my friends and colleagues for having accompanied me and supported me on this path, for giving me the strength to continue and for sharing so many valuable moments. Special thanks to Natalia, Lina, Esteban, Juan Pablo, Wilmar, Ali Mauricio, Manuel and Luis, for being there from the beginning. Thanks to those who in any way helped me make this work better. Thanks to Robin, Ilario and Damian for sharing the office with me in Zürich.

Last but not least, I thank my parents, for always being there, for always supporting me unconditionally, for guiding me and for giving me the strength I needed. Thank you both for teaching me how to knit my own path.

Agradecimientos

En primer lugar, agradezco al profesor José Daniel Muñoz Castaño por darme la oportunidad de pertenecer al grupo de Simulación de Sistemas Físicos (SSF+UN). Siento un gran aprecio y una profunda admiración hacia él, lo que ha hecho que trabajar con él sea una gran experiencia. Agradezco también a Miller Mendoza por sus brillantes ideas y por hacer de mi estancia en el ETH, Zürich una experiencia inovidable.

Agradezco al Profesor Hans J. Herrmann y al grupo Comphys en el Institute for Building Materials, ETH Zürich, por acogerme y ayudarme financieramente durante los seis meses de mi pasantía y por enriquecer mi formación tanto profesional como personalmente.

Agradezco a todos los miembros del Grupo de Simulación de Sistemas Físicos (SSF+UN) por haber compartido tantas cosas y por las discusiones que de alguna manera enriquecieron el entendimiento de este problema.

Agradezco a mis amigos y compañeros por haberme acompañado y apoyado en este camino, por darme fuerzas para seguir y por compartir tantos momentos valiosos. Gracias especiales a Natalia, Lina, Esteban, Juan Pablo, Wilmar, Ali Mauricio, Manuel y a Luis, por estar desde el comienzo. Gracias a aquellos que me ayudaron a que este trabajo fuera mejor. Gracias a Robin, a Ilario y a Damian por compartir la oficina conmigo en Zürich.

Finalmente pero no menos importante, agradezco a mis padres, por estar siempre allí, por apoyarme incondicionalmente siempre, por orientarme y por darme la fortaleza que necesité. Gracias a ambos por enseñarme a ir tejiendo mi propio camino. Por ellos y para ellos es este logro.

Abstract

Lattice-Boltzmann models (LBM) are very powerful simulation techniques for fluid dynamics, diffusion processes, mechanical waves, magneto- and electrodynamics. However, one of the main complications when working with these LBM is the necessity of an accurate implementation of the boundary conditions. Ideally, the boundaries are rectangular and parallel to the computational mesh, but most of the times, real-life problems have complex geometries and, therefore, boundaries are not easy to implement. This work develops an alternative lattice-Boltzmann model to reproduce the Advection-Diffusion Equation (ADE) on generalized coordinates in two and three dimensions. Our model introduces the geometry as a source term, which makes it much easier and more flexible to simulate curved geometries in two and three dimensions like disks, cylinders, torii, sinusoidal curved channels and any complex shape that can be described as an orthogonal coordinate transformation. The proposed LBM, which shows second-order accuracy, allows also to perform mesh refinements without losing isotropy, to avoid staircase approximations and to take advantage of geometrical symmetries, when present.

Our simulation results are in excellent agreement with the theoretical predictions in all studied cases in two and three dimensions, with or without symmetries, and even reproduce with great accuracy experimental results. In fact, we have defined our model in such a way that it facilitates to deal with real physical units (like centimeters, seconds, etc) - something that is not obvious when dealing with non-uniform cell sizes -, making easier the comparison with experimental data. Our model can be used on a broad range of applications, like heat diffusion in complex geometries, pollutant spreading in channels or pipes, and sediment transport in rivers. Because each geometry is defined by few parameters, and those parameters can be time-dependent, our model could be also used to simulate the ADE on time-varying geometries, like pulsing blood vessels, by computing our model with a Navier-Stokes equations solver with changing boundary conditions. This work contains a valuable contribution for the study of advection-diffusion phenomena. Because this phenomenon is relevant to many scientific, industrial and environmental applications, we expect it would be of great usefulness in future research.

Keywords: Lattice Boltzmann models, Advection-diffusion Processes, Generalized Coordinates.

Resumen

Los modelos Lattice-Boltzmann (LBM) son técnicas de simulación muy potentes para dinámica de fluidos, procesos de difusión, ondas mecánicas, magneto y electrodinámica. Sin embargo, una de las principales complicaciones cuando se trabaja con estos LBM es la necesidad de una implementación precisa de las condiciones de frontera. Idealmente, las fronteras son rectangulares y paralelas a la malla computacional, pero la mayoría de las veces los problemas de la vida real tienen geometrías complejas y, por lo tanto, las fronteras no son fáciles de implementar. Este trabajo desarrolla un modelo alternativo de Lattice-Boltzmann para reproducir la ecuación de Advección-Difusión (ADE) en coordenadas generalizadas en dos y tres dimensiones. Nuestro modelo presenta la geometría como un término fuente, lo que hace que sea mucho más fácil y más flexible simular geometrías curvas en dos y tres dimensiones como discos, cilindros, toros, canales sinusoidales y cualquier forma compleja que pueda describirse como una transformación de coordenadas ortogonales. El LBM propuesto, que muestra una precisión de segundo orden, también permite realizar refinamientos de malla sin perder isotropía, evitar aproximaciones en la escalera y aprovechar las simetrías geométricas, cuando están presentes.

Nuestros resultados de simulación están en excelente concordancia con las predicciones teóricas en todos los casos estudiados en dos y tres dimensiones, con o sin simetrías, e incluso se reproducen con gran precisión resultados experimentales. De hecho, hemos definido nuestro modelo de tal manera que facilita el manejo de unidades físicas reales (como centímetros, segundos, etc.), algo que no es obvio cuando se trata de tamaños de celdas no uniformes, lo que hace más fácil la comparación con datos experimentales. Nuestro modelo se puede utilizar en una amplia gama de aplicaciones como la difusión de calor en geometrías complejas, la propagación de contaminantes en canales o tuberías y el transporte de sedimentos en los ríos. Debido a que cada geometría está definida por pocos parámetros, y esos parámetros pueden depender del tiempo, nuestro modelo también se puede usar para simular la ADE en geometrías variables en el tiempo, como los vasos sanguíneos pulsantes, al acoplar nuestro modelo con otro que solucione las ecuaciones de Navier-Stokes con condiciones de frontera cambiantes. Este trabajo constituye una valiosa contribución al estudio de los fenómenos de advección-difusión. Debido a que este fenómeno es relevante para muchas aplicaciones científicas, industriales y ambientales, esperamos que sea de gran utilidad en futuras investigaciones.

Palabras clave: **Lattice Boltzmann, Procesos de Difusión y Advección, Coordenadas generalizadas.**

Scientific contributions

As part of the COLCIENCIAS project "Lattice-Boltzmann models in generalized curvilinear coordinates" (project number: 201010023099), an internship was done in the Comphys Group in the Institute for Building Materials at ETH Zürich (Switzerland), under the supervision of Dr. Miller Mendoza.

Results from this work have been presented in two scientific events as a Poster and Oral presentation:

- **2nd Spring School on Lattice Boltzmann Methods 2018**
- **27th International Conference on Discrete Simulations of Fluid Dynamics DSFD 2018**

In addition, we submitted a paper to **Journal of Computational of Physics** that is under revision.

Highlights

The highlights of this research are the following:

- We show it is possible to simulate the advection-diffusion equation (ADE) on generalized coordinates with a lattice-Boltzmann model (LBM) by including the geometry as a source term that is function of the metric tensor and Christoffel symbols.
- This model can be used in any complex shape in one, two and three dimensions that can be described with an orthogonal coordinate transformation. Actually, we test our model for the following cases: one-directional mesh stretching, a Gaussian-like mesh stretching, disks, cylinders, torii and sinusoidal curved channels.
- The proposed LBM model shows second-order accuracy when tested in the Gaussian-like mesh stretching case.
- This model allows us to increase the resolution in specific regions without losing isotropy, what is exemplified by a Gaussian-like mesh refinement.
- The model avoids staircase approximations and allows to take advantage of geometrical symmetries, when present (for instance, in systems with axial symmetry).
- The way it is defined, our model allows to use physical dimensions easily and, therefore, makes comparisons with experimental results a trivial task. In fact, we illustrate this point by reproducing the advection and diffusion of a disk of Helium ions on a Poiseuille flow in a cylindrical pipe as a two-dimensional problem.
- Our last example, i.e. the advection-diffusion of a Gaussian initial density on a sinusoidal-like curved channel with a Poiseuille-like flow, indicates how our model could be used to simulate advection-diffusion of pollutants in a river. This is a very promising area of future work.

Contents

Acknowledgements/Agradecimientos	vii
Abstract/Resumen	xi
Scientific contributions	xv
Highlights	xvii
List of figures	xxi
1 Introduction	1
2 Advection and diffusion processes	5
2.1 Advection-diffusion phenomena	5
2.2 Non-dimensionalisation of the ADE	6
2.3 Thermal processes	7
3 Lattice Boltzmann models for the advection-diffusion equation	11
3.1 What is a Lattice Boltzmann Model?	11
3.1.1 Boltzmann transport equation	12
3.1.2 Some LBM velocity schemes	13
3.1.3 Boundary conditions	14
3.2 Lattice Boltzmann models for the advection-diffusion equation in Cartesian coordinates	15
3.2.1 Diffusion and advection of an one-dimensional density step	15
3.2.2 Diffusion and advection of a Gaussian hill	16
3.3 Adding a source term to the advection-diffusion equation with an LBM	19
3.4 Lattice Boltzmann model for the advection-diffusion equation in generalized coordinates	20
3.4.1 Why generalized coordinates?	20
3.4.2 Numerical models for the ADE in generalized coordinates	21

3.4.3	Lattice-Boltzmann models in generalized coordinates	22
3.4.4	A Multiple Relaxation Time (MRT) LBM for the ADE in curvilinear systems	23
4	An alternative lattice Boltzmann model for the advection-diffusion equation in generalized coordinates	27
4.1	Our proposal	27
4.1.1	Computation of spacial derivatives within the lattice	29
4.1.2	LBM algorithm for the ADE on generalized coordinates	29
4.2	Benchmarks	30
4.2.1	Mesh stretching	30
4.2.2	Gaussian-like stretching	32
4.2.3	Convergence study	34
4.2.4	Polar coordinates	36
5	Applications of our model: Experimental comparison, torus and curved channels	39
5.1	A comparison with experimental data	39
5.2	Torus	42
5.3	A curved channel	46
6	Conclusions	49
A	Chapman-Enskog expansion for the advection-diffusion equation with a forcing term	51
A.1	Chapman-Enskog expansion	51
	Bibliography	54

List of Figures

2-1	Pure advection (left). Pure diffusion (center). Advection and diffusion (right). Reproduced with permission from [1]	6
2-2	Temperature plot of Rayleigh-Bénard convection. Snapshots at $t = 4000$ s, $t = 8000$ s, $t = 12000$ s and $t = 16000$ s. This simulation was performed with two coupled LBM: a D2Q9 for the fluid flow and a D2Q5 for the temperature field. The relaxation times are $\tau_{D2Q9} = 0.58$ and $\tau_{D2Q5} = 0.68$. The 2D computational domain is 256×64 cells, with $\Delta x = 1$ cm and $\Delta t = 1$ s. Dirichlet boundary conditions were imposed at upper and lower plates with dimensionless temperature values: $T_u = 1$ and $T_l = 0$	8
3-1	Streaming step in a lattice-Boltzmann method illustrated using a D2Q9 velocity scheme.	12
3-2	LBM velocity schemes in three dimensions.	14
3-3	Advection and diffusion of a one-dimensional density step in a channel 100 meters long simulated with a D3Q7 scheme after 1000 seconds. The diffusion coefficient is $D = 0.05$ and the advection velocity $u_x = 0.02$. An excellent agreement with the theoretical solution is found.	16
3-4	Advection and diffusion of an initial Gaussian density on a fluid with a uniform velocity field with $\sigma_0 = 6.4$ m, $D = 0.05$ m ² /s, $u_x = u_y = 0.03$ m/s and $(x_0, y_0) = (33\text{m}, 33\text{m})$. Snapshots every 200s.	17
3-5	Advection and diffusion of an initial Gaussian density on a fluid with an uniform velocity field with $\sigma_0 = 6.4$ m, $D = 0.05$ m ² /s, $u_x = u_y = 0.03$ m/s and $(x_0, y_0) = (33\text{m}, 33\text{m})$. a) One-dimensional profile along the diagonal from $(0, 0)$ to (L_x, L_y) every 200 s. b) Variance σ^2 against time t . A linear regression gives $D = 0.04995$ m ² /s, which matches the simulation parameter $D = 0.05$ m ² /s.	18
3-6	Different domains for consider a sphere. Staircase approximation mesh (left). Spherical coordinates mesh (center). Computational domain (right).	20

4-1	Diffusion of a Gaussian density distribution on a uniformly stretched mesh with the coordinate transformation $x = 2\eta_1$, $y = \eta_2$. Computational space (Left) and physical space (Right).	31
4-2	Variance of the marginal distributions on x and y against time for the diffusion of a two-dimensional Gaussian distribution on an uniformly stretched mesh. The diffusion in x -direction presents a good matching with the non-stretched mesh up to a scaling factor of $C = 2$, while in the y direction is isotropic for all scale factors.	32
4-3	Two-dimensional Gaussian-like stretching after 1000 seconds on an uniform advection velocity field with components $u_x = u_y = 0.05$ cm/s. Evolution of a Gaussian distribution. We see the distribution distorted in the computational space (left) while in the physical space (right) it recovers the isotropic distribution. The parameters used are: $\sigma = 6.4$ cm, $x_0 = y_0 = 25.6$ cm, $D = 0.01$ cm ² /s.	34
4-4	Convergence mesh study for the Gaussian-like stretching. Measured error for the concentration versus the number of nodes. $D = 0.01\Delta x^2/\Delta t$, $u_x = u_y = 0.05 \Delta x/\Delta t$. This plot evidences the second-order accuracy of this LBM. . .	35
4-5	Evolution of a Gaussian distribution centered at $r = 0$. $D = 0.017$ cm ² /s, $\sigma_0 = 22.5$ cm, $\tau = 0.5051$, after 24000 timesteps with $\Delta t = 1$ s. a) Colorplot of the evolution of a Gaussian distribution centered at $r = 0$. b) Normalized density profile (red-dotted) and analytical solution (blue-continued) across a diameter. c) Density profile (green-dotted) and analytical solution (blue-continued) across a diameter. After renormalization, the simulated profile fits the analytical solution.	36
4-6	Evolution of a Gaussian distribution centered at $r = a/2$, $\theta = \pi$. $D = 0.017$ cm ² /s, $\sigma_0 = 22.5$ cm, $\tau = 0.5051$, after 8000 timesteps. Distribution on the circular plate (left). Profile along a diameter across the center of the Gaussian distribution (right).	37
5-1	Simulation of the advection-diffusion evolution of an Helium ion sheet inside a cylindrical tube with a Poiseuille velocity field. The simulation parameters are shown in Table 5-1 and in the text thereafter. Figure shows a) initial state, b) and c) shows the snapshots of the evolution after 0.001995 s with a Cartesian LBM model and Cylindrical LBM model, respectively.	41
5-2	Comparison between the experimental data from [2] (red), our LBM model in cylindrical coordinates (green) and an LBM in Cartesian coordinates (blue). The values used for this simulation are shown in Table 5-1.	41
5-3	Torus Geometry.	42

5-4 Evolution of a Gaussian distribution in a Torus with a Poiseuille-like velocity in θ -direction, with $V = 0.641$ cm/s, $D = 1$ cm ² /s, $\sigma = 2.1$ cm, $R = 60$ cm, $a = 21$ cm, $\Delta t = 0.0078$ s, and a simulation time of 3000 timesteps. a) $t = 0$ s, b) $t = 7.8$ s, c) $t = 15.6$ s and d) $t = 23.4$ s.	44
5-5 Evolution of a Gaussian distribution in a Torus with diffusion only. $D = 1$ cm ² /s, $\sigma = 2.1$ cm, $R = 60$ cm, $a = 21$ cm, $\Delta t = 0.0078$ s, and a simulation time of 6000 timesteps. a) $t = 0$ s, b) $t = 15.6$ s, c) $t = 31.2$ s and d) $t = 46.8$ s.	45
5-6 Geometry of the sinusoidal curved channel.	46
5-7 Evolution of a Gaussian distribution in a Sinusoidal-like channel with a Poiseuille-like velocity field along the longitudinal coordinate ζ . The parameters employed are $r_{\min} = 1$ cm, $m = 0.08$, $b = 6.5$, $\Delta r = 1$ cm, $\Delta t = 0.017$ s and a diffusion coefficient $D = 1$ cm ² /s ($D = 0.017 \Delta r^2 / \Delta t$). Snapshots from upside at a) $t = 0$ s, b) $t = 8.5$ s, c) $t = 17$ s, d) $t = 25.5$ s.	48
5-8 Evolution of a Gaussian distribution in a Sinusoidal-like channel with a Poiseuille-like velocity field along the longitudinal coordinate ζ . The parameters employed are $r_{\min} = 1$ cm, $m = 0.08$, $b = 6.5$, $\Delta r = 1$ cm, $\Delta t = 0.017$ s and a diffusion coefficient $D = 1$ cm ² /s ($D = 0.017 \Delta r^2 / \Delta t$). Snapshots at a) $t = 8.5$ s and b) $t = 17$ s.	48

CHAPTER 1

Introduction

Advection-diffusion (AD) phenomena are widely found in science and engineering, in processes as pollutant spreading in air and rivers, salt diffusion in the sea, thermal convection [3], among others. All of them include particles transported by a flux plus some random motion. To investigate those phenomena, several numerical approaches have been developed based on finite-difference, finite-volume, and finite-element methods [4, 5]. However, these methods need at each timestep the information from adjacent cells to compute the new values of the desired quantities at each cell (pressure, velocity, etc.), slowing down their efficiency when implemented on massive parallel systems and hindering their implementation when complex geometries are considered.

An alternative to solve those problems is the lattice Boltzmann Method (LBM), a discrete scheme based on the kinetic theory of fluids [6, 1]. In this method, a fluid of (real or auxiliary) particles carries the information from cell to cell and recombines it through a collision step according to the Boltzmann transport equation. Every cell stores a discrete version of Boltzmann's distribution function for the fluid. Each timestep is divided into two stages: a *collision* step, when each cell relaxes its distribution functions towards equilibrium values (by using either a single [7, 8] or multiple [9, 10, 11] relaxation times), and an *advection* step, when each cell transports the new values to the neighbouring cells. None of those steps require the information of the adjacent cells to be locally known and, therefore, the evolution can be done fully in parallel, allowing a direct implementation on graphics cards and speeding up the computational times by orders of magnitude [12]. In the macroscopic limit of infinitely small cells, the model reproduces a set of conservation laws that can be adjusted to the desired ones by manipulating the equilibrium distributions and by adding forcing terms. The Lattice Boltzmann Method (LBM), introduced in 1988 by McNamara and Zanetti [13], has made a great progress in the study of fluid dynamics, since it allows a natural inclusion of interactions among particles and with the walls in both multiphase and multicomponent flows [14, 15] and, it has been also extended to the simulation of acoustic

waves [16, 17], magnetohydrodynamics [18, 19], electrodynamics [20], general relativity [21] and quantum mechanics [22, 23], all fields far from fluid mechanics.

On regarding advection and diffusion problems, two of the first works belong to Flekkoy [24], who proposed a model to simulate osmosis and miscible liquids, and Wolf-Gladrow [25], who formulated a lattice Boltzmann Model to reproduce diffusion without advection. Later on, Guo [26] coupled the thermal diffusion to a fluid model via the Boussinesq approximation and, Dawson et al. [27] used this method to study reaction-diffusion systems. Next, Sman et al. [28] developed an LBM for irregular lattices with rectangular Wigner-Seitz cells without using any interpolation or averaging step. Later, Zhang et al.[29] developed an LBM for advection and anisotropic dispersion. More recently, a forcing term was introduced by Shi et al. [30] to recover the advection-diffusion equation with a source term. For a more detailed review, the reader can refer to [31] and [32].

One of the major complications that emerge when using LBM, appears when handling curved geometries, because most models work on cubic cells only. As a solution, M. Mendoza proposed in his Ph.D. thesis [12] a model that is able to simulate fluids virtually on any kind of geometry that can be described by a curved coordinate system. The strategy consists in keeping a cubic lattice in the computer but adding the forcing terms required to reproduce the desired conservation laws in curved coordinates. Those extra terms are expressed by means of the metric tensor and Christoffel symbols for those coordinates. In contrast with previous proposals [33, 34, 35, 36], there is no need for interpolation steps. It was successfully used to study the flow through randomly curved manifolds [37], the Dean's instability in ellipsoidal coordinates [38] and, more recently, the energy dissipation in flows through curved spaces [39] and the vibrational modes of a trumpet [40]. A similar approach was proposed almost simultaneously in two dimensions by Budinsky [41], but using the Jacobian and Jacobian spatial derivatives. A recent work by H. Yoshida and M. Nagaoka [42] uses a multiple-relaxation-time (MRT) LB model to handle the apparent anisotropy of diffusion due to the coordinate transformation, and defines the equilibrium function such that the background velocity arising from the Jacobian's derivatives is properly reproduced. This model provides a second-order accuracy solution for the advection-diffusion equation in one and two dimensions, with a relaxation times matrix in the collision term defined in terms of the geometry.

Hereby we introduce an alternative model for the simulation of the advection-diffusion equation (ADE) in generalized coordinates, both 2D and 3D, by following Mendoza's strategy. The proposal is based on the model by Shi et al. [30] on cubic cells, which reproduces in a very clean way the advection-diffusion equation with a source term up to second order accuracy by adding a differential form of that source term to the Bhatnagar-Gross-Krook (BGK) evolution rule. In our model, that source term will contain the information about

the curvilinear coordinates, and it will be written in terms of the metric tensor and the Christoffel symbols [43].

This thesis is developed as follows: In Chapter 2 we review basic concepts and theory behind advection and diffusion processes, like the non-dimensionalization of the ADE and thermal flows. Chapter 3 introduces the lattice-Boltzmann method (LBM) for ADE in Cartesian coordinates, it presents some examples in and then shows the strategy proposed by Shi et al. [30] to implement a source term for ADE in the LBM. Later, we introduce the advantages of working in generalized coordinates and reviews some of the relevant works. This chapter ends by explaining the Multiple-Relaxation-Time (MRT) LBM for the ADE on curvilinear coordinates proposed by Yoshida and Nagaoka [42], and compare how both models are defined. In Chapter 4 we introduce our proposed LBM model for the ADE in generalized coordinates, which takes into account the geometry information as a source term and we present some benchmark tests: a one-dimensional mesh-stretching, Gaussian-like mesh stretching, polar coordinates; this last case allows us to take advantage of axial symmetry. In Chapter 5 we present some applications of our model: we use the axial symmetry to simulate the advection and diffusion of a disk of Helium ions on a Poiseuille flow in a cylindrical pipe as a two-dimensional problem. This simulation reproduces perfectly the experimental results. Next, as an example of a full three-dimensions problem we simulate the advection-diffusion of a Gaussian pulse inside a torus with a Poiseuille-like velocity flow and a Gaussian pulse of contaminant in a sinusoidal curved channel that resembles a river. Conclusions, main remarks and future work are presented and discussed in Chapter 6.

CHAPTER 2

Advection and diffusion processes

In this chapter we review the main concepts related to Advection and Diffusion processes that result relevant to the development of this thesis. We begin by explaining the Advection-Diffusion Equation (ADE) and identifying the different contributions and explore other applications. Then we explain some thermal problems related. Next, we discuss about the importance of using generalized coordinates in this kind of problems and finally, we discuss about the previous numerical works done for advection-diffusion phenomena.

2.1 Advection-diffusion phenomena

Advection-diffusion problems can be divided into two main phenomena, on one hand, *Advection* is the transport driven by the velocity field of a surrounding medium, and can be found, for example, when oil is spilled in a river and is transported by the stream.

Diffusion, on the other hand, is the macroscopic effect of an uniform and time-independent random motion, usually as a consequence of the random interactions of the molecules of a surrounding medium. Those two effects summarize in the Advection Diffusion Equation (ADE),

$$\partial_t \rho + \nabla \cdot (\rho \vec{u}) = D \nabla^2 \rho + \vec{S}(\vec{x}, t) . \quad (2-1)$$

Equation (2-1) shows that, time-dependent changes of the scalar field ρ can be obeyed to three main processes:

- First, the *advection* term (that is, the second term in the left-side of the equation) driven by an external velocity \vec{u} .
- Second, the *diffusion* term (that is, the first term on the right-hand side of the equation) characterized by the diffusion coefficient D with units [Length]²/[time]. In a general

case, this diffusion term is written as $\nabla \cdot (D \nabla \rho)$, where $\vec{j} = -D \nabla \rho$ is the diffusion flux and its relation with D and ρ is known as the Fick's first law.

- The last term on the right-hand side of the equation accounts for external sources or sinks that can increase or decrease ρ , for instance a constant deposition of a contaminant in a river.

In this work we assume that the diffusion is homogeneous and isotropic within the whole physical domain.

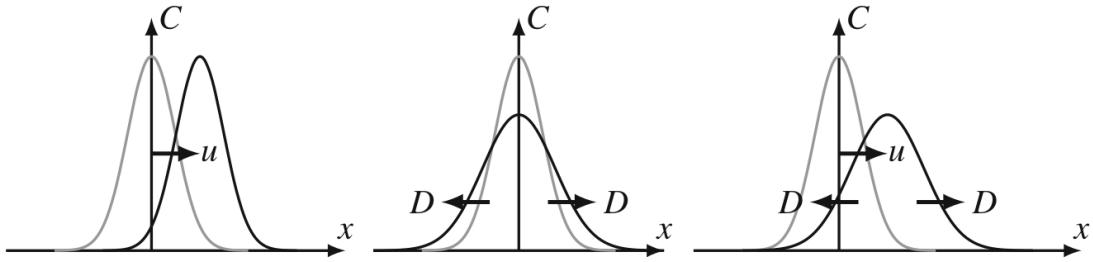


Figure 2-1: Pure advection (left). Pure diffusion (center). Advection and diffusion (right). Reproduced with permission from [1]

In many applications, more than one numerical model is needed to solve the equations for the fluid: one for the momentum and the velocity field \vec{u} and another for the scalar field ρ (not to be confused with the fluid density). If this scalar field is passive, i.e., it represents a chemical species or a pollutant that does not affect the fluid density, it is not necessary to couple both models. Otherwise, if the fluid density is affected by this scalar field, for example in thermal flows (i.e., the scalar field denoted by T), it is necessary to solve the coupled system of equations, as will be explained in Section 2.3.

2.2 Non-dimensionalisation of the ADE

Let us consider the advection-diffusion equation without a source term, written as follows:

$$\frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \vec{u}) = D \nabla^2 \rho \quad , \quad (2-2)$$

where $\nabla \equiv \frac{\partial}{\partial x_i}$ and $\nabla^2 \equiv \frac{\partial^2}{\partial x_i^2}$. In order to obtain a non-dimensional version of this equation, let us define characteristic units L_c , t_c and U_c , known as the characteristic length, time and velocity, respectively. The relation between these values and the physical quantities will be given by

$$t = t_c t^* \quad , \quad x_i = L_c x_i^* \quad , \quad \vec{U} = \vec{U}_c \vec{U}^* = \frac{L_c}{t_c} \vec{U}^* \quad , \quad (2-3)$$

where the star refers to the non-dimensionalised variable. Replacing these expressions into Equation (2-2), it yields

$$\frac{\partial \rho}{\partial(t_c t^*)} + \frac{\partial}{\partial(L_c x_i^*)}(\rho \vec{U}_c \vec{U}^*) = D \frac{\partial^2 \rho}{\partial(L_c x_i^*)^2} , \quad (2-4)$$

and after rearranging the characteristic parameters, one obtains

$$\frac{\partial \rho}{\partial(t^*)} + \frac{\partial}{\partial x_i^*}(\rho \vec{U}^*) = \frac{D}{L_c U_c} \frac{\partial^2 \rho}{\partial x_i^{*2}} . \quad (2-5)$$

where the inverse of the dimensionless quantity accompanying the diffusion term, is known as the Péclet number,

$$\text{Pe} = \frac{L_c U_c}{D} . \quad (2-6)$$

This dimensionless number can tell us if the evolution of the scalar field ρ is mainly driven by advection or diffusion. For example, if Pe is greater than 1, the contribution to the change of ρ will be mainly given to the advection (or external) velocity, rather than to the diffusivity in the medium. On the contrary, if Pe is less than one, the evolution will be dominated by diffusion. So, we obtain the adimensional ADE in terms of the Péclet number,

$$\frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \vec{u}) = \frac{1}{\text{Pe}} \nabla^2 \rho . \quad (2-7)$$

2.3 Thermal processes

Thermal problems can be also described by an advection-diffusion equation,

$$\frac{\partial T}{\partial t} + \nabla \cdot (T \vec{u}) = \kappa \nabla^2 T + S , \quad (2-8)$$

where T is the temperature and κ is the *thermal diffusivity*.

Usually, the scalar field ρ does not affect the fluid dynamics, and the velocity is often provided by a Navier-Stokes solver (could be also LBM) or an analytical solution, where ρ is not considered. In this case, ρ is known as a *passive* scalar field.

Nevertheless, when dealing with thermal flows, the fluid dynamics is often affected by the temperature field. The most familiar examples are evaporation, boiling and melting, and more generally, when the viscosity or fluid density changes with temperature. In this case, this phenomenon is treated as a forcing term, as in the Rayleigh-Bénard convection.

The Rayleigh-Bénard convection is a well studied problem [44] which is often used as a benchmark test for numerical thermal problems. This phenomenon consists in a coupling

between momentum and advection-diffusion in a thermal flow such that the density depends on temperature. Let us consider two parallel plates separated by a distance L . The lower plate is kept at a temperature T_l and the upper one at T_u such that: $T_l > T_u$. Both plates are also subject to a no-slip boundary condition for the fluid momentum. Between these plates there is a fluid with a thermal expansion coefficient α at constant pressure p . This can be simulated with two coupled lattice-Boltzmann models (Chapter 3) and is illustrated in Figure 2-2.

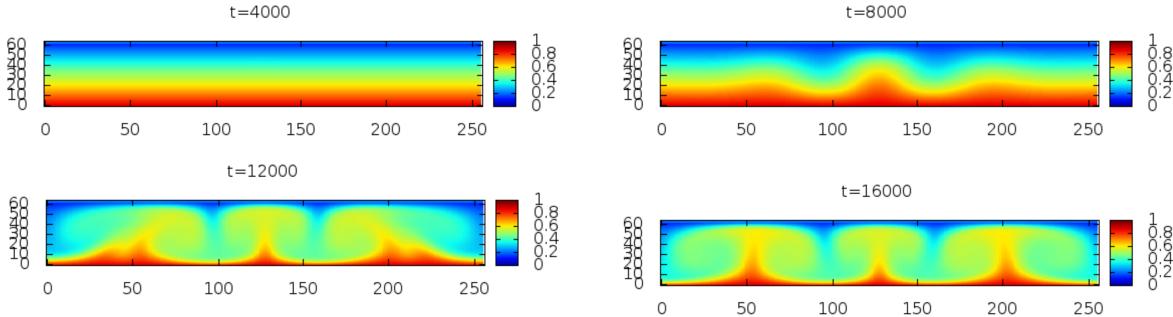


Figure 2-2: Temperature plot of Rayleigh-Bénard convection. Snapshots at $t = 4000$ s, $t = 8000$ s, $t = 12000$ s and $t = 16000$ s. This simulation was performed with two coupled LBM: a D2Q9 for the fluid flow and a D2Q5 for the temperature field. The relaxation times are $\tau_{D2Q9} = 0.58$ and $\tau_{D2Q5} = 0.68$. The 2D computational domain is 256×64 cells, with $\Delta x = 1$ cm and $\Delta t = 1$ s. Dirichlet boundary conditions were imposed at upper and lower plates with dimensionless temperature values: $T_u = 1$ and $T_l = 0$.

A very common approximation for the density is to assume that it changes linearly with temperature as

$$\rho(T) \approx \rho_0[1 - \alpha(T - T_0)] \quad . \quad (2-9)$$

Once the relation between the fluid density and the temperature is set, we need to couple the ADE with the NSE. To do so, another approximation is considered. In consequence, small changes in the density leads to a buoyancy force density

$$\vec{F}_b = [\rho(T) - \rho_0] \vec{g} = -\alpha \rho_0 (T - T_0) \vec{g} \quad (2-10)$$

when a gravitational field with acceleration \mathbf{g} is present. This is known as the Boussinesq approximation.

With this expression in mind, let us consider again the Rayleigh-Bénard convection. Due to the temperature increase in the lower plate, the fluid density will decrease there and, therefore, a buoyancy force will emerge, leading to convection cells that are characteristics

of this phenomenon, as can be seen in Figure 2-2 . A small perturbation (in momentum or in temperature) is needed, to start the process.

The phenomenon is governed by the *Rayleigh number*,

$$\text{Ra} = \frac{g\alpha(T_l - T_u)L^3}{\kappa\nu} , \quad (2-11)$$

where κ and ν are the thermal diffusivity and kinematic viscosity of the fluid, respectively. For high Rayleigh numbers, buoyancy can overcome dissipation and lead to convection.

The main advantage of using the Boussinesq approximation is that the effect of the temperature-dependent density is treated as a external body force, like any other in the Navier Stokes equations. In this case, the fluid density that appears in the NSE remains unchanged in order to avoid dealing with mass conservation problems.

CHAPTER 3

Lattice Boltzmann models for the advection-diffusion equation

In this Chapter, we describe the basic notions of the Lattice Boltzmann Method (LBM), starting from the Boltzmann equation for transport and its discretization describing then how it can be implemented to solve the advection-diffusion equation (ADE) in Cartesian coordinates (Section 3.2). In Sec. 3.1.2, we describe some of the most important velocity schemes. Next, in Sec. 3.1.3 we present different approaches for boundary conditions. Later on, we present some examples of the ADE in Cartesian coordinates. In Section 3.3 we introduce a BGK scheme that allows to simulate the ADE with a source term, which will be useful in the next chapter to construct our model in generalized coordinates. Next, in Section 3.4 we explain the advantages of using generalized coordinates, previous works for the ADE in generalized coordinates Finally, we describe a Multiple-Relaxation-Time (MRT) LBM proposed by Yoshida and Nagaoka in 2014 [42], to solve the advection-diffusion equation in generalized coordinates.

3.1 What is a Lattice Boltzmann Model?

A lattice Boltzmann model (LBM) is a Cellular Automata (CA) that implements a discrete form of the Boltzmann equation. The space is divided into cells, the time, into discrete steps, and there is at each cell a set of discrete velocities \vec{c}_λ transferring the information to their neighbouring cells. At each cell and associated with each velocity there is a so-called *distribution function* representing the probability to find a particle with that direction and within that cell. The evolution is divided into two stages: collision and streaming.

3.1.1 Boltzmann transport equation

Let us define a distribution function f representing the number of molecules between r and $r + d$, with velocities between c y $c + dc$ at a time t . If an external force F is applied on these molecules, its velocity will change to $c + Fdt$ and its position to $r + cdt$. If we consider no collisions within the process, the number of molecules in an initial time will remain unchanged until the end of the process. On the contrary, if there are collisions the number of molecules will change. This changing rate between the initial and the final state is represented by a *collision operator* Ω , such that

$$f(r + cdt, c + Fdt, t + dt)drdc - f(r, c, t)drdc = \Omega(f)drdcdt . \quad (3-1)$$

By taking $dt \rightarrow 0$ and dividing by $drdcdt$, we obtain the *Boltzmann Transport Equation* [45],

$$\frac{df}{dt} = \Omega(f) . \quad (3-2)$$

The above equation states that the total time derivative of the distribution is equal to the collision rate. Taking into account that f is a function of r, c and t and that there is no external forcing, the Boltzmann equation can be written as

$$\frac{\partial f}{\partial t} + \mathbf{c} \cdot \nabla f = \Omega . \quad (3-3)$$

In the BGK approximation [46], once the equation 3-3 has been discretized and the collision operator Ω is replaced, the Boltzmann equation reads

$$f_\lambda(\vec{x} + \vec{c}_\lambda \Delta t, t + \Delta t) - f_\lambda(\vec{x}, t) = -\frac{1}{\tau}[f_\lambda(\vec{x}, t) - f_\lambda^{eq}(\vec{x}, t)] , \quad (3-4)$$

where $f_\lambda(\vec{x}, t)$ and f_λ^{eq} are the distribution function and the *equilibrium* distribution function, respectively, at cell \vec{x} and time t , and τ is a relaxation time. The right-hand term is known as the collision term.

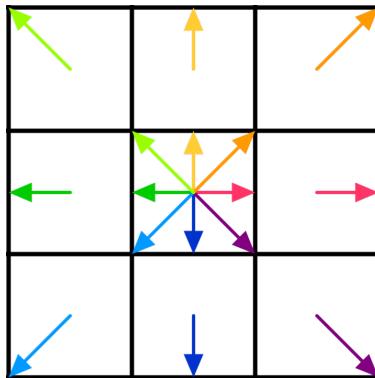


Figure 3-1: Streaming step in a lattice-Boltzmann method illustrated using a D2Q9 velocity scheme.

The equilibrium distribution function f_λ^{eq} is chosen to reproduce the desired conservation laws; for instance,

$$f_\lambda^{eq} = \omega_\lambda \rho \left(1 + \frac{\vec{c}_\lambda \cdot \vec{u}}{c_s^2} + \frac{(\vec{c}_\lambda \cdot \vec{u})^2}{2c_s^4} - \frac{|\vec{u}|^2}{2c_s^2} \right) \quad (3-5)$$

reproduces in the continuous limit of infinitely small cells the Navier-Stokes equation for an isothermal ideal gas in the limit of low-density fluctuations and low Mach numbers [47, 48]. The density ρ and the momentum $\rho\vec{U}$ are computed as the zeroth and first-order moments of f_λ ,

$$\rho = \sum_\lambda f_\lambda \quad , \quad \rho\vec{u} = \sum_\lambda \vec{c}_\lambda f_\lambda \quad . \quad (3-6)$$

3.1.2 Some LBM velocity schemes

The weights ω_λ , the discrete velocities \vec{c}_λ and the speed c_s in Eq. (3-5) are chosen through a Gauss-Hermite quadrature such that the discrete sums in Eq. 3-6 coincide with the zero-th and first order momenta of the continuous Boltzmann's distribution function $f(\vec{x}, \vec{v}, t)$ [1]. The most common schemes are D2Q9 in 2D and D3Q19 in 3D, where D stands for dimensions and Q for number of vectors. In a D3Q19 scheme (the velocities are shown in Fig. (3-2b), $c_s^2 = c^2/3$ and $c = \Delta x/\Delta t$, where Δx is the lattice spacing and the weights are presented in Table 3-1

Scheme	c_s	Velocities \vec{c}_λ	number	Weight ω_λ
D1Q3	$1/\sqrt{3}$	0	1	4/6
		± 1	2	1/6
D2Q5	$1/2$	(0, 0)	1	1/3
		$(\pm 1, 0), (0, \pm 1)$	4	1/6
D2Q9	$1/\sqrt{3}$	(0, 0)	1	4/9
		$(\pm 1, 0), (0, \pm 1)$	4	1/9
		$(\pm 1, \pm 1)$	4	1/36
D3Q7	$1/2$	(0,0,0)	1	1/4
		$(\pm 1,0,0), (0,\pm 1,0), (0,0,\pm 1)$	6	1/8
D3Q19	$1/\sqrt{3}$	(0,0,0)	1	1/3
		$(\pm 1,0,0), (0,\pm 1,0), (0,0,\pm 1)$	6	1/18
		$(\pm 1,\pm 1,0), (\pm 1,0,\pm 1), (0,\pm 1,\pm 1)$	12	1/36

Table 3-1: LBM Schemes in two and three dimensions

The more moments of the distribution involved in the conservation laws, the more velocities needed. This can lead to the so-called *extended lattices* [1]. In the case of the ADE, using a lattice configuration D3Q7 is enough to reproduce the necessary moments of the distribution

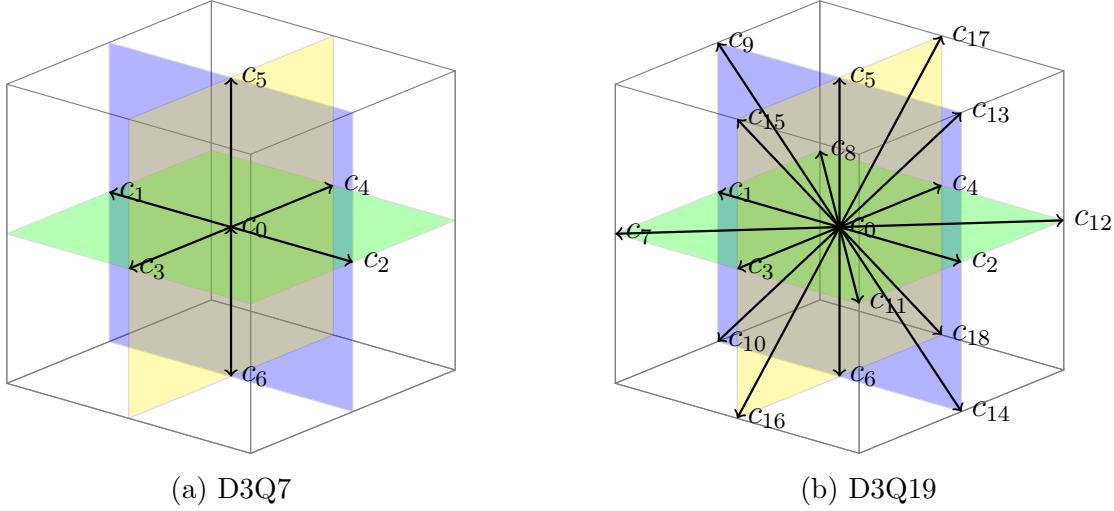


Figure 3-2: LBM velocity schemes in three dimensions.

and can be as accurate as other extended lattices when convection is not very strong and the boundary effects are not significant. Additionally, this LBM scheme has less vectors than the D3Q19, and, therefore, it reduces the computational cost of the model. [49]. In a D3Q7 scheme, (Fig. 3-2a) $c_s^2 = c^2/4$ and $c = \Delta x/\Delta t$.

3.1.3 Boundary conditions

The solutions of a partial differential equation (PDE) are determined by the boundary condition imposed on some surfaces in the domain. In a second-order PDE there are two main boundary conditions, known as Dirichlet, or von Neumann boundary conditions.

Dirichlet boundary conditions consist in imposing the value of a determined field in a wall, in this case, a concentration ρ_w or a temperature T_w . This kind of boundary conditions is implemented when a wall is kept at a constant temperature or when the value of the concentration is known at the border. *Neumann boundary conditions* consist in imposing a flux (that is, a fixed value for the spatial derivative of the field) on the boundary. Usually, it is zero, so a good approach is to copy the information of the post-collision distribution functions to the boundaries to guarantee a zero flux. Another option is to convert the Neumann boundary conditions into a Dirichlet one by solving for the unknown concentration.

Boundary conditions can be imposed on LBMs in many ways. For the case of Dirichlet boundary conditions, two strategies are preferred: either to fix the desired value of the field to compute the equilibrium distribution functions (instead of computing them from the distribution functions) or to fix the distribution functions to take the corresponding equilibrium distribution functions. It turns out to be more accurate to impose the boundary conditions on the distribution functions f_λ rather than in the macroscopic quantities (ρ, \vec{u}, T) ,

since in the boundary cells there are some f_λ known that can be used to compute the required value, depending on the used scheme.

3.2 Lattice Boltzmann models for the advection-diffusion equation in Cartesian coordinates

Equations (3-4) and (3-5) reproduce the advection-diffusion equation,

$$\partial_t \rho + \nabla \cdot (\rho \vec{u}) = D \nabla^2 \rho , \quad (3-7)$$

if the velocity field \vec{u} is externally given, instead of being computed from the functions f_λ . As a consequence, this model does not conserve momentum [1]. The constant D is the diffusion coefficient, and is related to the relaxation time τ by

$$D = c_s^2 \left(\tau - \frac{1}{2} \right) \Delta t . \quad (3-8)$$

An alternative also found in literature [24, 32] chooses the equilibrium function to be linear in velocities,

$$f_\lambda^{eq} = \omega_\lambda \rho \left(1 + \frac{\vec{c}_\lambda \cdot \vec{u}}{c_s^2} \right) , \quad (3-9)$$

which reproduces the advection-diffusion processes with some corrections [32].

3.2.1 Diffusion and advection of an one-dimensional density step

Let us consider a one-dimensional diffusion-advection problem. We define a channel 100m long with an uniform constant velocity of $u_x = 0.02\text{m/s}$ and an initial concentration distribution given by

$$\rho_0(x, t=0) = \begin{cases} 1 & \text{if } x < x_0 \\ 0 & \text{if } x \geq x_0 \end{cases} , \quad (3-10)$$

with $x_0 = 50\text{m}$, whose analytical solution is given as

$$\rho(x, t) = 0.5 \left(1 - \operatorname{erf} \left(\frac{x - x_0 - u_x t}{\sqrt{4Dt}} \right) \right) . \quad (3-11)$$

We simulate this phenomenon with a D3Q7 LBM scheme and a computational array of $200 \times 2 \times 2$ cells ($\Delta x = 0.5\text{m}$). We implement periodic boundary conditions in $y-$ and $z-$ directions and open boundary conditions in $x-$ direction. The diffusion coefficient is $D = 0.05\text{m}^2/\text{s}$ i.e. $\tau = 0.7$. We observe an excellent agreement between the theoretical expression and the numerical results, as shown in Fig. 3-3

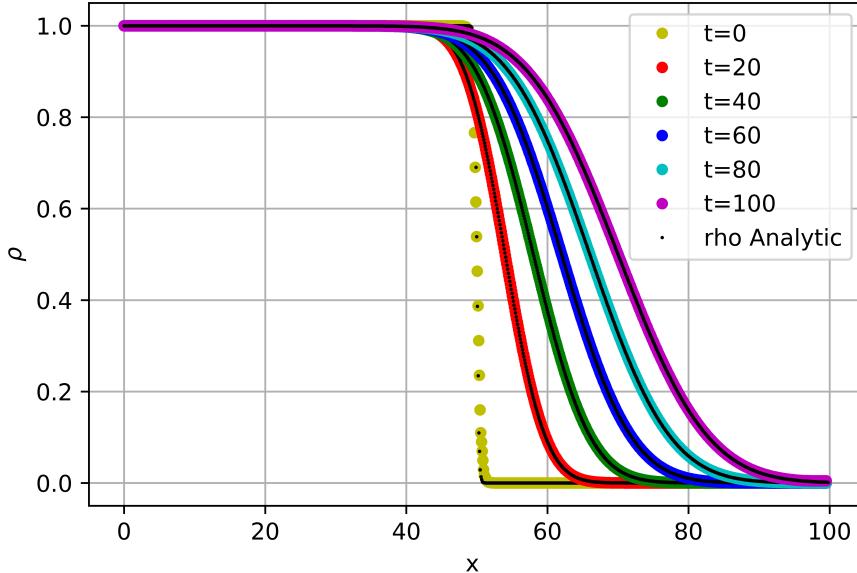


Figure 3-3: Advection and diffusion of a one-dimensional density step in a channel 100 meters long simulated with a D3Q7 scheme after 1000 seconds. The diffusion coefficient is $D = 0.05$ and the advection velocity $u_x = 0.02$. An excellent agreement with the theoretical solution is found.

3.2.2 Diffusion and advection of a Gaussian hill

Let us consider a two-dimensional problem, where an initial Gaussian distribution is located at (x_0, y_0) and the surrounding medium shows an uniform and constant velocity \vec{u} . The initial distribution is given by

$$\rho(x, y, t) = \frac{A}{1 + 2Dt/\sigma_0^2} \exp\left(-\frac{(x - \bar{x}_0)^2 + (y - \bar{y}_0)^2}{2(\sigma_0^2 + 2Dt)}\right) , \quad (3-12)$$

where $\bar{x}_0 = x_0 + u_x t$, $\bar{y}_0 = y_0 + u_y t$, A is the maximum density and D is the diffusion coefficient. For our simulation we choose a computational domain of $L_x \times L_y \times L_z = 100 \times 100 \times 1$ cells, $\sigma_0 = 6.4\text{m}$, $D = 0.05 \text{ m}^2/\text{s}$, $u_x = u_y = 0.03\text{m/s}$ and $(x_0, y_0) = (33\text{m}, 33\text{m})$. We simulate this phenomenon during 1000s, as shown in Fig. 3-4. Fig. 3-5 (a)) plots the one-dimensional profile along the diagonal from $(0, 0)$ to (L_x, L_y) every 200s. Fig. 3-5 (b)) shows the variance (σ^2) against time exhibiting the expected linear behavior, $\sigma^2 = \sigma_0^2 + 2Dt$, where the slope is related to the diffusion coefficient. In this case, from the linear regression we find that the diffusion coefficient is $D^* = 0.04995 \text{ m}^2/\text{s}$, which matches the simulation parameter.

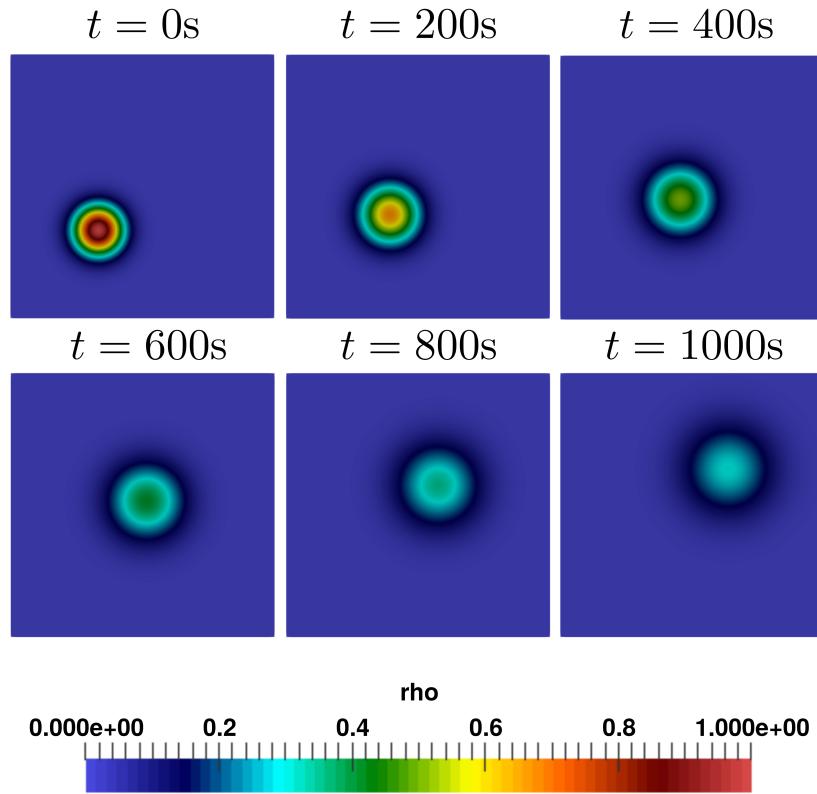


Figure 3-4: Advection and diffusion of an initial Gaussian density on a fluid with a uniform velocity field with $\sigma_0 = 6.4\text{m}$, $D = 0.05 \text{ m}^2/\text{s}$, $u_x = u_y = 0.03\text{m/s}$ and $(x_0, y_0) = (33\text{m}, 33\text{m})$. Snapshots every 200s.

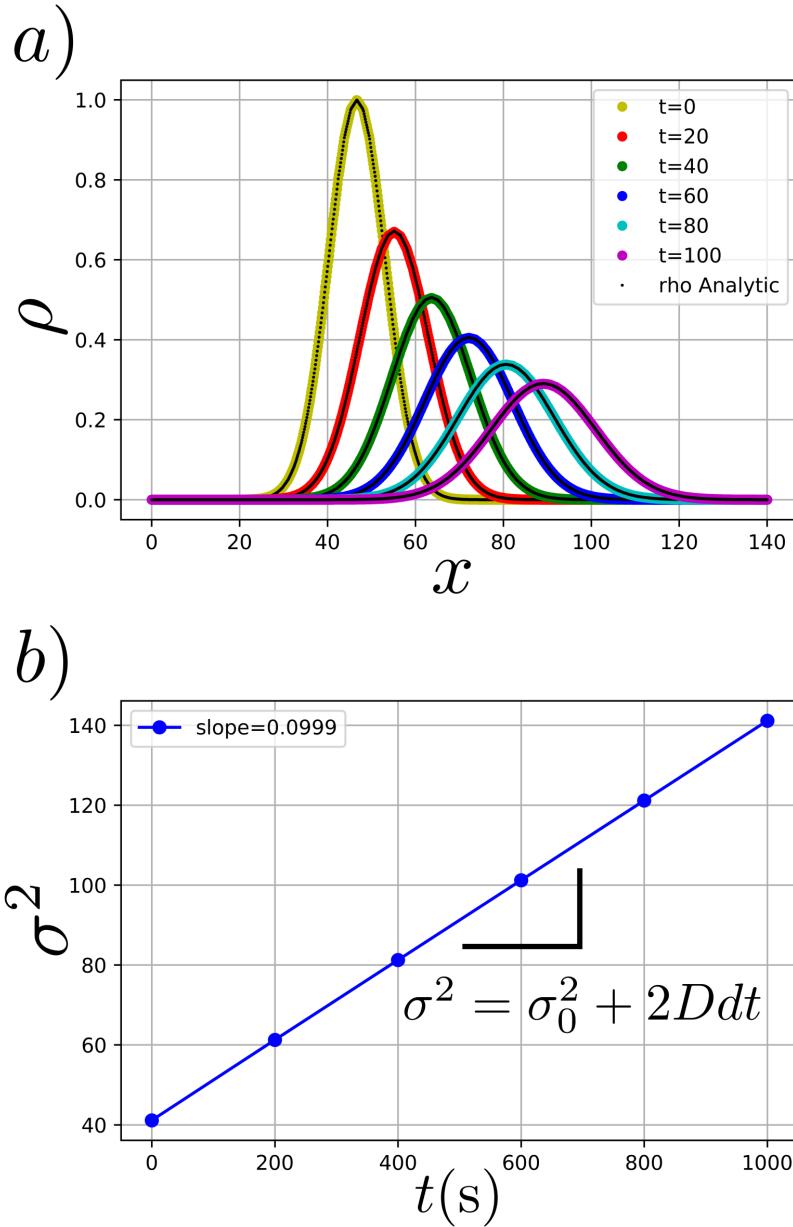


Figure 3-5: Advection and diffusion of an initial Gaussian density on a fluid with an uniform velocity field with $\sigma_0 = 6.4\text{m}$, $D = 0.05 \text{ m}^2/\text{s}$, $u_x = u_y = 0.03\text{m/s}$ and $(x_0, y_0) = (33\text{m}, 33\text{m})$. a) One-dimensional profile along the diagonal from $(0, 0)$ to (L_x, L_y) every 200 s. b) Variance σ^2 against time t . A linear regression gives $D = 0.04995 \text{ m}^2/\text{s}$, which matches the simulation parameter $D = 0.05 \text{ m}^2/\text{s}$.

3.3 Adding a source term to the advection-diffusion equation with an LBM

In many circumstances, it is useful to reproduce the ADE with an external source, as a waste drainage in a river, a chimney in the air or a heat source, among others. There are several proposals to introduce a source term to work with hydrodynamic problems [50], [51], [52]. Regarding the ADE, the most studied problem is related with the implementation of a heat source [53]. Especially for this case, Shi et al. [30] developed a scheme where the source term is accurate to second order. This method consists of adding two terms to the evolution step (Eq. (3-4)),

$$f_\lambda(\vec{x} + \vec{c}_\lambda \Delta t, t + \Delta t) - f_\lambda(\vec{x}, t) = -\frac{1}{\tau}[f_\lambda(\vec{x}, t) - f_\lambda^{eq}(\vec{x}, t)] + \underbrace{\Delta t S_\lambda + \frac{\Delta t^2}{2} \bar{D}_\lambda S_\lambda}_{\text{new terms}} , \quad (3-13)$$

with

$$S_\lambda = \omega_\lambda S \left(1 + \frac{\tau - \frac{1}{2} \vec{c}_\lambda \cdot \vec{u}}{\tau - \frac{\theta}{2} c_s^2} \right) , \quad (3-14)$$

and S the desired source term, with the operator $\bar{D}_\lambda = \partial_t + \theta \vec{c}_\lambda \cdot \nabla$. Here, $\theta \in [0, 1]$ is a parameter for different schemes whose value in this work is $\theta = 1$ and, therefore, the differential \bar{D}_λ is computed with a forward scheme

$$D_\lambda S_\lambda = \frac{S_\lambda(\vec{x}, t) - S_\lambda(\vec{x} - \vec{c}_\lambda \Delta t, t - \Delta t)}{\Delta t} \quad (3-15)$$

Eq. (3-14) fulfills

$$\sum_\lambda S_\lambda = S , \quad \sum_\lambda \vec{c}_\lambda S_\lambda = \frac{\tau - \frac{1}{2}}{\tau - \frac{\theta}{2}} S \vec{u} . \quad (3-16)$$

Through a Chapman-Enskog expansion [30] (See Appendix A for a detailed derivation), it is possible to show that the ADE with a source term,

$$\partial_t \rho + \nabla \cdot (\rho \vec{u}) = D \nabla^2 \rho + \vec{S}(\vec{x}, t) \quad (2-1)$$

is reproduced in the continuous limit. The equilibrium function is still defined as in Eq. (3-5), and the density as in Eq. (3-6). This scheme will be used in the next section to introduce, as a source term, the information about the curved geometry.

3.4 Lattice Boltzmann model for the advection-diffusion equation in generalized coordinates

3.4.1 Why generalized coordinates?

One of the main complications that researchers deal with when working with numerical models to solve the advection-diffusion equation is the necessity of an accurate implementation of the boundary conditions. It is easier when the boundaries are rectangular and parallel to the computational mesh. However, most of the times, real-life problems have complex geometries and, therefore, boundaries are not easy to implement. One of the approaches to deal with these boundary conditions is known as staircase approximations where the curved boundary is built from squares in 2D or cubes in 3D, and the accuracy depends on the resolution. Another solution to implement these boundary conditions is to mesh the domain with irregular polygons or polyhedra, for example, triangles or tetrahedra , but those meshes result better to implement only for some computational methods as the Finite Element Methods. A third alternative, known as *immersed boundary methods*, employs interpolation steps to adjust the shape of the boundary, but those extra steps slow down the computation. In the specific case of lattice-Boltzmann methods, where the domain needs to be discretized with uniform cells to maintain the isotropy and consistency in the evolution step, there exist many treatments on the boundary conditions [54], [55], immersed boudndaries (IB) [56], among others.

Fig. 3-6 shows three ways to treat a spherical geometry: in the left side, a very common staircase approximation, where the resolution and cell size play the main role; in the center, a mesh obtained from a spherical coordinate transformation fits perfectly the boundary but carries on a singularity at $r = 0$, and on the right side, our computational domain where the points are uniformly spaced and each direction represents a coordinate.

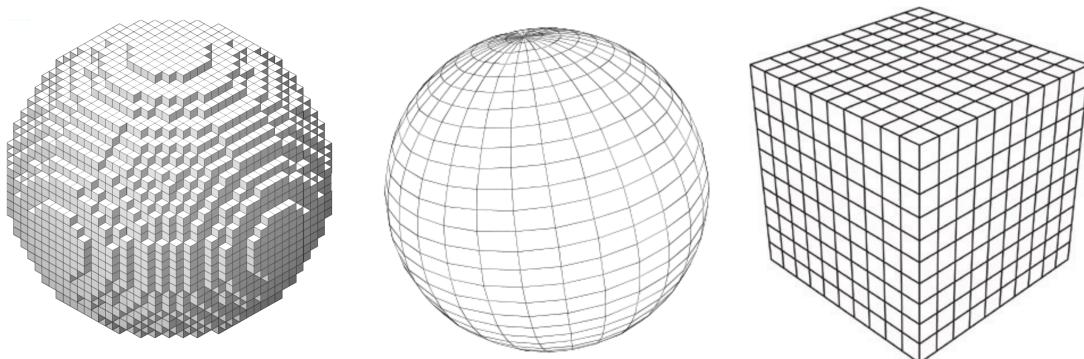


Figure 3-6: Different domains for consider a sphere. Staircase approximation mesh (left). Spherical coordinates mesh (center). Computational domain (right).

In brief, our motivation to implement generalized coordinates is:

- To avoid staircase approximation when defining the geometries.
- To simplify boundary condition implementation in curved and/or complex geometries.
- To take advantage of the geometry's symmetries.
- To make a mesh refinement in some specific regions of the physical domain without affecting the evolution in the computational domain.
- To deal with more complex and even time-changing geometries.

3.4.2 Numerical models for the ADE in generalized coordinates

There have been many different models to solve numerically the advection-diffusion equation in generalized coordinates and/or complex geometries, some of them based on Semi-Lagrangian algorithms [57] and Finite-Volume Methods [54], [58].

In 1997, Karpik and Crockett [57] presented a semi-Lagrangian method for the solution of the unsteady advection-diffusion equation in complex geometries. They define an orthogonal, boundary-fitted coordinate system which allows them to discretize the computational domain into a set of control volumes. Both processes, the advective and diffusive, are decoupled through a splitting method and solved separately. The value of the transported field is computed via an interpolation scheme after the calculation of back trajectories along characteristic lines. The advection velocity field, which is assumed stationary, is obtained from a Finite-Differences or Finite-Volume code for fluid dynamics. Their method is simple to implement and produces high accuracy results. Their model is implemented in one- and two-dimensional problems and in an annular domain with an angular velocity.

In 2000, using Finite-Volume Methods (FVM), Calhoun and LeVeque [54] developed a model on a Cartesian grid to solve the ADE in irregular geometries. This algorithm employs a Cartesian grid where some of the cells are cut by the presence of a boundary. To improve the accuracy of the model they implement a 'capacity function' to know which percentage of the cell is inside the boundary, so it would be only partially available to the fluid. The model is tested in geometries like porous media, inclined channels, and an annulus.

In 2010 Ashbourn et al. [58] proposed a numerical model to simulate two- and three-dimensional axisymmetric Advection-Diffusion systems with complex geometries using finite-volume methods. Their model deals accurately with complicated, curved boundaries. One of the advantages of using finite-volume methods is the possibility of creating unstructured meshes, which are suitable to boundaries in complex geometries, but also usually difficult

to implement and computationally expensive. They modify the two-dimensional cut-cell technique of [54] to adjust the mesh to the desired boundary through a method of lines (MOL) but, unlike Calhoun and LeVeque, they treat the discretization of diffusion and advection parts simultaneously, obtaining a single system of differential equations to solve. This system is, therefore, solved with advanced time-stepping schemes of high order, ensuring efficiency and accuracy. Their model is tested on three examples: a fluid carrying a dye and rotating in an annulus, the bone fracture healing in mice, and the use of vessels of different geometry in an ultracentrifuge.

3.4.3 Lattice-Boltzmann models in generalized coordinates

In 1997, He and Doolen [34] proposed an LBM model for the 2D simulation of a vortex shedding behind a circular obstacle by defining the LBM on polar coordinates. To achieve this goal, they used the so-called interpolation-supplemented lattice-Boltzmann equation (ISLBE) method. This method allows to discretize the flow domain into an arbitrary grid mesh and consists of three stages: collision, streaming and interpolation. This interpolation step allows to evaluate the distribution function on the non-uniformly distributed lattice points using the values of the distribution function that propagate regularly on a square grid. Nevertheless, the interpolation algorithm should be at least second order accurate to avoid destructive numerical diffusivity and viscosity, and the additional interpolation steps slow down the simulation.

Later on, in 2012, Mendoza [12] proposed in his PhD thesis an LBM for hydrodynamics that is able to simulate fluids virtually on any kind of geometry that can be described by a curved coordinate system. Written in generalized coordinates, the hydrodynamic equations present additional terms that depend on the geometric transformation defining the coordinates. Indeed, those extra terms are usually expressed by means of the metric tensor and Christoffel symbols for the coordinates. The strategy consists in keeping a cubic lattice in the computer but adding the forcing terms required to reproduce those extra geometric terms. In contrast with previous proposals [33, 34, 35, 36], there is no need for interpolation steps. This LBM was successfully used to study the flow through randomly curved manifolds [37], the Dean's instability in double curved channels on ellipsoidal coordinates [38] and, more recently, the energy dissipation in flows through curved spaces [39] and the vibrational modes of a trumpet [40].

Simultaneously in 2012, Budinsky [41] developed a method for modeling 2D flows on curvilinear coordinates to simulate flows in complex geometries. Their physical domain of interest is enclosed by a curvilinear mesh that fits all the boundaries, and by computing the metrics of the transformation, the computational domain is settled and the correspondence between the physical and the lattice nodes is established. The additional geometric terms related

with the Jacobian and its spatial derivatives are introduced as forcing terms. This model simplifies the implementation avoiding interpolation steps or additional discretizations since all the geometric information is taken into account in the equilibrium function and the forcing term. They test their model in three cases: a straight prismatic channel angled 45° to the main axes, a pool of circular shape and a prismatic flume in a 180° bend.

Regarding advection-diffusion processes, in 2000, Van der Sman and Ernst [28] presented a model for ADE on irregular but rectangular (orthorrombic and de Bravais) lattices where there is no need of interpolation or coarse-graining steps. Their strategy bases on the fact that the velocity moments of the equilibrium particle distribution function must equal those of the classical Maxwell-Boltzmann distribution.

In 2014, Yoshida and Nagaoka [42] proposed a Multiple-Relaxation-Time Lattice Boltzmann Model (MRT-LBM) for the ADE in curvilinear coordinates, where the relaxation times depend on the geometry. Their model does not need any interpolation steps and its implementation is basically the same as in a regular lattice system. This model shows a solution for the ADE that is second-order accurate with respect to the grid interval according to an asymptotical analysis. As benchmark tests, they treat three punctual problems: a one-dimensional diffusion problem using two different non-uniform grids, a two-dimensional problem to simulate the Taylor-Aris problem using a rectangular grid and a two-dimensional axisymmetric problem on a grid represented by an orthogonal curvilinear coordinate transformation. This model is explained in detail in Section 3.4.4

3.4.4 A Multiple Relaxation Time (MRT) LBM for the ADE in curvilinear systems

Hereby we summarize the model for the advection-diffusion equation in curvilinear systems proposed by Yoshida and Nagaoka in 2014 [42]. They implement a Multiple Relaxation Time (MRT) model. This model begins with the advection-diffusion equation in Cartesian coordinates

$$\frac{\partial \rho}{\partial t} + \frac{\partial}{\partial x_i}(\rho u_i) = \frac{\partial}{\partial x_i} \left(D \frac{\partial \rho}{\partial x_i} \right) , \quad (3-17)$$

where $\rho(0, \vec{x}) = \rho_0(\vec{x})$. Here, the spatial coordinates in a Cartesian system are denoted by \vec{x} , the diffusion coefficient by $D(\vec{x}, t)$ and the advection velocity by $\vec{u}(\vec{x}, t)$.

To introduce the problem in curvilinear coordinates, they define a transformation given by $x_i = x_i(\vec{\eta})$, where $\vec{\eta}$ are the curvilinear coordinates. The computational grid points are equally spaced in a cuboid domain, while in the physical domain those points are located

non-uniformly. The ADE expressed in a curvilinear coordinate system reads as follows:

$$\frac{\partial \rho}{\partial t} + \frac{1}{J} \frac{\partial}{\partial \eta_i} (J \rho U_i) = \frac{1}{J} \frac{\partial}{\partial \eta_i} \left(D J g^{ij} \frac{\partial \rho}{\partial \eta_j} \right) , \quad (3-18)$$

where J is the Jacobian of transformation, U_i is the contravariant component of the advection velocity \vec{u} and g^{ij} is the contravariant metric tensor, defined as

$$J = \det \left(\frac{\partial x_i}{\partial \eta_j} \right) , \quad U_j = u_k \frac{\partial \eta_j}{\partial x_k} , \quad g^{ij} = \frac{\partial \eta_i}{\partial x_k} \frac{\partial \eta_j}{\partial x_k} . \quad (3-19)$$

In this LBM model, the distribution function and the conserved quantity P are defined such that the summation of f_λ over all directions approximates the product of the Jacobian J and the physical density ρ ,

$$P = \rho J = \sum_{\lambda} f_{\lambda} . \quad (3-20)$$

For the simulations they use D1Q3 and D2Q5 schemes (see Table 3-1).

The lattice-Boltzmann equation is given by

$$f_{\lambda}(\vec{\eta} + \vec{c}_{\lambda} \Delta \eta, t + \Delta t) - f_{\lambda}(\vec{\eta}, t) = \Omega(f_{\lambda}(\vec{\eta}, t)) , \quad (3-21)$$

where Ω is the collision operator defined as

$$\Omega(f_{\lambda}(\vec{\eta}, t)) = \sum_{\beta} (M^{-1} S M)_{\lambda \beta} (f_{\beta}^{eq}(\rho(\vec{\eta}, t)) - f_{\beta}(\vec{\eta}, t)) . \quad (3-22)$$

Here, f_{λ}^{eq} is the equilibrium distribution function,

$$f_{\lambda}^{eq}(\rho) = P \omega_{\lambda} \left(1 + \frac{\Delta t}{\Delta \eta c_s^2} V_i c_{i\lambda} \right) , \quad V_i = U_i + D g^{ij} \frac{\partial \ln J}{\partial \eta_j} . \quad (3-23)$$

The weight coefficients ω_{λ} are defined as

$$\omega_{\lambda} = \begin{cases} \kappa & \text{if } \lambda = 0 \\ \frac{1-\kappa}{m-1} & \text{if } \lambda = 1, \dots, m \end{cases} \quad (\text{for a DdQm model}) ,$$

where $\kappa \in (0, 1)$. Here, c_s^2 is the coefficient of the second-order tensor defined as

$$\sum_{\lambda} \omega_{\lambda} c_{i\lambda} c_{j\lambda} = c_s^2 \delta_{ij} , \quad (3-24)$$

where δ_{ij} is the Kronecker's delta. Specifically, $c_s^2 = 1 - \kappa$ in the D1Q3 model, and $c_s^2 = (1 - \kappa)/2$ in the D2Q5 model.

In comparison with the BGK evolution step (3-4), where there is only one relaxation time in the collision term, in this MRT model there is a full matrix of parameters that determine the rate at which each momentum of the distribution functions f_λ will relax. The size of these matrices M and S is $m \times m$, i.e., the more velocities are used, the larger the matrix will be.

In a MRT-LBM model (Equation 3-22), the distribution function f_λ is projected onto momentum space by a matrix M , in which each component is a moment of f_λ . Those moments relax to the equilibrium distribution with the relaxation-time coefficients in S . The explicit expression of M for the two velocity schemes employed are

$$M_{D1Q3} = \begin{pmatrix} 1 & 1 & 1 \\ 0 & 1 & -1 \\ 2 & -1 & -1 \end{pmatrix}, \quad M_{D2Q5} = \begin{pmatrix} 1 & 1 & 1 & 1 & 1 \\ 0 & 1 & -1 & 0 & 0 \\ 0 & 0 & 0 & 1 & -1 \\ 4 & -1 & -1 & -1 & -1 \\ 0 & 1 & 1 & -1 & -1 \end{pmatrix}. \quad (3-25)$$

The first component of the projection coincides with the value of P . The flux of P is related to the second component in the case of the D1Q3 model or the second and third components in the case of the D2Q5 model ($\sum_\lambda \vec{c}_\lambda f_\lambda$). The other components correspond to the second-order moments with respect to \vec{c}_λ , which are defined in such a manner that the row vectors of M are all orthogonal. The matrix S consists of the relaxation-time coefficients that are assigned to each component in momentum space:

$$S_{D1Q3}^{-1} = \begin{pmatrix} \tau_0 & 0 & 0 \\ 0 & \bar{\tau}_1 & 0 \\ 0 & 0 & \tau_2 \end{pmatrix}, \quad S_{D2Q5}^{-1} = \begin{pmatrix} \tau_0 & 0 & 0 & 0 & 0 \\ 0 & \bar{\tau}_{11} & \bar{\tau}_{21} & 0 & 0 \\ 0 & \bar{\tau}_{12} & \bar{\tau}_{22} & 0 & 0 \\ 0 & 0 & 0 & \tau_3 & 0 \\ 0 & 0 & 0 & 0 & \tau_4 \end{pmatrix}. \quad (3-26)$$

The relaxation-time coefficient $\bar{\tau}_{ij}$ is directly related to the diffusion coefficient D through the following expression:

$$\bar{\tau}_{ij} = \frac{1}{2} \delta_{ij} + \frac{D \Delta t}{c_s^2 \Delta \eta^2} g^{ij} . \quad (3-27)$$

In this model, the coefficient τ_0 for the conserved quantity P does not affect the numerical solution, while the other ones τ_2 , τ_3 and τ_4 affect the numerical solution of the ADE only through error terms.

Their algorithm for the implementation of the LBM goes as follows:

1. Initial condition:

$$f_\lambda(\vec{\eta}, t) = \left(1 + \frac{\Delta t}{\Delta \eta c_s^2} V_i c_{i\lambda} \right) P_0 \omega_\lambda - \sum_\beta \Delta \eta \frac{\partial P_0}{\partial \eta_i} (M^{-1} S^{-1} M)_{\lambda\beta} c_{i\beta} \omega_\beta . \quad (3-28)$$

2. Collision process:

$$f_\lambda^*(\vec{\eta}, t) = f_\lambda(\vec{\eta}, t) + \Omega(f_\lambda(\vec{\eta}, t)). \quad (3-29)$$

3. Streaming process:

$$f_\lambda(\vec{\eta} + \vec{c}_\lambda \Delta \eta, t + \Delta t) = f_\lambda^*(\vec{\eta}, t). \quad (3-30)$$

4. Boundary rule for the Dirichlet condition:

$$f_\lambda(\vec{\eta}, t + \Delta t) = -f_\beta^*(\vec{\eta}, t) + c_s^2 J \Psi_d, \quad (3-31)$$

where β indicates the opposite direction to λ , and Ψ_d is the boundary value as function of \vec{x} and t .

5. Boundary rule for Neumann condition:

$$f_\lambda(\vec{\eta}, t + \Delta t) = f_\beta^*(\vec{\eta}, t) + \frac{\Delta t \Psi_n}{\Delta \eta |\vec{n}|}, \quad (3-32)$$

where \vec{n} is the normal vector, $|\vec{n}| = \sqrt{n_j^2}$ and Ψ_n is the boundary value as function of \vec{x} and t .

6. Computation of $P(\vec{\eta}, t + \Delta t)$, using $f_\lambda(\vec{\eta}, t + \Delta t)$.

7. Repeat steps (2.) through (6.).

By defining a curvilinear system that fits the boundary guarantees that the grid is always parallel to the boundary, so the Neumann condition should be accurate enough.

With this model, Yoshida and Nagaoka perform three numerical examples: a one-dimensional problem with non-uniform grids to simulate vapor diffusion, where a coordinate transformation is defined such that more detailed information is recovered in the region of interest. As a two-dimensional problem, they simulate the Taylor-Aris dispersion, where the advection and diffusion of the concentration between two parallel plates with an external flux are analyzed. These two examples are benefited from the ability to perform a mesh refinement. As the last example, they study an axisymmetric two-dimensional problem with an orthogonal curvilinear coordinate grid, namely the diffusion-limited current at an oblate hemispherical electrode. On the electrode's surface the concentration is $\rho = 0$, and the rest of the domain is filled with $\rho > 0$. In this case, the behavior is studied based on the diffusion equation.

This MRT-LBM model solves the ADE using a curvilinear coordinate system that allows to maintain a computational domain where the points are uniformly distributed. The additional terms are well treated with the Multiple-Relaxation-Time collision operator that incorporates these effects locally. This model presents second-order accuracy for the two first examples, while for the last one, the accuracy is slightly decreased.

CHAPTER 4

An alternative lattice Boltzmann model for the advection-diffusion equation in generalized coordinates

In this chapter, we describe our proposed Lattice Boltzmann model for the simulation of the advection-diffusion equation on generalized coordinates and present some benchmarks. Our strategy to implement complex boundary conditions in an LBM is to choose coordinates that follow the boundaries' shape, to write down the partial differential equations on those coordinates and to introduce the extra terms so generated as a source term. So, the mesh in computational space is still squared/cubic and homogenously spaced, but in the physical space it is non-uniformly distributed and allows to represent the geometry more accurately. As benchmarks, we present a one-dimensional mesh stretching, a 2D Gaussian-like mesh stretching and a Polar coordinate transformation, to study the advection and diffusion of a Gaussian distribution profile.

4.1 Our proposal

Now that we have a model to simulate the advection-diffusion equation with a source term in Cartesian coordinates (Section 3.3), the strategy to extend the model to generalized coordinates is the following: We still consider in computer's memory a Cartesian grid of integer numbers (i, j, k) , but each point represents a place in real space through a smooth map $\vec{x} = (x, y, z) = (x(\eta_1, \eta_2, \eta_3), y(\eta_1, \eta_2, \eta_3), z(\eta_1, \eta_2, \eta_3))$, where $\eta_i = \eta_i(i, j, k)$ are the generalized coordinates. Because of the non-Cartesian coordinates, extra terms will appear in the conservation laws to be reproduced. Those terms will be added as source terms in the advection-diffusion equation. So, there is no need of modifying the velocities within the

cells. Moreover, for the case of the diffusion equation neither the equilibrium function nor the macroscopic quantities need to be modified, as we will show below.

Let us start with the advection-diffusion equation with a source term in Cartesian coordinates discussed in Section 3.3, which in index notation reads

$$\partial_t \rho + \partial_i(\rho u^i) = D \partial_i^2 \rho + S_i . \quad (4-1)$$

In order to write the above equation in terms of generalized coordinates, we need to consider the general spatial differential operators [43], i.e. the Laplace-Beltrami and divergence operators given, respectively, by

$$\nabla^2 \rho = \frac{1}{\sqrt{g}} \frac{\partial}{\partial \eta_i} (\sqrt{g} g^{ij} \frac{\partial \rho}{\partial \eta_j}) , \quad (4-2)$$

$$\nabla_i(\rho u^i) = \frac{\partial}{\partial \eta_i} (\rho u^i) + \Gamma_{ij}^i \rho u^j , \quad (4-3)$$

where g is the determinant of the metric tensor g^{ij} and Γ_{jk}^i are the Christoffel symbols. They are defined, respectively, as

$$g_{ij} = \sum_k \frac{\partial x_k}{\partial \eta_i} \frac{\partial x_k}{\partial \eta_j} , \quad (4-4)$$

$$\Gamma_{jk}^i = \frac{1}{2} g^{il} \left(\frac{\partial g_{kl}}{\partial \eta_j} + \frac{\partial g_{jl}}{\partial \eta_k} - \frac{\partial g_{jk}}{\partial \eta_l} \right) , \quad (4-5)$$

with x_k are the Cartesian coordinates and η_i the generalized coordinates. Thus, by replacing the Cartesian terms with the generalized formulation of the spatial differential operators, the ADE in generalized coordinates with a source term reads

$$\partial_t \rho + \partial_i(\rho u^i) = \frac{D}{\sqrt{g}} \partial_i(\sqrt{g} g^{ij} \partial_j \rho) - \Gamma_{ij}^i \rho u^j + S_{\text{phys}} , \quad (4-6)$$

where D is still the diffusion coefficient and S_{phys} is an actual physical source term for the lattice. At this point we will introduce the terms associated with the geometry into a new source term S , but as the new scheme proposed reproduces the ADE in **Cartesian** coordinates, we will have a diffusive Cartesian contribution that needs to be removed,

$$\partial_t \rho + \partial_i(\rho u^i) = \underbrace{D \partial_i^2 \rho - D \partial_i^2 \rho}_{\text{Cartesian contribution}} + \underbrace{\frac{D}{\sqrt{g}} \partial_i(\sqrt{g} g^{ij} \partial_j \rho) - \Gamma_{ij}^i \rho u^j + S_{\text{phys}}}_{\text{Geometry information}} . \quad (4-7)$$

The goal is to recover an ADE with a source term than can be simulated with an LBM in Cartesian coordinates,

$$\partial_t \rho + \partial_i(\rho u^i) = D \partial_i^2 \rho + S . \quad (4-8)$$

This is achieved, indeed, by defining a new source term

$$S = -D \partial_i^2 \rho + \frac{D}{\sqrt{g}} \partial_i(\sqrt{g} g^{ij} \partial_j \rho) - \Gamma_{ij}^i \rho u^j + S_{\text{phys}} \quad (4-9)$$

4.1.1 Computation of spacial derivatives within the lattice

The source term (Eq. (4-9)) requires to compute the spatial derivatives of some quantities (ρ and g_{ij}). To achieve that goal we use the following expression proposed by Thampi et al. [59]:

$$\partial_k \phi(\vec{x}) = \frac{1}{c_s^2 \Delta t} \sum_{\lambda}^m \omega_{\lambda} c_{\lambda}^k \phi(\vec{x} + c_{\lambda} \Delta t) , \quad (4-10)$$

where ϕ is any field whose value is known at every point (ρ , g_{ij} in our case), m is the number of velocities and c_{λ}^k is the k -th component of the velocity vector \vec{c}_{λ} . By using this tool it is also possible to compute gradients, divergences and curls in a DnQm scheme, respectively as

Gradient:

$$\nabla \phi(\vec{x}) = \frac{1}{c_s^2 \Delta t} \sum_{\lambda}^m \omega_{\lambda} \vec{c}_{\lambda} \phi(\vec{x} + c_{\lambda} \Delta t) + O(\nabla^3) , \quad (4-11)$$

Divergence:

$$\nabla \cdot \phi(\vec{x}) = \frac{1}{c_s^2 \Delta t} \sum_{\lambda}^m \omega_{\lambda} \vec{c}_{\lambda} \cdot \phi(\vec{x} + c_{\lambda} \Delta t) + O(\nabla^3) , \quad (4-12)$$

Rotational:

$$\nabla \times \phi(\vec{x}) = \frac{1}{c_s^2 \Delta t} \sum_{\lambda}^m \omega_{\lambda} \vec{c}_{\lambda} \times \phi(\vec{x} + c_{\lambda} \Delta t) . \quad (4-13)$$

4.1.2 LBM algorithm for the ADE on generalized coordinates

The computational algorithm to simulate the ADE with our proposed LBM is:

1. Initialize the variables, define the metric tensor g_{ij} and compute the Christoffel symbols Γ_{ij}^i associated to the geometry.
2. Impose the initial conditions. In this case, the distribution functions f_{λ} are defined as the equilibrium function $f_{\lambda}^{eq}(\rho_0, \vec{u})$, where ρ_0 is the initial distribution (usually given by a Gaussian distribution) and \vec{u} is the external advection velocity.
3. Save the data to files (if desired).
4. Collision step. This step includes the calculation of the source term. Eqs. (3-13 , 4-9)
5. Streaming step. The distribution functions are advected to adjacent cells. Eq.(3-13).
6. Impose boundary conditions.
7. Repeat steps (3.) through (6.) while $t < T_{\max}$.

4.2 Benchmarks

To validate our model, we simulate the time evolution of a Gaussian density profile in two and three dimensions, whose theoretical solution is given by [31]

$$\rho(\vec{x}, t) = \frac{A}{(1 + 2Dt/\sigma_0^2)^{n/2}} \exp\left[-\frac{(\vec{x} - \vec{x}_0 - \vec{u}_i t)^2}{2(\sigma_0^2 + 2Dt)}\right]. \quad (4-14)$$

Here, D is the diffusion coefficient, A is the maximal concentration, σ_0 is the initial width of the Gaussian distribution, n is the number of dimensions, \vec{x}_0 is the origin of the distribution and \vec{u} is the external advection velocity.

In all cases, the grid in the computer forms cubic cells of length one denoted by the indexes. The mesh nodes at the center of each cell are at coordinates (η_1, η_2, η_3) , which are integer numbers. Each node corresponds to a point $\vec{x} = (x, y, z) = (x(\eta_1, \eta_2, \eta_3), y(\eta_1, \eta_2, \eta_3), z(\eta_1, \eta_2, \eta_3))$ in real space. The transformation must be defined so that there exists a one-to-one correspondence between the square/cubic mesh in the computational domain and the curvilinear mesh in the physical domain. For example, in cylindrical coordinates η_1, η_2, η_3 would be represented by r, θ, z and a cylinder of radius a and length Z would be covered by a mesh of $L_r \times L_\theta \times L_z$ cells with $x = r \cos \theta$, $y = r \sin \theta$ and $z = z$, where $r = r_{\min} + i\Delta r$ (to avoid the singularity at the origin), $\theta = j\Delta\theta$ and $z = k\Delta z$, with $\Delta r = (a - r_{\min})/L_r$, $\Delta\theta = 2\pi/L_\theta$, $\Delta z = Z/L_z$.

4.2.1 Mesh stretching

As a first test, let us simulate the evolution of a Gaussian density distribution in two dimensions on a grid that is uniformly stretched along one axis only. The coordinate transformation is $x = C\eta_1$, $y = \eta_2$, where $\eta_1 = i\Delta\eta_1$ and $\eta_2 = j\Delta\eta_2$, leading to

$$\frac{dx}{d\eta_1} = C \quad , \quad \frac{dy}{d\eta_2} = 1 \quad . \quad (4-15)$$

If the differential d is replaced by the discrete Δ , from the previous expression, the relation between spacing emerges as,

$$\Delta x = C\Delta\eta_1 \quad , \quad \Delta y = \Delta\eta_2 \quad , \quad (4-16)$$

where $\Delta\eta_1$ and $\Delta\eta_2$ have physical units, for example $\Delta\eta_1 = \Delta\eta_2 = 1\text{cm}$. The diagonal metric tensor is

$$g_{ij} = \begin{pmatrix} C^2 & 0 \\ 0 & 1 \end{pmatrix} \quad (4-17)$$

and the Christoffel symbols are all null. The simulation domain (shown in Fig. (4-1)) is an array of $91 \times 91 \times 1$ cubic cells with periodic boundary conditions in all directions and a

D3Q19 scheme. We choose for this simulation $\Delta t = 1$ s. The initial width for the Gaussian is chosen $\sigma_0 = 9.1$ cm and the diffusion coefficient, as $D = 0.01 \text{ cm}^2/\text{s}$

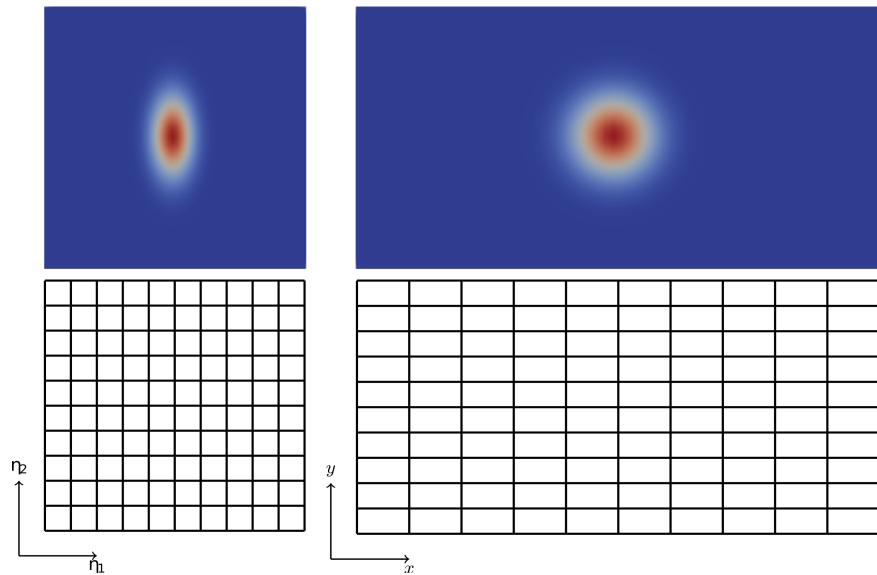


Figure 4-1: Diffusion of a Gaussian density distribution on a uniformly stretched mesh with the coordinate transformation $x = 2\eta_1$, $y = \eta_2$. Computational space (Left) and physical space (Right).

Figure 4-1 shows the density distribution after 1000 timesteps. It can be observed that our LBM preserves the isotropy in the evolution process, although the mesh is not isotropic in the physical space. By plotting the variance σ^2 of the marginal distribution along the x and y axis against time (Fig. 4-2) one can observe that the evolution remains isotropic for stretching factors up to $C = 2$, since all them show the same linear behavior $\sigma^2 = \sigma_0^2 + 2Dnt$ in both directions. For higher values than $C \gtrsim 2$ the information travels from one cell to another faster than the physical process and, the diffusion becomes anisotropic. Nevertheless, the slope remains the same, because the diffusion coefficient is $D = 0.01 \text{ cm}^2/\text{s}$ for every simulation.

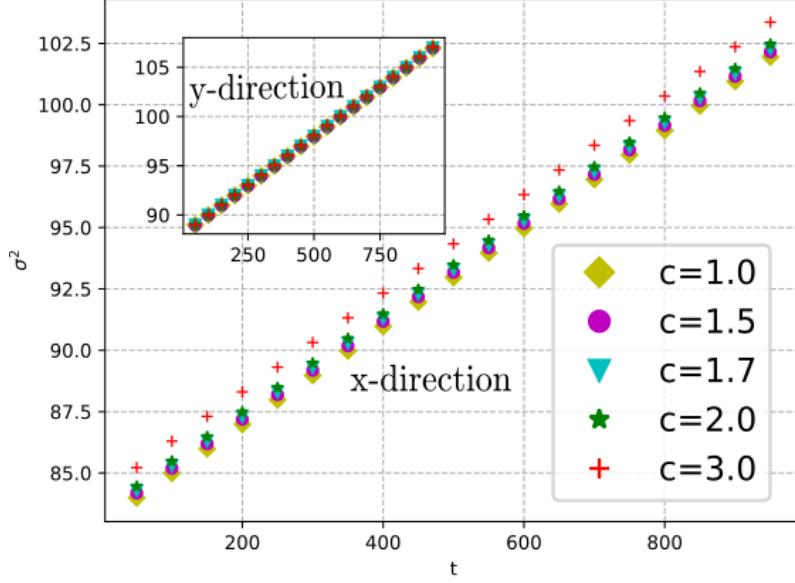


Figure 4-2: Variance of the marginal distributions on x and y against time for the diffusion of a two-dimensional Gaussian distribution on an uniformly stretched mesh. The diffusion in x -direction presents a good matching with the non-stretched mesh up to a scaling factor of $C = 2$, while in the y direction is isotropic for all scale factors.

4.2.2 Gaussian-like stretching

The second test we carry out corresponds to a non-homogenous mesh stretching in both x and y directions, described by the transformation

$$x = (\eta_1 - \eta'_1) - a \exp\left[-\frac{(\eta_1 - \eta'_1)^2}{2\varsigma^2}\right] \quad , \quad (4-18)$$

$$y = (\eta_2 - \eta'_2) - a \exp\left[-\frac{(\eta_2 - \eta'_2)^2}{2\varsigma^2}\right] \quad , \quad (4-19)$$

which leads to

$$\frac{dx}{d\eta_1} = 1 + a \frac{(\eta_1 - \eta'_1)}{\varsigma^2} \exp\left[-\frac{(\eta_1 - \eta'_1)^2}{2\varsigma^2}\right] \quad , \quad (4-20)$$

$$\frac{dy}{d\eta_2} = 1 + a \frac{(\eta_2 - \eta'_2)}{\varsigma^2} \exp\left[-\frac{(\eta_2 - \eta'_2)^2}{2\varsigma^2}\right] \quad , \quad (4-21)$$

where a , η'_i and ς are parameters controlling the stretching (for this simulation we choose $a = \varsigma = 10$ cm and $\eta'_1 = \eta'_2 = L/2$, with L the length of the computational array. Approximating

the differentials with discrete increments $d \rightarrow \Delta$, we have

$$\Delta x \simeq \left(1 + a \frac{(\eta_1 - \eta'_1)}{\varsigma^2} \exp\left[-\frac{(\eta_1 - \eta'_1)^2}{2\varsigma^2}\right] \right) \Delta\eta_1, \quad (4-22)$$

$$\Delta y \simeq \left(1 + a \frac{(\eta_2 - \eta'_2)}{\varsigma^2} \exp\left[-\frac{(\eta_2 - \eta'_2)^2}{2\varsigma^2}\right] \right) \Delta\eta_2, \quad (4-23)$$

with $\Delta\eta_1 = \Delta\eta_2 = 1$ cm. Eqs. (4-22) and (4-23) describe the change of the spacing in the physical plane (Δx , Δy) with relation to the uniform spacing in the computational plane ($\Delta\eta_1$, $\Delta\eta_2$). The non-zero components of the metric tensor are given by

$$g_{00} = \left(1 + a \frac{(\eta_1 - \eta'_1)}{\varsigma^2} \exp\left[-\frac{(\eta_1 - \eta'_1)^2}{2\varsigma^2}\right] \right)^2, \quad (4-24)$$

$$g_{11} = \left(1 + a \frac{(\eta_2 - \eta'_2)}{\varsigma^2} \exp\left[-\frac{(\eta_2 - \eta'_2)^2}{2\varsigma^2}\right] \right)^2, \quad (4-25)$$

and all Christoffel symbols are null. To guarantee that the cells do not overlap, the transformation should increase monotonically ($(\partial x / \partial \eta_1) > 0$). Moreover, the Courant criterion imposes an additional condition on the timestep, $(\Delta x) \geq 3\sigma$ with $\sigma = \sqrt{2D\Delta t}$, even for the smallest cell size Δx .

For our simulation we assume $\Delta t = 1$ s and an uniform diagonal advection velocity with components $u_x = u_y = 0.05$ cm/s. Nevertheless, this uniform velocity field in physical space is not uniform in the computacional space and, therefore, it must be transformed as $u'^\mu = (\partial\eta^\mu / \partial x^\nu) u^\nu$, where u'^μ is the velocity component in computational space and u^ν is the same component in physical space. [60]

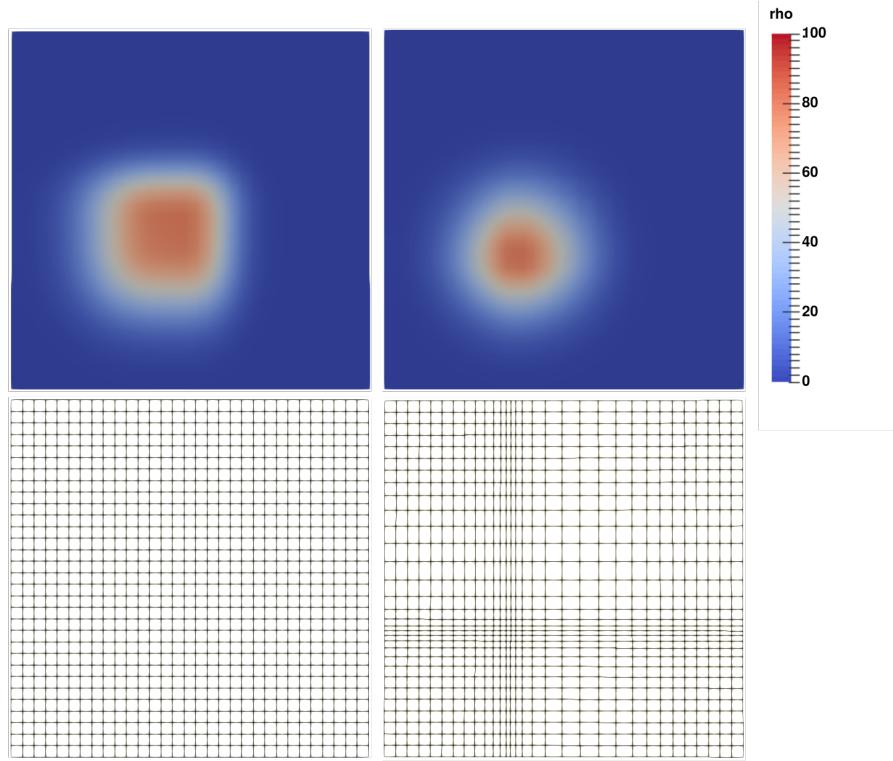


Figure 4-3: Two-dimensional Gaussian-like stretching after 1000 seconds on an uniform advection velocity field with components $u_x = u_y = 0.05$ cm/s. Evolution of a Gaussian distribution. We see the distribution distorted in the computational space (left) while in the physical space (right) it recovers the isotropic distribution. The parameters used are: $\sigma = 6.4$ cm, $x_0 = y_0 = 25.6$ cm, $D = 0.01$ cm²/s.

Figure (4-3) shows the evolution of a Gaussian distribution (Eq. 4-14) with initial width $\sigma = 6.4$ cm, $x_0 = y_0 = 25.6$ cm, after 1000 seconds on an uniform advection velocity field with components $u_x = u_y = 0.05$ cm/s. The diffusion coefficient is chosen as $D = 0.01$ cm²/s. The computational space used is $64 \times 64 \times 1$ cells with the Gaussian stretching from Eqs. (4-18), with $a = \varsigma = 6.4$ cm and $\eta'_1 = \eta'_2 = 0$. The timestep has been chosen as $\Delta t = 1$ s. It can be observed that the evolution in the physical space (right) is isotropic as desired whereas in the computational space (left) is distorted.

4.2.3 Convergence study

Maintaining the same physical system implies scaling both the cell length and the timestep. Let us consider, for instance, a cell of size Δx , a simulation evolving with a timestep Δt , a diffusion coefficient $D = 0.01 \Delta x^2 / \Delta t$ and an advection velocity $u = 0.05 \Delta x / \Delta t$. If we dou-

ble the number of nodes, the cell size es now $\Delta x/2$. Then, choosing a new timestep $\Delta t/4$ and a velocity $u = 0.025$ $(\Delta x/2)/(\Delta t/4)$ keeps the same D and u , and the same Péclet number, $\text{Pe} = uL/D$. All simulations started from the same initial distribution: a two-dimensional Gaussian profile with maximal amplitude $A = 100$ particles per cm^2 , $\sigma = 6.4\text{cm}$ centered at $x_0 = y_0 = 25.6\text{ cm}$, and they were evolved for the same time of 100 s, which corresponds to 100 timesteps for the mesh size of 32×32 ($\Delta\eta_1 = \Delta\eta_2 = 1\text{ cm}$) and to 6400 timesteps for the mesh size of 256×256 ($\Delta\eta_1 = \Delta\eta_2 = 0.125\text{ cm}$).

To perform the convergence test for our model we use the previous configuration (Gaussian-like stretching) and compare the analytical solution (ρ_a) with the numerical result (ρ_n) through the L_2 error norm [1]

$$\epsilon_\rho(t) := \sqrt{\frac{\sum_i (\rho_n(x, t) - \rho_a(x, t))^2}{\sum_i \rho_a(x, t)^2}} , \quad (4-26)$$

where the sums run along 1000 equally spaced fixed points on the diagonal from $(0,0)$ to (L_x, L_y) , indexed by i , but including only points where the analytical solution was greater than 5 particles per cm^2 . The sizes implemented are $L = (32, 48, 64, 80, 96, 128, 256)$. The results in Fig. 4-4 show a power-law relation between the error and the number of cells with an exponent -2.1535 , evidencing a second-order accuracy for this model [30].

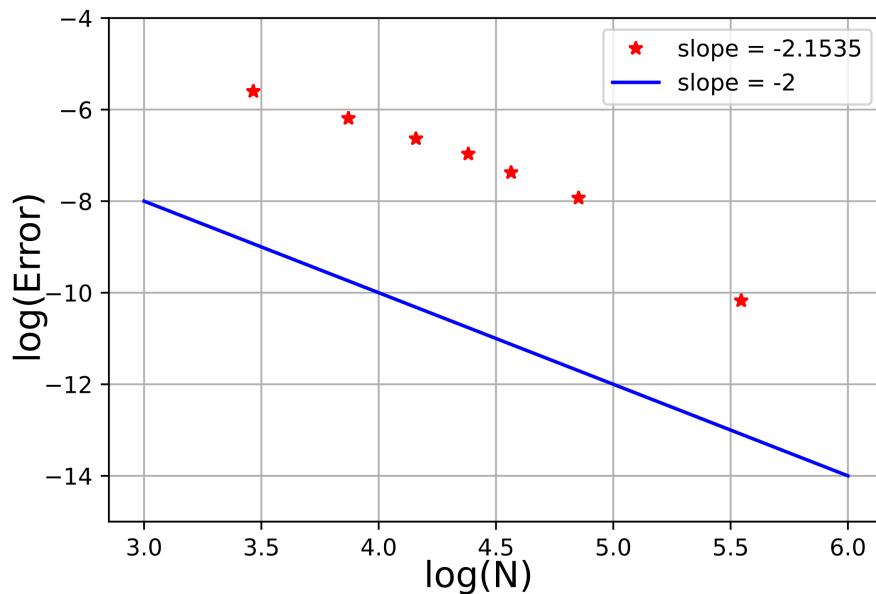


Figure 4-4: Convergence mesh study for the Gaussian-like stretching. Measured error for the concentration versus the number of nodes. $D = 0.01\Delta x^2/\Delta t$, $u_x = u_y = 0.05\Delta x/\Delta t$. This plot evidences the second-order accuracy of this LBM.

4.2.4 Polar coordinates

The next benchmark is performed on polar coordinates, described by the transformation

$$x = r \cos \theta \quad , \quad y = r \sin \theta \quad , \quad (4-27)$$

which leads to a diagonal metric tensor with components $g_{00} = 1$, $g_{11} = r^2$; therefore, the non-zero Christoffel symbols are $\Gamma_{\theta\theta}^r = -r$ and $\Gamma_{r\theta}^\theta = \Gamma_{\theta r}^\theta = 1/r$.

With this coordinates we can take advantage of the axial symmetry presented when the initial Gaussian distribution is centered at the origin, allowing us to represent this two-dimensional problem as a one-dimensional simulation on the radial coordinate. Of course, the transformation has a singularity at $r = 0$. To avoid this problem we consider our simulation domain starting from a minimal radius r_{\min} that should be as close as possible to the origin. Thus, the correspondence between the coordinates and the computational points is $r = r_{\min} + i\Delta r$, $\theta = j\Delta\theta$, with $\Delta r = (a - r_{\min})/L_r$, $\Delta\theta = 2\pi/L_\theta$.

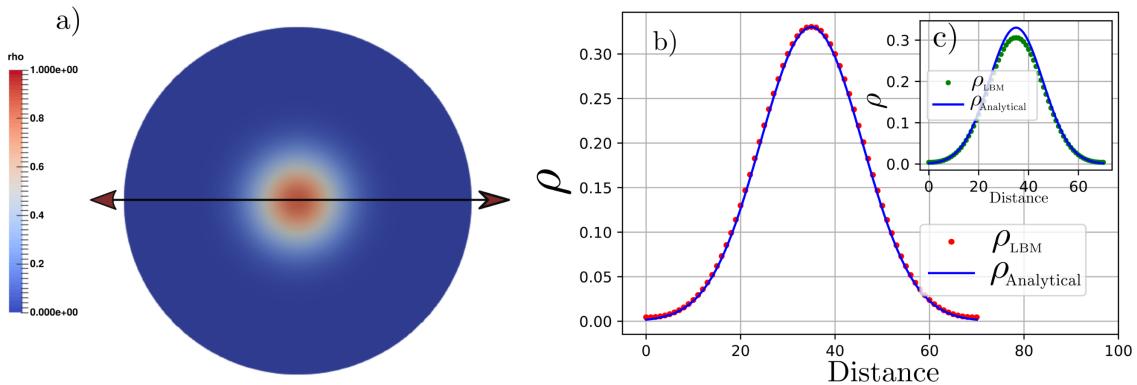


Figure 4-5: Evolution of a Gaussian distribution centered at $r = 0$. $D = 0.017 \text{ cm}^2/\text{s}$, $\sigma_0 = 22.5 \text{ cm}$, $\tau = 0.5051$, after 24000 timesteps with $\Delta t = 1 \text{ s}$. a) Colorplot of the evolution of a Gaussian distribution centered at $r = 0$. b) Normalized density profile (red-dotted) and analytical solution (blue-continued) across a diameter. c) Density profile (green-dotted) and analytical solution (blue-continued) across a diameter. After renormalization, the simulated profile fits the analytical solution.

For our simulations we choose a computational domain of $152 \times 1 \times 1$ cells, $a = 152 \text{ cm}$, $r_{\min} = 2 \text{ cm}$, $\Delta r = 1 \text{ cm}$ and $\Delta t = 1 \text{ s}$. We fix open boundary conditions at both ends, implemented by copying the information of the f_λ in the adjacent cell. The boundary conditions are chosen periodic in θ and z direction. Figure (4-5) shows the comparison of the D3Q7 scheme with the simulation of a Gaussian pulse with origin at $r = 0$, $\sigma_0 = 22.5 \text{ cm}$,

$D = 0.017 \text{ cm}^2/\text{s}$ and $\tau = 0.5051$ after 24000 s. Altough the simulated profile is lower than the analytical solution by an 8%, after renormalization it fits the analytical solution. This behavior may be an effect of the first-order boundary condition imposed at $r = r_{\min}$.

Our model also allows us to study the evolution of a distribution in polar coordinates whose origin is not centered at $r = 0$, that is without axial symmetry. For this case we use a computational domain $L_r \times L_\theta \times L_z = 100 \times 200 \times 1$ cells, with $r_{\min} = 2 \text{ cm}$. The boundary conditions are fixed open in $r = r_{\min}$ and $r = a$, and periodic in θ - and z -directions, as before. In this case, the larger the number of cells in the θ -direction, the more accurate the numerical solution is. Figure 4-6 shows the evolution of a Gaussian distribution centered at $r = a/2$, $\theta = \pi$ with $\sigma_0 = 22.5 \text{ cm}$ (left) after 8000 timesteps. The agreement between the numerical results and the anaytical solution is excellent.

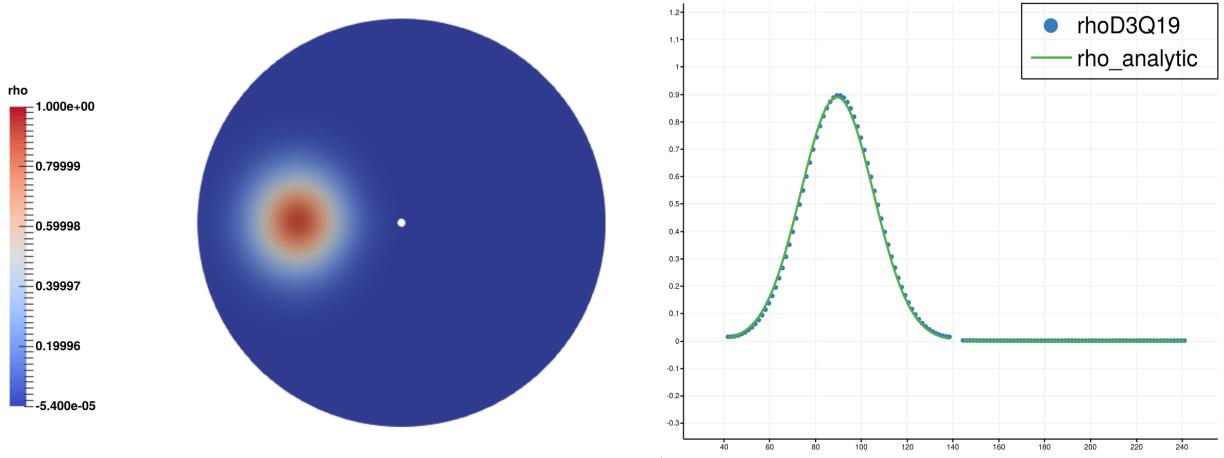


Figure 4-6: Evolution of a Gaussian distribution centered at $r = a/2$, $\theta = \pi$. $D = 0.017 \text{ cm}^2/\text{s}$, $\sigma_0 = 22.5 \text{ cm}$, $\tau = 0.5051$, after 8000 timesteps. Distribution on the circular plate (left). Profile along a diameter across the center of the Gaussian distribution (right).

CHAPTER 5

Applications of our model: Experimental comparison, torus and curved channels

In this Chapter we present some applications of our model. Taking advantage of the axial symmetry we simulate the advection and diffusion of a disk of Helium ions on a Poiseuille flow in a cylindrical pipe as a two-dimensional problem. This simulation reproduces perfectly the experimental results using less computational resources than a Cartesian LBM with the same simulation parameters. Next, as an example of a full three-dimensional problem we simulate the advection-diffusion of a Gaussian pulse inside a torus with a Poiseuille-like velocity flow. Finally, we simulate a Gaussian pulse of contaminant in a sinusoidal curved channel that resembles a river. In both cases, the simulation resembles qualitatively the expected results.

5.1 A comparison with experimental data

Our next step is to compare our simulations with an experimental result. With this purpose, let us consider the experiment referenced by Jarvis [2] and performed by Ferguson, Fehsenfeld and Schmeltekopf [61], where a thin Helium ion sheet difuses on a Poiseuille flow inside a cylindrical tube of radius a and length L_Z . A thin sheet of ions is generated at $z = 0$ at time $t = 0$ by a pulsed beam of electrons perpendicular to the axis of the tube. The ion sheet is transported and deformed along the cylindrical pipe by both diffusion and advection on the non-uniform velocity field. In our simulation we model the initial ion sheet as a Gaussian distribution in the z -direction,

$$\rho(r, \theta, z) = \exp\left(-\frac{(z - z_0)^2}{2\sigma_0^2}\right), \quad (5-1)$$

that only depends on z , where $\sigma_0 = 1.65$ cm and $z_0 = 10.8$ cm. The external advection velocity is given by a Poiseuille flow,

$$u_z(r) = 2V \left(1 - \left(\frac{r}{a} \right)^2 \right) , \quad (5-2)$$

where $V = 6800$ cm/s and $a = 4$ cm. The diffusion coefficient is $D = 2400$ cm²/s.

We simulated this phenomenon using two different LBM models, both with a D3Q7 LBM scheme: a Cartesian model, where the tube is described by a staircase approximation, and our LBM model using cylindrical coordinates,

$$x = r \cos \theta \quad , \quad y = r \sin \theta \quad , \quad z = z . \quad (5-3)$$

In this case the diagonal metric tensor has the components $g_{00} = 1$, $g_{11} = r^2$, $g_{22} = 1$ and the Christoffel symbols are $\Gamma_{\theta\theta}^r = -r$ and $\Gamma_{r\theta}^\theta = \Gamma_{\theta r}^\theta = 1/r$. The model also allows us to take advantage of the axial symmetry by reducing the three-dimensional problem to only two dimensions (r and z).

The parameters for our simulations are shown in Table 5-1:

Quantity	Comp. units	Exp. value
D	0.0048 cells ² /timestep	2400cm ² /s
V	0.004896 cells/timestep	6800cm/s
a	11 cells	3.96cm \approx 4cm
Z	430 cells	154.8cm
$\Delta x, \Delta r, \Delta z$	1 cell	0.36cm
Δt	1 timestep	2.592×10^{-7} s
t_{\max}	57000 timestep	0.147744s

Table 5-1: Parameters used for the simulations and their experimental values.

For the Cartesian model we use a computational array of $L_x \times L_y \times L_z = 23 \times 23 \times 430$ cells, with periodic boundary conditions in z -direction and Dirichlet boundary conditions ($\rho_s = 0$) on the cylinder surface described by $(x-a)^2 + (y-a)^2 = a^2$, with $x = i\Delta x$, $y = j\Delta y$, and $\Delta x = \Delta y = \Delta z = 0.36$ cm. The initial point of the Gaussian distribution is chosen $z_0 = 10.8$ cm to avoid numerical errors originated from the periodic boundary conditions. For the model in cylindrical coordinates, the computational array is $L_r \times L_\theta \times L_z = 12 \times 1 \times 430$ cells. To avoid the singularity at $r = 0$ we choose the simulation domain from $r_{\min} = \Delta z = 0.36$ cm to $r = a$. The boundary conditions are periodic in z - and θ directions, open at $r = r_{\min}$ and Dirichlet ($\rho_{r=a} = 0$) at $r = a$.

Figure 5-1 shows the ion sheet distribution simulated with both models.



Figure 5-1: Simulation of the advection-diffusion evolution of an Helium ion sheet inside a cylindrical tube with a Poiseuille velocity field. The simulation parameters are shown in Table 5-1 and in the text thereafter. Figure shows a) initial state, b) and c) shows the snapshots of the evolution after 0.001995 s with a Cartesian LBM model and Cylindrical LBM model, respectively.

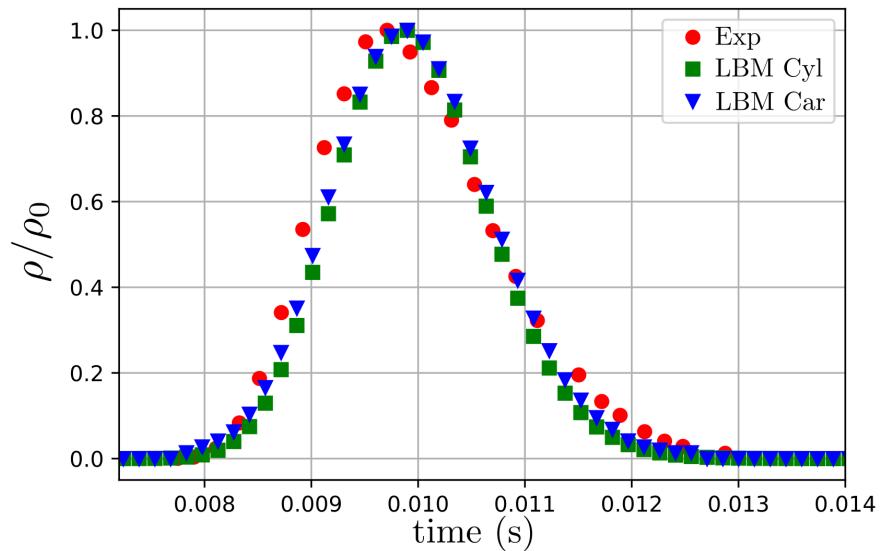


Figure 5-2: Comparison between the experimental data from [2] (red), our LBM model in cylindrical coordinates (green) and an LBM in Cartesian coordinates (blue). The values used for this simulation are shown in Table 5-1.

To compare the evolution of the ion sheet along the Poiseuille flow from both models with the experimental data, we measure the concentration on the axis at $z_m = 123.8$ cm and plot their normalized concentrations against the time until $t_{\max} = 0.1477$ s. The results

are shown in Fig. 5-2. We observe a good agreement between the two computational models and the experimental data. Since the experimental data is taken at $r = 0$ and our model in cylindrical coordinates has a singularity at the origin, the measurement point is not exactly located on the axis, but at r_{\min} . Although the Cartesian model allows us to take the measurement exactly in $r = 0$, it requires a computational array 48 times larger than the one implemented in cylindrical coordinates. That is the advantage of using cylindrical coordinates. In addition, the cylindrical model captures the boundary condition at $r = a$ without any staircase approximation.

5.2 Torus

As the next geometry implemented, we choose a torus, which is easily described by the coordinates

$$\begin{aligned} x &= (R + r \cos \phi) \cos \theta , \\ y &= (R + r \cos \phi) \sin \theta , \\ z &= r \sin \phi \end{aligned} \quad (5-4)$$

with $r \in (0, a)$, $\theta \in (0, 2\pi)$, $\phi \in (0, 2\pi)$ and $R > a$ the constant distance between the torus' axis of symmetry and the circle of all points with $r = 0$. The metric tensor is

$$g_{ij} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & (R + r \cos \phi)^2 & 0 \\ 0 & 0 & r^2 \end{pmatrix} . \quad (5-5)$$

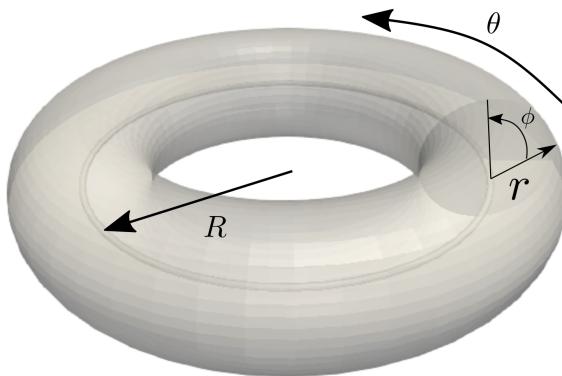


Figure 5-3: Torus Geometry.

We calculate numerically the Christoffel symbols (Eq. (4-5)) using the procedure described by Eq. (4-10). However, their analytical expression is given by

$$\begin{aligned}\Gamma_{\theta\theta}^r &= -(R + r \cos \phi) \cos \phi \quad , & \Gamma_{\phi\phi}^r &= -r \quad , \\ \Gamma_{\theta r}^\theta &= \frac{\cos \phi}{R + r \cos \phi} \quad , & \Gamma_{\phi\theta}^\theta &= -\frac{r \sin \phi}{R + r \cos \phi} \quad , \\ \Gamma_{\theta\theta}^\phi &= \frac{(R + r \cos \phi) \sin \phi}{r} \quad , & \Gamma_{\phi r}^\phi &= \frac{1}{r} \quad .\end{aligned}$$

For our simulation we use a D3Q7 scheme and an array of $L_r \times L_\theta \times L_\phi = 20 \times 50 \times 50$ cells. The parameters are $D = 1 \text{ cm}^2/\text{s}$, $\sigma = 2.1 \text{ cm}$, $R = 60 \text{ cm}$, $a = 21 \text{ cm}$, $r_{\min} = 1 \text{ cm}$, $\Delta t = 0.078 \text{ s}$, $\Delta r = 1 \text{ cm}$, and a simulation time of 3000 timesteps ($T_{\max} = 23.4 \text{ s}$). The external velocity follows a Poiseuille profile

$$u_\theta(r) = 2V \left(1 - \left(\frac{r}{a} \right)^2 \right) \quad , \quad (5-6)$$

where $V = 0.641 \text{ cm/s}$. We choose open boundary conditions at both ends in r -direction and periodic boundary conditions in both θ and ϕ direction. To deal with the singularity at $r = 0$ we also consider a minimum radius $r_{\min} = 1 \text{ cm}$.

The initial Gaussian distribution is

$$\rho(r, \theta, \phi) = \exp \left(-\frac{(\theta - \theta_0)^2}{2\sigma_0^2} \right) \quad , \quad (5-7)$$

which only depends on θ . Figure 5-4 shows the temporal evolution of the density with both advection and diffusion. We observe that a homogeneous distribution of the density ρ is reached when a Poiseuille-like flow is considered. For the evolution of the Gaussian density profile with only diffusion we considered a total simulation time of 6000 timesteps (46.8 s), as shown in Figure 5-5. The fluid flow in a curved channel is characterized by the Dean number, defined as $\text{De} = \text{Re} \sqrt{d/R_c}$, with Re the Reynolds number, d the channel width and R_c the channel curvature. For low Dean numbers, the flow is laminar and resembles a Poiseuille flow, but for higher Dean numbers centripetal forces have an effect on the axial component of the velocity field [62].

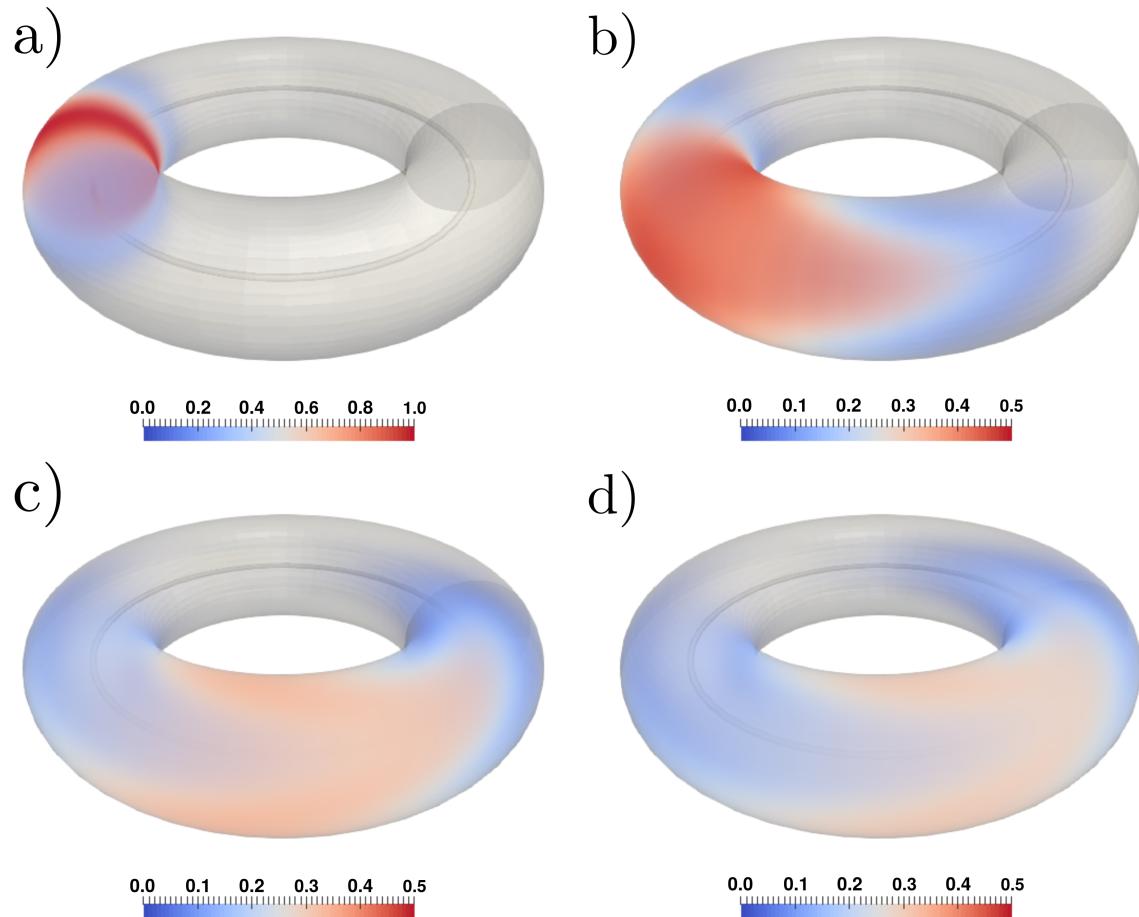


Figure 5-4: Evolution of a Gaussian distribution in a Torus with a Poiseuille-like velocity in θ -direction, with $V = 0.641$ cm/s, $D = 1$ cm²/s, $\sigma = 2.1$ cm, $R = 60$ cm, $a = 21$ cm, $\Delta t = 0.0078$ s, and a simulation time of 3000 timesteps. a) $t = 0$ s, b) $t = 7.8$ s, c) $t = 15.6$ s and d) $t = 23.4$ s.

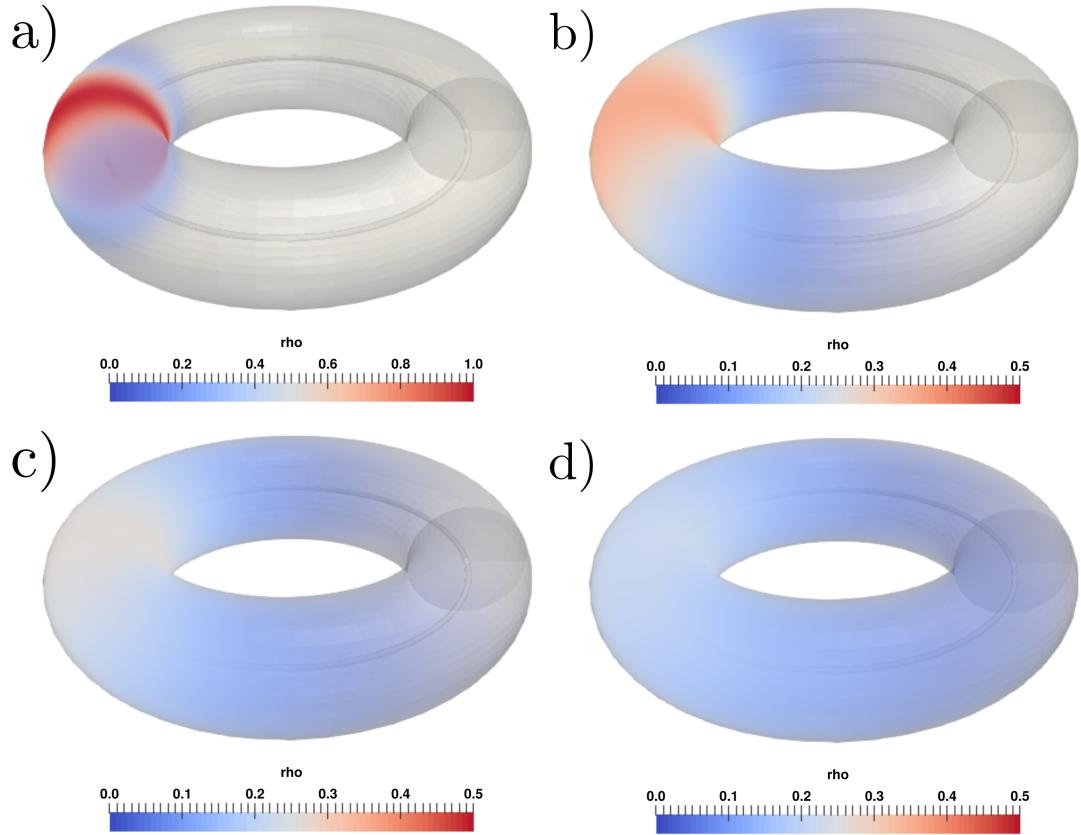


Figure 5-5: Evolution of a Gaussian distribution in a Torus with diffusion only. $D = 1 \text{ cm}^2/\text{s}$, $\sigma = 2.1 \text{ cm}$, $R = 60 \text{ cm}$, $a = 21 \text{ cm}$, $\Delta t = 0.0078 \text{ s}$, and a simulation time of 6000 timesteps. a) $t = 0 \text{ s}$, b) $t = 15.6 \text{ s}$, c) $t = 31.2 \text{ s}$ and d) $t = 46.8 \text{ s}$.

5.3 A curved channel

As a last implementation we built a coordinate transformation to generate a sinusoidal channel, by starting from cylindrical coordinates and introducing some additional terms to get the desired geometry. Here the coordinates are r , θ and ζ . We use the following coordinate transformation [63], [64]:

$$\begin{aligned} x &= \frac{r \cos(\theta)}{L} + b \sin(m\zeta) \quad , \\ y &= r \sin(\theta) \quad , \\ z &= \zeta - \frac{mbr \cos(\theta) \cos(m\zeta)}{L} \quad , \end{aligned}$$

where $L = \sqrt{(1 + m^2 b^2 \cos^2(m\zeta))}$ and m and b are parameters that define the amplitude and period of the pipe, with $r \in (0, a)$, $\theta \in (0, 2\pi)$, $\zeta \in (0, L_{\max})$. This transformation leads to the following diagonal metric tensor:

$$g_{00} = 1 \quad , \tag{5-8}$$

$$g_{11} = r^2 \quad , \tag{5-9}$$

$$g_{22} = 1 + bm^2 \left[b \cos^2(m\zeta) + r \cos \theta \sin(m\zeta) \left(\frac{2}{L} + \frac{bm^2 \cos \theta \sin(m\zeta)}{L^4} \right) \right] \quad . \tag{5-10}$$

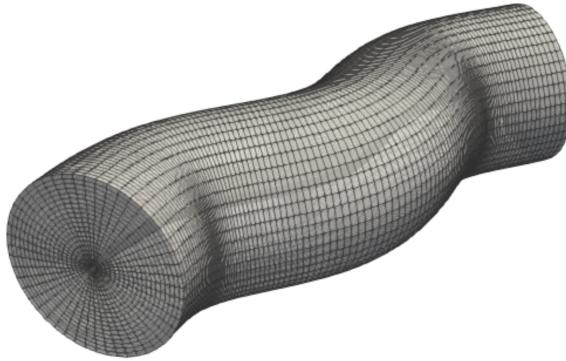


Figure 5-6: Geometry of the sinusoidal curved channel.

The Christoffel symbols (Eq. (4-5)) were computed numerically using the procedure described by Eq. (4-10), although it is also possible to use their analytical expression given

by

$$\begin{aligned}
\Gamma_{\theta\theta}^r &= -r \quad , \\
\Gamma_{\zeta\zeta}^r &= -\frac{bm^2 C_\theta S_{m\zeta} (1 + 2b^2 m^2 C_{m\zeta}^2 + b^4 m^4 C_{m\zeta}^4 + bm^2 r L C_\theta S_{m\zeta})}{L^5} \quad , \\
\Gamma_{\theta r}^\theta &= \frac{1}{r} \quad , \\
\Gamma_{\zeta\zeta}^\theta &= \frac{bm^2 S_\theta S_{m\zeta} (1 + 2b^2 m^2 C_{m\zeta}^2 + b^4 m^4 C_{m\zeta}^4 + bm^2 r L C_\theta S_{m\zeta})}{r L^5} \quad , \\
\Gamma_{\zeta r}^\zeta &= \frac{bm^2 C_\theta S_{m\zeta} (1 + 2b^2 m^2 C_{m\zeta}^2 + b^4 m^4 C_{m\zeta}^4 + bm^2 r L C_\theta S_{m\zeta})}{L^5 \left(1 + bm^2 \left(b C_{m\zeta}^2 + r C_\theta S_{m\zeta} \left(\frac{2}{L} + \frac{bm^2 r C_\theta S_{m\zeta}}{L^4}\right)\right)\right)} \quad , \\
\Gamma_{\zeta\theta}^\zeta &= -\frac{bm^2 r S_\theta S_{m\zeta} (1 + 2b^2 m^2 C_{m\zeta}^2 + b^4 m^4 C_{m\zeta}^4 + bm^2 r L C_\theta S_{m\zeta})}{L^5 \left(1 + bm^2 \left(b C_{m\zeta}^2 + r C_\theta S_{m\zeta} \left(\frac{2}{L} + \frac{bm^2 r C_\theta S_{m\zeta}}{L^4}\right)\right)\right)} \quad , \\
\Gamma_{\zeta\zeta}^\zeta &= \frac{bm^3 C_{m\zeta} \left(\frac{(r+b^2 m^2 r) C_\theta}{L^3} - b S_{m\zeta} - \frac{bm^2 \rho^2 (-2-3b^2 m^2 + b^2 m^2 C_{m\zeta}) C_\theta^2 S_{m\zeta}}{2L^6}\right)}{1 + bm^2 \left(b C_{m\zeta}^2 + r C_\theta S_{m\zeta} \left(\frac{2}{L} + \frac{bm^2 r C_\theta S_{m\zeta}}{L^4}\right)\right)} \quad ,
\end{aligned}$$

where we used the notation C_θ and S_θ to represent $\cos \theta$ and $\sin \theta$, respectively. The external velocity field is Poiseuille-like in the axial direction

$$u_\zeta(r) = 2V \left(1 - \left(\frac{r}{a}\right)^2\right) \quad , \quad (5-11)$$

with $V = 2.94$ cm/s ($V = 0.05 \Delta r^2 / \Delta t$) and a the pipe radius. For our simulations we built a computational array of $L_r \times L_\theta \times L_\zeta = 20 \times 50 \times 160$ cells with a D3Q7 scheme. We impose open boundary conditions at $\zeta = 0$, $\zeta = L_{\max}$, bounce-back [1] at $r = r_{\min}$, Dirichlet boundary conditions ($\rho_a = 0$) at $r = a$ and periodic boundary conditions in θ direction. The parameters employed are $r_{\min} = 1$ cm, $m = 0.08$, $b = 6.5$, $\Delta r = 1$ cm, $\Delta t = 0.017$ s and a diffusion coefficient $D = 1$ cm²/s ($D = 0.017 \Delta r^2 / \Delta t$), and a simulation time of 34 s (2000 timesteps).

Figures 5-8 and 5-7 show the time-evolution of an initial Gaussian distribution

$$\rho(r, \theta, \zeta) = \exp \left(-\frac{(z(r, \theta, \zeta) - z_0(r, \theta, \zeta))^2}{2\sigma_0^2} \right) \quad (5-12)$$

with $z_0 = 20$ cm, $\sigma_0 = 2.1$ cm. We observe that the distribution evolves as expected for an advection-diffusion process, and it is possible to determine where the concentration is higher. This approach is valid for laminar and low Dean number flows, since the velocity is defined as a Poiseuille-like profile. We highlight the fact that it was not necessary to use complex algorithms to define the evolution of the LBM, because the curvilinear coordinates avoid the complex boundary conditions arising from a staircase approximation.

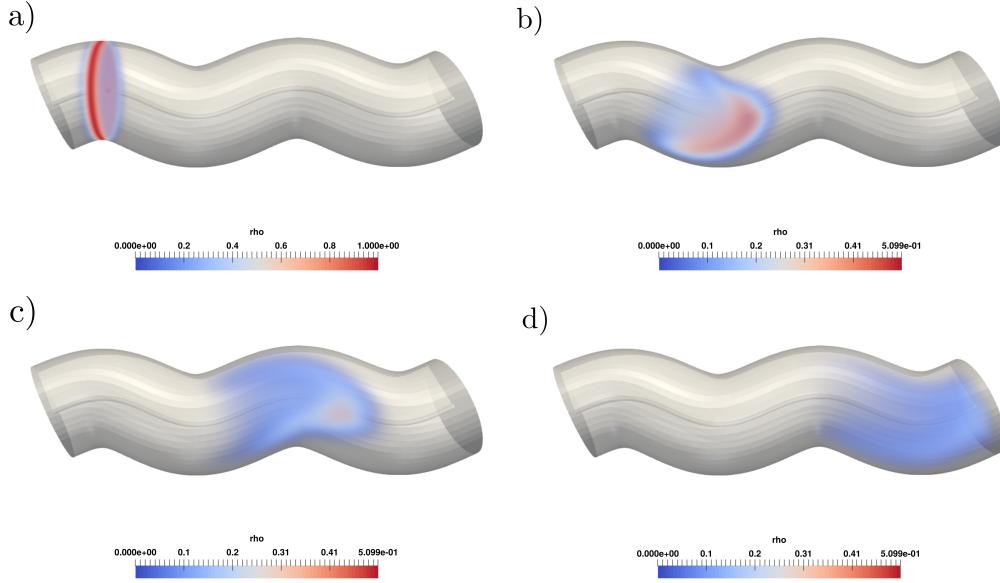


Figure 5-7: Evolution of a Gaussian distribution in a Sinusoidal-like channel with a Poiseuille-like velocity field along the longitudinal coordinate ζ . The parameters employed are $r_{\min} = 1\text{cm}$, $m = 0.08$, $b = 6.5$, $\Delta r = 1\text{ cm}$, $\Delta t = 0.017\text{ s}$ and a diffusion coefficient $D = 1\text{ cm}^2/\text{s}$ ($D = 0.017 \Delta r^2 / \Delta t$). Snapshots from upside at a) $t = 0\text{ s}$, b) $t = 8.5\text{ s}$, c) $t = 17\text{ s}$, d) $t = 25.5\text{ s}$.

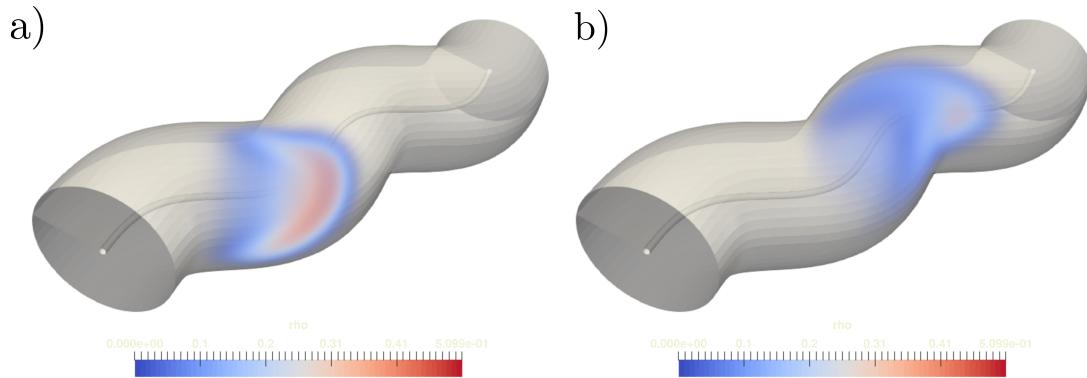


Figure 5-8: Evolution of a Gaussian distribution in a Sinusoidal-like channel with a Poiseuille-like velocity field along the longitudinal coordinate ζ . The parameters employed are $r_{\min} = 1\text{ cm}$, $m = 0.08$, $b = 6.5$, $\Delta r = 1\text{ cm}$, $\Delta t = 0.017\text{ s}$ and a diffusion coefficient $D = 1\text{ cm}^2/\text{s}$ ($D = 0.017 \Delta r^2 / \Delta t$). Snapshots at a) $t = 8.5\text{ s}$ and b) $t = 17\text{ s}$.

CHAPTER 6

Conclusions

We developed an alternative three-dimensional lattice Boltzmann model to simulate the advection-diffusion equation (ADE) in generalized coordinates. The computational mesh maintains the usual cubic cells, velocity schemes and boundary conditions of an LBM in Cartesian coordinates, and the geometry is introduced as an additional source in terms of the metric tensor and the Christoffel symbols. Once the transformation to the desired coordinate system is established and the metric and the Christoffel symbols are computed (which can also be done numerically), the proposed model reproduces the advection-diffusion equation on the desired coordinate system with second order accuracy. This allows to simulate complex geometries without using staircase approximations and their inherent complex boundary conditions. Moreover, since the proposed model uses a constant and uniform relaxation time, it constitutes a simpler alternative to previous works.

In all tested cases we found an excellent agreement between our model and the theoretical predictions, or even compared against experimental results. When simulating the advection-diffusion of an initial Gaussian distribution in two dimensions on a uniform velocity field, we obtained a perfect agreement with theoretical expectations, either by using a simple one dimensional stretching, polar coordinates or a Gaussian-like mesh stretching (a much more complex geometry). This last case was chosen to perform a convergence test, founding that the error decays as a second-order power law with the number of nodes. Moreover, our LBM model in polar coordinates allows us to take advantage of the axial symmetry when reproducing the free diffusion of a two-dimensional Gaussian centered at the origin as a one-dimensional problem. That property allowed us, too, to simulate the advection and diffusion of a disk of Helium ions on a Poiseuille flow in a cylindrical pipe as a two-dimensional problem. This simulation reproduces perfectly the experimental results reported by Ferguson et al. [61] and Jarvis [2] using 48 times less cells and almost 6 times less computational time than a Cartesian LBM with the same simulation parameters and resolution. Next, as an example of a full three-dimensions problem we simulated the advection-diffusion of a Gaus-

sian pulse inside a torus with a Poiseuille-like velocity flow. We also simulated a Gaussian pulse of contaminant in a sinusoidal curved channel that resembles a river. In both cases, the simulation resembles qualitatively the expected results.

The benchmarks listed in the previous paragraph evidence the advantages of using our proposed model. The model allows to perform mesh refinements without losing isotropy, to implement curved geometries by avoiding staircase approximations and to take advantage of geometrical symmetries in the problem (like axial). The way we defined our model facilitates to deal with real physical units, (like centimeters, seconds, etc) - something that is not obvious when dealing with non-uniform cell sizes -, making easier the comparison with experimental data. Furthermore, the last benchmark exemplifies a future application of this model, namely the simulation of the advection-diffusion of contaminants in a real river, something that, of course, will require a more detailed definition of the geometry and the places where a high resolution is required.

A future improvement for our model would be to couple an LBM for fluids in generalized coordinates, like the one developed by Mendoza [37]. This would allow to simulate advection-diffusion on non-stationary fluxes with higher Dean numbers. In addition, modelling with coordinates defined by some parameters that could change in time allows to simulate the advection-diffusion in time-varying channels, like pulsating arterials. In this case, it could be possible to implement dynamic geometries where it would be necessary to compute at each timestep the metric tensor and Christoffel symbols. Another possible improvement would be the inclusion of physical sources or sinks (like drainages for density or heaters for temperature), or even external forces such as gravity that may affect the behavior of the scalar field, an ingredient that is fundamental in sedimentation processes. Our model could be of great advantage in those problems where the advection-diffusion is relevant, like the temperature distribution inside a cooling coil or magnetic resonator, or in the plasma inside a Spheromak or Tokamak, which would also require a two-directional coupling between our LBM model and the one for the fluid flow in generalized coordinates. All these problems would be subject of further work.

The proposed model constitutes a valuable and easy-to-implement alternative to the simulation of advection-diffusion processes in three dimensions with LBM on complex curvilinear geometries. Because this phenomenon is relevant to many scientific, industrial and environmental applications, we expect it would be of great usefulness in future research.

APPENDIX A

Chapman-Enskog expansion for the advection-diffusion equation with a forcing term

In this Appendix we review the Chapman-Enskog expansion from B. Shi et al. [30].

A.1 Chapman-Enskog expansion

With the purpose of being able to study physical phenomena occurring at different timescales, the time derivative and the distribution function are expanded until 2nd order and the spatial derivative until 1st order as

$$f_\lambda = f_\lambda^{(0)} + \epsilon f_\lambda^{(1)} + \epsilon^2 f_\lambda^{(2)}, \quad (\text{A-1})$$

$$\partial_t = \epsilon \partial_{t_1} + \epsilon^2 \partial_{t_2}, \quad (\text{A-2})$$

$$\nabla = \epsilon \nabla_1, \quad S = \epsilon S^{(1)}, \quad (\text{A-3})$$

where ϵ is known as the *Knudsen number*, which relates the molecular mean free path with a physical length. Now, replacing Eq. (A-3) in Eqs. (3-6) and (3-14), the following expression is obtained:

$$\sum_\lambda f_\lambda = \sum_\lambda (f_\lambda^{(eq)} + \epsilon f_\lambda^{(1)} + \epsilon^2 f_\lambda^{(2)}) = \rho, \quad (\text{A-4})$$

$$f_\lambda^{(k)} = 0 \text{ for } k \geq 2 \quad (\text{A-5})$$

and

$$\sum_\lambda S_\lambda^{(1)} = S^{(1)}, \quad \sum_\lambda \vec{c}_\lambda S_\lambda^{(1)} = \frac{\tau - \frac{1}{2}}{\tau - \frac{\theta}{2}} S^{(1)} \vec{u}, \quad (\text{A-6})$$

where

$$S_\lambda^{(1)} = \omega_\lambda S^{(1)} \left(1 + \frac{\tau - \frac{1}{2}}{\tau - \frac{\theta}{2}} \frac{\vec{c}_\lambda \cdot \vec{u}}{c_s^2} \right) . \quad (\text{A-7})$$

Now, with these expressions in mind, a Taylor expansion of Eq. (3-13) reads

$$\Delta t D_\lambda f_\lambda + \frac{\Delta t^2}{2} D_\lambda^2 f_\lambda + \dots = -\frac{1}{\tau} [f_\lambda - f_\lambda^{eq}] + \Delta t S_\lambda + \frac{\Delta t}{2} \bar{D}_\lambda S_\lambda , \quad (\text{A-8})$$

where $D_\lambda = (\partial_t + \vec{c}_\lambda \cdot \nabla)$. Taking into account the different orders of the time derivative, let us denote $D_{1\lambda} = (\partial_{t_1} + \vec{c}_\lambda \cdot \nabla_1)$ and $\bar{D}_{1\lambda} = (\partial_{t_1} + \theta \vec{c}_\lambda \cdot \nabla_1)$. Substituting these expressions and equating the orders of the same magnitude, we obtain

$$\epsilon : \quad D_{1\lambda} f_\lambda^{eq} = -\frac{1}{\tau} f_\lambda^{(1)} + S_\lambda^{(1)} , \quad (\text{A-9a})$$

$$\epsilon^2 : \quad \partial_{t_2} f_\lambda^{eq} + D_{1\lambda} f_\lambda^{(1)} + \frac{\Delta t}{2} D_{1\lambda}^2 f_\lambda^{eq} = -\frac{1}{\Delta t \tau} f_\lambda^{(2)} + \frac{\Delta t}{2} \bar{D}_{1\lambda} S_\lambda^{(1)} . \quad (\text{A-9b})$$

Combining both (A-9) equations to eliminate second-order derivatives, one obtains

$$\partial_{t_2} f_\lambda^{eq} + \left(1 - \frac{1}{2\tau} \right) D_{1\lambda} f_\lambda^{(1)} + \frac{\Delta t}{2} D_{1\lambda} S_\lambda^{(1)} = -\frac{1}{\tau \Delta t} f_\lambda^{(2)} + \frac{\Delta t}{2} \bar{D}_{1\lambda} S_\lambda^{(1)} , \quad (\text{A-10})$$

which could be simplified as

$$\partial_{t_2} f_\lambda^{eq} + \left(1 - \frac{1}{2\tau} \right) D_{1\lambda} f_\lambda^{(1)} = -\frac{1}{\tau \Delta t} f_\lambda^{(2)} + \frac{\Delta t}{2} (\theta - 1) \vec{c}_\lambda \cdot \nabla S_\lambda^{(1)} . \quad (\text{A-11})$$

Summing over λ and employing Eqs. (3-6) and (A-6), Eqs. (A-9a) and (A-11) give

$$\partial_{t_1} \rho + \nabla_1 \cdot (\rho \vec{u}) = S_\lambda^{(1)} , \quad (\text{A-12})$$

$$\partial_{t_2} \rho + \left(1 - \frac{1}{2\tau} \right) \nabla_1 \cdot \sum_\lambda \vec{c}_\lambda f_\lambda^{(1)} = \frac{\Delta t}{2} (\theta - 1) \frac{\tau - \frac{1}{2}}{\tau - \frac{\theta}{2}} \nabla_1 \cdot (S_\lambda^{(1)} \vec{u}) . \quad (\text{A-13})$$

The unknown term $\sum_\lambda \vec{c}_\lambda f_\lambda^{(1)}$ can be computed by using Eqs. (A-9a), (3-6) and (A-6), obtaining

$$\sum_\lambda \vec{c}_\lambda f_\lambda^{(1)} = -\tau \Delta t \sum_\lambda \vec{c}_\lambda [(\partial_{t_1} + \vec{c}_\lambda \cdot \nabla_1) f_\lambda^{eq} - S_\lambda^{(1)}] \quad (\text{A-14})$$

$$= -\tau \Delta t \left(\partial_{t_1} (\rho \vec{u}) + \nabla_1 \cdot (\rho \vec{u} \vec{u} + c_s^2 \rho \mathbf{I}) - \frac{\tau - \frac{1}{2}}{\tau - \frac{\theta}{2}} S_\lambda^{(1)} \vec{u} \right) . \quad (\text{A-15})$$

It is possible to consider that the velocity is smaller than the sound speed in the lattice; therefore Eq. (A-15) can be approximated as

$$\sum_{\lambda} \vec{c}_{\lambda} f_{\lambda}^{(1)} = -\tau \Delta t \left(\underbrace{\vec{u} [\partial_{t_1} \rho + \nabla_1 \cdot (\rho \vec{u})]}_{S_{\lambda}^{(1)}} + c_s^2 \nabla_1 \rho - \frac{\tau - \frac{1}{2}}{\tau - \frac{\theta}{2}} S_{\lambda}^{(1)} \vec{u} \right) \quad (\text{A-16})$$

$$= -\tau \Delta t \left(c_s^2 \nabla_1 \rho + \frac{1 - \theta}{2(\tau - \frac{\theta}{2})} S_{\lambda}^{(1)} \vec{u} \right). \quad (\text{A-17})$$

So, substituting Eq. (A-15) in (A-13) we obtain

$$\partial_{t_2} \rho - \left(\tau - \frac{1}{2} \right) \Delta t \left(c_s^2 \nabla_1^2 \rho + \frac{1 - \theta}{2(\tau - \theta/2)} \nabla_1 \cdot (S_{\lambda}^{(1)} \vec{u}) \right) = \Delta t \left(\tau - \frac{1}{2} \right) \frac{\theta - 1}{2(\tau - \frac{\theta}{2})} \nabla_1 \cdot (S_{\lambda}^{(1)} \vec{u}) . \quad (\text{A-18})$$

Then

$$\partial_{t_2} \rho = \Delta t \left(\tau - \frac{1}{2} \right) c_s^2 \nabla_1^2 \rho , \quad (\text{A-19})$$

and combining Eq. (A-19) with Eq. (A-12), one obtains

$$\partial_t \rho + \nabla \cdot (\rho \vec{u}) = D \nabla^2 \rho + \vec{S} , \quad (\text{A-20})$$

which is the same as Eq. (2-1) with $D = \Delta t \left(\tau - \frac{1}{2} \right) c_s^2$ as the diffusion coefficient.

Bibliography

- [1] Timm Krueger, Halim Kusumaatmaja, Alexandre Kuzmin, Orest Shardt, Goncalo Silva, and Erlend Magnus Viggen. *The Lattice Boltzmann Method: Principles and Practice*. Graduate Texts in Physics. Springer, 2016.
- [2] Jr. Stephen Jarvis. On the diffusion of an ion sheet in poiseuille flow. *Journal of Research of the National Bureau of Standards - B. Mathematical Sciences*, 1968.
- [3] R.B. Bird, W.E. Stewart, and E.N. Lightfoot. *Transport Phenomena*. A Wiley International edition. Wiley, 2007.
- [4] John C Strikwerda. *Finite difference schemes and partial differential equations*. SIAM, 2004.
- [5] Hans-G Roos, Martin Stynes, and Lutz Tobiska. *Robust numerical methods for singularly perturbed differential equations. Convection-Diffusion-Reaction and Flow Problems*. Springer-Verlag Berlin Heidelberg, 2008.
- [6] Shiyi Chen and Gary D. Doolen. Lattice boltzmann method for fluid flows. *Annual Review of Fluid Mechanics*, 30:329–364, Jan 1998.
- [7] FJ Higuera et al. Boltzmann approach to lattice gas simulations. *EPL (Europhysics Letters)*, 9(7):663, 1989.
- [8] R. Benzi, S. Succi, and M. Vergassola. The lattice boltzmann equation: theory and applications. *Physics Reports*, 222(3):145–197, 1992.
- [9] YH Qian, Dominique d’Humières, and Pierre Lallemand. Lattice bgk models for navier-stokes equation. *EPL (Europhysics Letters)*, 17(6):479, 1992.
- [10] Dominique d’Humières. Multiple-relaxation-time lattice boltzmann models in three dimensions. *Philosophical Transactions of the Royal Society of London A: Mathematical, Physical and Engineering Sciences*, 360(1792):437–451, 2002.

- [11] Pierre Lallemand and Li-Shi Luo. Theory of the lattice boltzmann method: Dispersion, dissipation, isotropy, galilean invariance, and stability. *Phys. Rev. E*, 61:6546–6562, Jun 2000.
- [12] Miller Mendoza. *Relativistic Fluid Dynamics in Complex Systems Doctoral Thesis*. PhD thesis, ETH Zürich, 2012.
- [13] Guy R McNamara and Gianluigi Zanetti. Use of the boltzmann equation to simulate lattice-gas automata. *Physical Review Letters*, 61(20):2332, 1988.
- [14] Andrew K. Gunstensen, Daniel H. Rothman, Stéphane Zaleski, and Gianluigi Zanetti. Lattice boltzmann model of immiscible fluids. *Phys. Rev. A*, 43:4320–4327, Apr 1991.
- [15] Daryl Grunau, Shiyi Chen, and Kenneth Eggert. A lattice boltzmann model for multi-phase fluid flows. *Physics of Fluids A: Fluid Dynamics*, 5(10):2557–2562, 1993.
- [16] B. Chopard, P. O. Luthi, and J. F. Wagen. Lattice boltzmann method for wave propagation in urban microcells. *IEE Proceedings - Microwaves, Antennas and Propagation*, 144(4):251–255, Aug 1997.
- [17] Yan Guangwu. A lattice boltzmann equation for waves. *Journal of Computational Physics*, 161(1):61 – 69, 2000.
- [18] Shiyi Chen, Hudong Chen, Daniel Martnez, and William Matthaeus. Lattice boltzmann model for simulation of magnetohydrodynamics. *Phys. Rev. Lett.*, 67:3776–3779, Dec 1991.
- [19] M Mendoza and JD Munoz. Three-dimensional lattice boltzmann model for magnetic reconnection. *Physical Review E*, 77(2):026713, 2008.
- [20] M Mendoza and JD Munoz. Three-dimensional lattice boltzmann model for electrodynamics. *Physical Review E*, 82(5):056708, 2010.
- [21] E. Ilseven and M. Mendoza. Lattice boltzmann model for numerical relativity. *Phys. Rev. E*, 93:023303, Feb 2016.
- [22] S. Succi and R. Benzi. Lattice boltzmann equation for quantum mechanics. *Physica D: Nonlinear Phenomena*, 69(3):327 – 332, 1993.
- [23] M Mendoza, H J Herrmann, and S Succi. Lattice boltzmann model for electronic structure simulations. *Journal of Physics: Conference Series*, 640(1):012018, 2015.
- [24] EG Flekkøy. Lattice bhatnagar-gross-krook models for miscible fluids. *Physical Review E*, 47(6):4247, 1993.

- [25] Dieter Wolf-Gladrow. A lattice boltzmann equation for diffusion. *Journal of Statistical Physics*, 79(5):1023–1032, Jun 1995.
- [26] Zhaoli Guo, Baochang Shi, and Chuguang Zheng. A coupled lattice bgk model for the boussinesq equations. *International Journal for Numerical Methods in Fluids*, 39(4):325–342, 2002.
- [27] S. Ponce Dawson, S. Chen, and G. D. Doolen. Lattice boltzmann computations for reaction diffusion equations. *The Journal of Chemical Physics*, 98(2):1514–1523, 1993.
- [28] R.G.M. van der Sman and M.H. Ernst. Convection-diffusion lattice boltzmann scheme for irregular lattices. *Journal of Computational Physics*, 160(2):766–782, 2000.
- [29] Xiaoxian Zhang, Anthony G. Bengough, John W. Crawford, and Iain M. Young. A lattice {BGK} model for advection and anisotropic dispersion equation. *Advances in Water Resources*, 25(1):1–8, 2002.
- [30] Baochang Shi, Bin Deng, Rui Du, and Xingwang Chen. A new scheme for source term in LBGK model for convection-diffusion equation. *Computers and Mathematics with Applications*, 55(7):1568–1575, 2008.
- [31] S. Suga. Numerical Schemes Obtained from Lattice Boltzmann Equations for Advection Diffusion Equations. *International Journal of Modern Physics C*, 17:1563–1577, 2006.
- [32] B Chopard, JL Falcone, and J Latt. The lattice boltzmann advection-diffusion model revisited. *The European Physical Journal-Special Topics*, 171(1):245–249, 2009.
- [33] Francesca Nannelli and Sauro Succi. The lattice boltzmann equation on irregular lattices. *Journal of Statistical Physics*, 68(3):401–407, Aug 1992.
- [34] Xiaoyi He and Gary D. Doolen. Lattice boltzmann method on a curvilinear coordinate system: Vortex shedding behind a circular cylinder. *Phys. Rev. E*, 56:434–440, Jul 1997.
- [35] Olga Filippova and Dieter Hänel. Grid refinement for lattice-bgk models. *Journal of Computational Physics*, 147(1):219 – 228, 1998.
- [36] C. Shu, Y. T. Chew, and X. D. Niu. Least-squares-based lattice boltzmann method: A meshless approach for simulation of flows with complex geometry. *Phys. Rev. E*, 64:045701, Sep 2001.
- [37] M. Mendoza, S. Succi, and H. J. Herrmann. Flow through randomly curved manifolds. *Scientific Reports*, pages 1 – 6, 2013.
- [38] J-D Debus, M Mendoza, and HJ Herrmann. Dean instability in double-curved channels. *Physical Review E*, 90(5):053308, 2014.

- [39] J-D Debus, M Mendoza, S Succi, and HJ Herrmann. Energy dissipation in flows through curved spaces. *Scientific Reports*, 7, 2017.
- [40] A.M. Velasco, J.D. Muñoz, and M. Mendoza. Lattice boltzmann model for the simulation of the wave equation in curvilinear coordinates. *Journal of Computational Physics*, 376:76 – 97, 2019.
- [41] Budinksy Ljubomit. Lattice boltzmann method for 2d flows in curvilinear coordinates. *Journal of Hydroinformatics*, 2012.
- [42] Hiroaki Yoshida and Makoto Nagaoka. Lattice Boltzmann method for the convection-diffusion equation in curvilinear coordinate systems. *Journal of Computational Physics*, 257:884–900, 2014.
- [43] D. F. Lawden. *Introduction to tensor calculus, relativity and cosmology*. John Wiley and Sons, 1982.
- [44] Xiaowen Shan. Simulation of rayleigh-bénard convection using a lattice boltzmann method. *Physical Review E*, 55(3):2780, 1997.
- [45] Abdulmajeed A Mohamad. *Lattice Boltzmann method: fundamentals and engineering applications with computer codes*. Springer Science & Business Media, 2011.
- [46] P. L. Bhatnagar, E. P. Gross, and M. Krook. A model for collision processes in gases. i. small amplitude processes in charged and neutral one-component systems. *Phys. Rev.*, 94:511–525, May 1954.
- [47] Qisu Zou, Shuling Hou, Shiying Chen, and Gary D. Doolen. A improved incompressible lattice boltzmann model for time-independent flows. *Journal of Statistical Physics*, 81(1):35–48, Oct 1995.
- [48] Xiaoyi He and Li-Shi Luo. Lattice boltzmann model for the incompressible navier–stokes equation. *Journal of Statistical Physics*, 88(3):927–944, Aug 1997.
- [49] Like Li, Renwei Mei, and James F. Klausner. Lattice boltzmann models for the convection-diffusion equation: D2q5 vs d2q9. *International Journal of Heat and Mass Transfer*, 108, 5 2017.
- [50] Li-Shi Luo. Unified Theory of Lattice Boltzmann Models for Nonideal Gases. *Physical Review Letters*, 81(8):1618–1621, 1998.
- [51] Zhaoli Guo, Chuguang Zheng, and Baohang Shi. Discrete lattice effects on the forcing term in the lattice Boltzmann method. *Physical Review E - Statistical, Nonlinear, and Soft Matter Physics*, 65(4):1–6, 2002.

- [52] Xiaowen Shan and Hudong Chen. Simulation of nonideal gases and liquid-gas phase transitions by the lattice Boltzmann equation. *Physical Review E*, 49(4):2941–2948, 1994.
- [53] Christian Huber, Andrea Parmigiani, Bastien Chopard, Michael Manga, and Olivier Bachmann. Lattice boltzmann model for melting with natural convection. *International Journal of Heat and Fluid Flow*, 29(5):1469–1480, 2008.
- [54] Donna Calhoun and Randall J. LeVeque. A cartesian grid finite-volume method for the advection-diffusion equation in irregular geometries. *Journal of Computational Physics*, 157(1):143 – 180, 2000.
- [55] Simon Bogner, Regina Ammer, and Ulrich Rüde. Boundary conditions for free interfaces with the lattice Boltzmann method. *Journal of Computational Physics*, 297:1–12, 2015.
- [56] Zhi-Gang Feng and Efstatios E Michaelides. The immersed boundary-lattice boltzmann method for solving fluid-particles interaction problems. *Journal of Computational Physics*, 195(2):602–628, 2004.
- [57] Stephen R. Karpik and Stephen R. Crockett. Semi-lagrangian algorithm for two-dimensional advection-diffusion equation on curvilinear coordinate meshes. *Journal of Hydraulic Engineering*, 123(5):389–401, 1997.
- [58] J. M. A. Ashbourn, L. Geris, A. Gerisch, and C. J. S. Young. Numerical simulation of two-dimensional and three-dimensional axisymmetric advection-diffusion systems with complex geometries using finite-volume methods. *Proceedings of the Royal Society of London A: Mathematical, Physical and Engineering Sciences*, 466(2118):1621–1643, 2010.
- [59] Sumesh P. Thampi, Santosh Ansumali, R. Adhikari, and Sauro Succi. Isotropic discrete laplacian operators from lattice hydrodynamics. *Journal of Computational Physics*, 234(1):1–7, 2013.
- [60] C. Cercignani and G.M. Kremer. *The Relativistic Boltzmann Equation: Theory and Applications*. Progress in Mathematical Physics. Birkhäuser Basel, 2002.
- [61] E.E. Ferguson, F.C. Fehsenfeld, and A.L. Schmeltekopf. Flowing afterglow measurements of ion-neutral reactions. In D.R. Bates and Immanuel Estermann, editors, *Advances in Atomic and Molecular Physics*, volume 5 of *Advances in Atomic and Molecular Physics*, pages 1 – 56. Academic Press, 1969.
- [62] W.R. Dean M.A. Lxxii. the stream-line motion of fluid in a curved pipe (second paper). *The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science*, 5(30):673–695, 1928.

- [63] S. Murata, Y. Miyake, and T. Inaba. Laminar flow in a curved pipe with varying curvature. *Journal of Fluid Mechanics*, 73(4):735–752, 1976.
- [64] Y. Zhan, F. Ladeinde, Kirk HG., and McDonald KT. The effects of pipe geometry on fluid flow in a muon collider particle production system. *Journal of Fluids Engineering*, 136(10):101203–101203–12, 2014.